CORE Year 1 Whole Genome Sequencing Final Data Format Requirements

To all incumbent contractors of CORE year 1 WGS contracts, the following acts as the agreed to sample parameters issued by NHLBI for data to be delivered to TOPMed directed IRC.

30X CRAM data delivered will incorporate 4-bin quality score compression. The 2-6 scores correspond to Illumina error codes and will be left as-is by recalibration. Bin base quality scores by rounding off to the nearest bin value, in probability space. The 4-bin scheme is:

Bin range	Assigned Bin value
2	2
3	3
4	4
5	5
6	6
7 - 12	10
13 - 22	20
23+	30

The following describes TOPMed's CORE Year 1 protocol for mapping TOPMed whole genome sequence data to build 38. It extracts salient points from the PipelineStandard.md document jointly developed by TOPMed and the Centers for Common Disease Genomics and available as: https://github.com/CCDG/Pipeline-Standardization/blob/master/PipelineStandard.md. That file provides more detailed explanation of the steps than space allows here. The version that was obtained at the issuance of this document has been included as Appendix 1 and is the official reference for CORE year 1 IDIQ issuance.

Read mapping:

Align using bwa mem version 0.7.15 to the 1000 Genomes Project GRCh38DH human genome reference sequence that includes decoy sequences and alternate versions of the HLA locus.

This is the file GRCh38_full_analysis_set_plus_decoy_hla.fa, available from directory: ftp://ftp.1000genomes.ebi.ac.uk/vol1/ftp/technical/reference/GRCh38_reference_genome/.

Use bwa mem parameters -Y -K 100000000 and not -M, as explained in PipelineStandard.md.

Provide a .alt file which identifies the alternate contigs for bwa.

Use samblaster version 0.1.24 or later with arguments -a –addMateTags, or another tool, to add MC and MQ tags to the bwa mem output.

Then sort by genome coordinates and merge multiple readgroups into one file per sample.

Further processing:

Mark duplicate reads using Picard MarkDuplicates or an equivalent program. This should match Picard's current definition of duplicates for primary alignments where both reads of a pair align to the reference genome. It should mark a read whose mate is unmapped as a duplicate if there is another read with the same alignment. In this case, the unmapped mate should also be marked as a duplicate. If a primary alignment is marked as duplicate, all supplementary or secondary alignments for that read should also be marked as duplicates.

Do not perform local realignment around indels.

Do recalibrate base call quality scores using either GATK BaseRecalibrator or bamUtil dedup, masking the known sites found in the three files: Homo_sapiens_assembly38.dbsnp138.vcf, Mills_and_1000G_gold_standard.indels.hg38.vcf.gz and

Homo_sapiens_assembly38.known_indels.vcf.gz from the GATK hg38 resource bundle. Report recalibrated base call qualities using the 4 bin system described above.

Details for recalibrating using GATK are given in the PipelineStandard.md document. If using bamUtil, the following command line simultaneously marks duplicates and recalibrates base call quality scores according to the 4 bin scheme:

```
BamUtil dedup_LowMem  --recab  --dbsnp  Homo_sapiens_assembly38.dbsnp138.vcf.gz
  --allReadNames  --binCustom  --binQualS 0:2,3:3,4:4,5:5,6:6,7:10,13:20,23:30
```

Final file format:

The final sequence data file should be in lossless CRAM format, coordinate sorted, with one file for each sequenced sample. When converted to BAM, the BAM file should be valid according to Picard's ValidateSamFile. Every read should carry an RG read group tag. Read group header lines should contain at least the tags ID, PL, PU, LB and SM. Tag CN is recommended. Retain @PG records from all processing steps. Retain original query names. Do not retain OQ (original quality) information.

If there are questions in implementation, please include your COR, the admin COR (George Papanicolaou), and the CO (Jennifer Swift) in all emails. You may reach out to Tom Blackwell at the IRC for clarification of these standards via email. Always remember that the COR and CO must approve any changes in contract implementation.

-George J. Papanicolaou, PhD
CORE COR Admin
5/23/17

Appendix 1: PipelineStandard.md

This pipeline standard was developed to aid in coordination of the Centers for Common Disease Genomics project. It was tested with HiSeq X data on pipeline implementations from five centers.

# Alignment pipeline standards

## Reference genome version

Each center should use exactly the same reference genome.

Standard:

* GRCh38DH, [1000 Genomes Project version](http://ftp.1000genomes.ebi.ac.uk/vol1/ftp/technical/reference/GRCh38_reference_genome/)

* Includes the standard set of chromosomes and alternate sequences from GRCh38

* Includes the decoy sequences

* Includes additional alternate versions of the HLA locus

## Alignment

Each center should use exactly the same alignment strategy

Standard:

* Aligner: BWA-MEM

* Version: We will use 0.7.15 (https://github.com/lh3/bwa/releases/tag/v0.7.15)

* Standardized parameters:

    * Do not use `-M` since it causes split-read alignments to be marked as "secondary" rather than "supplementary" alignments, violating the BAM specification

    * Use `-K 100000000` to achieve deterministic alignment results (Note: this is a hidden option)

