

# iPCR pipeline

```
graph TD
    subgraph Inputs
        dbSNP_138_VCF["dbSNP_138.VCF  
+ SNPs  
+ Indels"]
        frag_rev_info["fragment_rev.info  
+ read_id: string  
+ adapter: string"]
        frag_forw_info["fragment_forw.info  
+ read_id: string  
+ barcode_length: int  
+ barcode: string  
+ sample_id: string"]
        sorted_bam["<sample>.sorted.bam  
+ read_id: string  
+ positions: int  
+ map_quality: int"]
    end

    subgraph GATK_Steps
        GATK_GVCFs1["GATK: GenotypeGVCFs"]
        GATK_SelectVariants1["GATK: SelectVariants"]
        GATK_GVCFs2["GATK: GenotypeGVCFs"]
        GATK_SelectVariants2["GATK: SelectVariants"]
        GATK_GVCFs3["GATK: GenotypeGVCFs"]
    end

    subgraph VCF_Outputs
        raw_callset_VCF["raw_callset.VCF  
+ SNPs  
+ Indels"]
        raw_callset_indels_VCF["raw_callset.indels.VCF  
+ Indels"]
        raw_callset_snps_VCF["raw_callset.snps.VCF  
+ SNPs"]
        no_alleles_info["no_alleles.info  
+ seqId: string  
+ start: int  