    * Use `-Y` to force soft-clipping rather than default hard-clipping of supplementary alignments

    * Include a `.alt` file for consumption by BWA-MEM; do not perform post-processing of alternate alignments

* Optional parameters (may be useful for convenience and not expected to alter results):

    * `-p` (for interleaved fastq)

    * `-C` (append FASTA/FASTQ comment to SAM output)

    * `-v` (logging verbosity)

    * `-t` (threading)

    * `-R` (read group header line)

* Post-alignment modification:

    * In order to reduce false positive calls due to bacterial contamination randomly aligning to the human genome, reads and their mates may be marked by setting 0x4 bit in the SAM flag if the following conditions apply:

        1. The primary alignment has less than 32 aligned bases

        2. The primary alignment is soft clipped on both sides

    * This filtering is optional

    * The original mapping information will be encoded in a Previous Alignment (PA) tag on the marked reads using the same format as the SA tag in the BAM specification.

    * Modification of other flags after alignment will not be performed.

## Duplicate marking

Different centers can use different tools, as long as the same number of reads are marked duplicate and results are functionally equivalent. During the pipeline synchronization exercise we evaluated four tools: Picard MarkDuplicates, bamUtil, samblaster, and sambamba. After the exercise, centers are using Picard and bamUtil.

Standard:

* Match Picard's current definition of duplicates for primary alignments where both reads of a pair align to the reference genome. Both samblaster and bamUtil already attempt to match Picard for this class of alignments.

* If a primary alignment is marked as duplicate, then all supplementary alignments for that same read should also be marked as duplicates. Both Picard and bamUtil have modified to exhibit this behavior. For Picard, you must use >= version 2.4.1 and run on a queryname sorted input file. BamUtil must be version >=TODO. Samblaster supports this behavior, but Sambamba does not.

* Orphan alignments (where the mate paired read is unmapped) will be marked as duplicates if there's another read with the same alignment (mated, or orphaned)

* The unmapped mate of duplicate orphan reads is required to also be marked as a duplicate.

* It is not a requirement for duplicate marking software to choose the best pair based on base quality sum, but results must be functionally equivalent. In practice we have moved away from using samblaster for this reason.

* If a primary alignment is marked as duplicate, then all secondary alignments for that read should also be marked as duplicates. However, given that no secondary alignments will exist using our proposed alignment strategy, it is optional for software to implement.

* There was a discussion about whether duplicate marking should be deterministic. We did not reach a decision on this.

* We have discussed the preferred behavior for marking duplicates in datasets with multiple sequencing libraries and have decided that this is a minor concern given that very few samples should have multiple libraries. Currently MarkDuplicates supports multiple libraries with the caveat that the term "Library" isn't exactly defined (consider a technical replicate that starts somewhere in the middle of the LC process, how early must it be to be called a different library?)

## Indel realignment

This computationally expensive data processing step is dispensable given the state of current indel detection algorithms and will not be performed.

## Base quality score recalibration

There was discussion about dropping BQSR given evidence that the impact on variant calling performance is minimal. However, given that this project will involve combined analysis of data from multiple centers and numerous sequencers, generated over multiple years, and that we cannot ensure the consistency of Illumina base-calling software over time, we decided that it is preferable to perform BQSR. We evaluated two tools, GATK BaseRecalibrator (both GATK3 and GATK4) and bamUtil.

Standard:

* We will use the following files from the [GATK hg38 bundle](https://console.cloud.google.com/storage/browser/genomics-public-data/resources/broad/hg38/v0/) for the site list:

    * Homo_sapiens_assembly38.dbsnp138.vcf

    * Mills_and_1000G_gold_standard.indels.hg38.vcf.gz

    * Homo_sapiens_assembly38.known_indels.vcf.gz

* The recalibration table may optionally be generated using only the autosomes (chr1-chr22)

* Downsampling of the reads is optional

* per-base alignment qualities (BAQ) algorithm is optional

Command line:

For users of GATK, the following command line options should be utilized for the BaseRecalibrator tool:

```

-R ${ref_fasta} \

-I ${input_bam} \

-O ${recalibration_report_filename} \

-knownSites "Homo_sapiens_assembly38.dbsnp138.vcf" \

-knownSites “Mills_and_1000G_gold_standard.indels.hg38.vcf.gz" \

-knownSites “Homo_sapiens_assembly38.known_indels.vcf.gz”

```

For users of GATK, the following command line options are optional for efficiency and can be utilized for the BaseRecalibrator tool:

Note: we've tested .1 downsampling fractions. Lower fractions should be tested for functional equivalence.

```

--downsample_to_fraction .1 \

   -L chr1 \

   -L chr2 \

   -L chr3 \
    -L chr4 \

    -L chr5 \

    -L chr6 \

    -L chr7 \

    -L chr8 \

    -L chr9 \

    -L chr10 \

    -L chr11 \

    -L chr12 \

    -L chr13 \

    -L chr14 \

    -L chr15 \

    -L chr16 \

    -L chr17 \

    -L chr18 \

    -L chr19 \

    -L chr20 \

    -L chr21 \

    -L chr22
```

For users of GATK, the following command line options are optional:

* `-rf BadCigar`

* `--preserve_qscores_less_than 6`

* `--disable_auto_index_creation_and_locking_when_reading_rods`

* `--disable_bam_indexing`

* `-nct`

* `--useOriginalQualities`

## Base quality score binning scheme

Additional base quality score compression is required to reduce file size. It is possible to achieve this with minimal adverse impacts on variant calling.

Standard:

* 4-bin quality score compression. The 4-bin scheme is 2-6, 10, 20, 30. The 2-6 scores correspond to Illumina error codes and will be left as-is by recalibration.

* Bin base quality scores by rounding off to the nearest bin value, in probability space. This feature is already implemented in the current version of GATK.

Command line:

For users of GATK, the following command line options should be utilized for the PrintReads (GATK3) or ApplyBQSR (GATK4) tool:

```

-R ${ref_fasta} \

-I ${input_bam} \

-O ${output_bam_basename}.bam \

-bqsr ${recalibration_report} \

-SQQ 10 -SQQ 20 -SQQ 30 \

--disable_indel_quals

```

For users of GATK, the following command line options are optional:

* `--globalQScorePrior -1.0`

* `--preserve_qscores_less_than 6`

* `--useOriginalQualities`

* `-nct`

* `-rf BadCigar`

* `--createOutputBamMD5`

* `--addOutputSAMProgramRecord`

## File format

Each center should use the same file format, while retaining flexibility to include additional information for specific centers or projects.

Standard:

* Lossless CRAM. Upon conversion to BAM, the BAM file should be valid according to Picard's ValidateSamFile.

* Read group (@RG) tags should be present for all reads.

   * The header for the RG should contain minimally the ID tag, PL tag, PU tag, SM tag, and LB tag.

   * The CN tag is recommended.

   * Other tags are optional.

   * The ID tag must be unique within the CRAM. ID tags may be freely renamed to maintain uniqueness when merging CRAMs. No assumptions should be made about the permanence of RG IDs.

   * The PL tag should indicate the instrument vendor name according to the SAM spec (CAPILLARY, LS454, ILLUMINA, SOLID, HELICOS, IONTORRENT, ONT, and PACBIO). PL values are case insensitive.

   * The PU tag is used for grouping reads for BQSR and should uniquely identify reads as belonging to a sample-library-flowcell-lane (or other appropriate recalibration unit) within the CRAM file. PU is not required to contain values for fields that are uniform across the CRAM (e.g., single sample CRAM or single library CRAM). The PU tag is not guaranteed to be sufficiently informative after merging with other CRAMs, and anyone performing a merge should consider modifying PU values appropriately.

   * SM should contain the individual identifier for the sample (e.g., NA12878) without any other process or aliquot-specific information.

   * The LB tag should uniquely identify the library for the sample; it must be present even if there is only a single library per sample or CRAM file.

   * If the PM tag is used, values should conform to one of the following (for Illumina instruments): “HiSeq-X”, “HiSeq-4000”, “HiSeq-2500”, “HiSeq-2000”, “NextSeq-500”, or “MiSeq”.

* Retain original query names.

* Retain @PG records for bwa, duplicate marking, quality recalibration, and any other tools that was run on the data.

* Retain the minimal set of tags (RG, MQ, MC and SA). NOTE: an additional tool may be needed to add the MQ and MC tags if none of the tools add these tags otherwise. One option is to pipe the alignment through [samblaster](https://github.com/GregoryFaust/samblaster) with the options `-a --addMateTags` as it comes out of BWA

* Groups can add custom tags as needed.

* Do not retain the original base quality scores (OQ tag).

* it is recommended that users use samtools version >=1.3.1 to convert from BAM/SAM to CRAM (The use of htsjdk/Picard/GATK for converting BAM to CRAM is not currently condoned). Users that would like to convert back from CRAM to BAM (and want to avoid ending up with an invalid BAM) need to either convert to SAM and then to BAM (piping works) or compile samtools with HTSLib version >=1.3.2. To enable this you need to: configure the build of samtools with the parameter `--with-htslib=/path/to/htslib-1.3.2`.

# Functional equivalence evaluation

All pipelines used for this effort need to be validated as functionally equivalent. The validation methodology will be published alongside a test data set.

# Pathway for updates to this standard

Pipelines will need to be updated during the project, but this should be a tightly controlled process given the need to reprocess vast amounts of data each time substantial pipeline modifications occur.

(BELOW CANNOT BE IMPLEMENTED ON THIS CONTRACT WITHOUT CONSULATION AND WRITTEN APPROVAL OF NHLBI)

Draft plan:

* Initial pipeline versions should serve for project years 1-2.

* Efficiency updates passing functional equivalence tests are always allowed.

* Propose to start a review process in late 2017: invite proposals for pipeline updates that incorporate new aligners, reference genomes and data processing steps.

* If substantial improvements are achievable, implement new pipelines for project years 3-4.

* There will need to be a decision about how large the potential variant calling improvements should be to warrant pipeline modification and data reprocessing.