+ stop: int  
+ readId: string  
+ orientation: bool  
+ seqForward: string  
+ seqReverse: string  
+ cigarForward: string  
+ cigarReverse: string  
+ mapQuality: int  
+ barcode: string  
+ readCount: string  
+ sampleId: string"]
    end

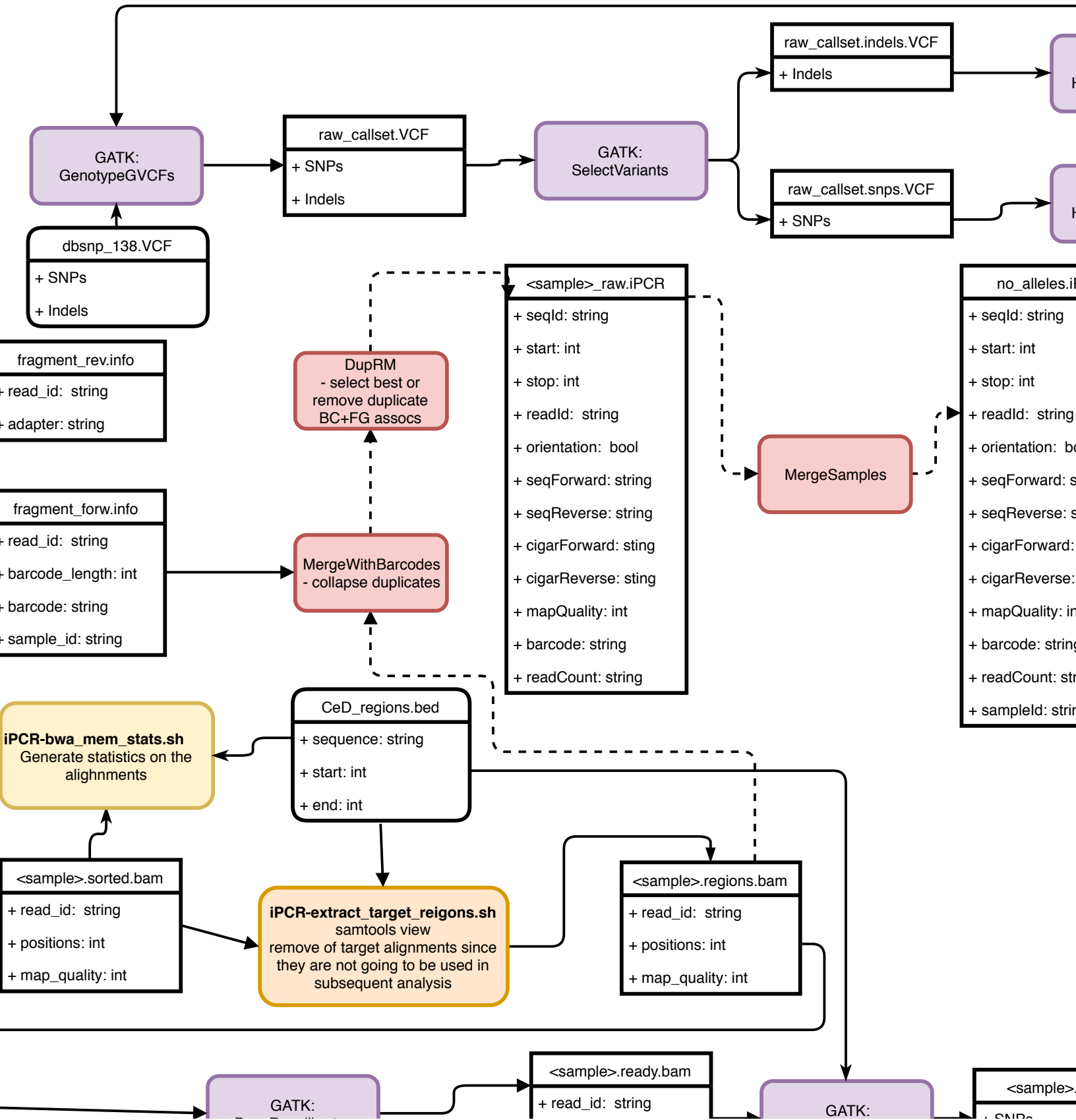
    subgraph BAM_Outputs
        regions_bam["<sample>.regions.bam  
+ read_id: string  
+ positions: int  
+ map_quality: int"]
        ready_bam["<sample>.ready.bam  
+ read_id: string"]
    end

    subgraph Scripts
        iPCR_bwa_stats["iPCR-bwa_mem_stats.sh  
Generate statistics on the alignments"]
        iPCR_extract_regions["iPCR-extract_target_reigons.sh  
samtools view  
remove of target alignments since they are not going to be used in subsequent analysis"]
        MergeWithBarcodes["MergeWithBarcodes  
- collapse duplicates"]
        DupRM["DupRM  
- select best or remove duplicate BC+FG assocs"]
        MergeSamples["MergeSamples"]
    end

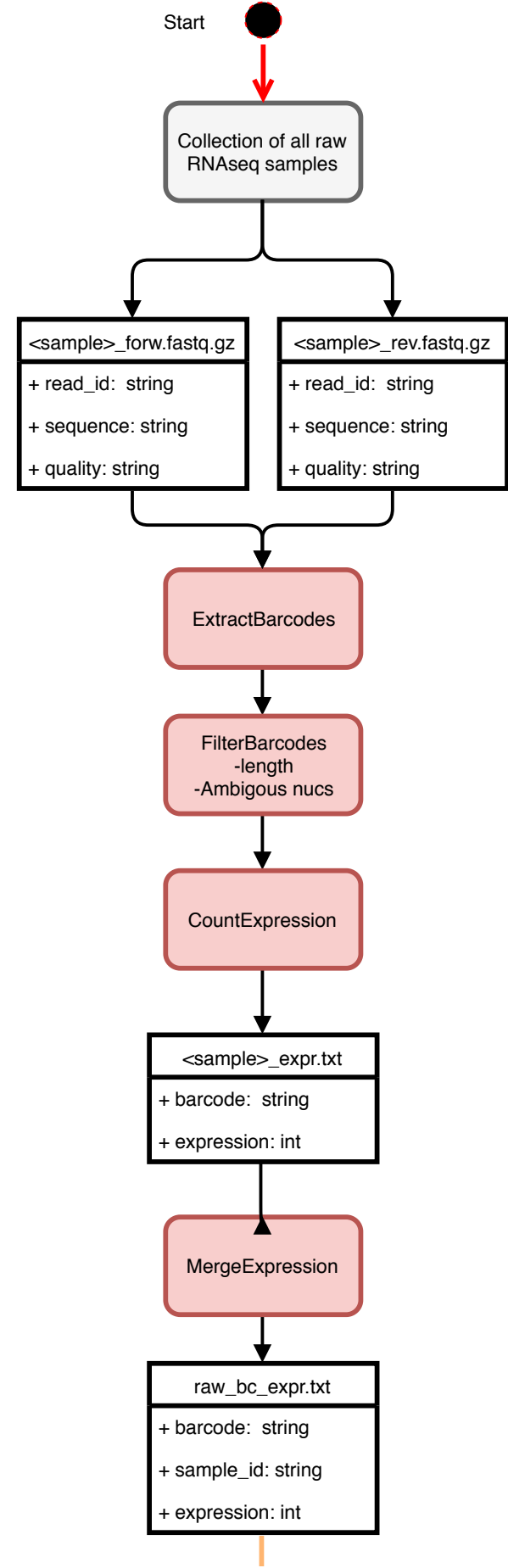
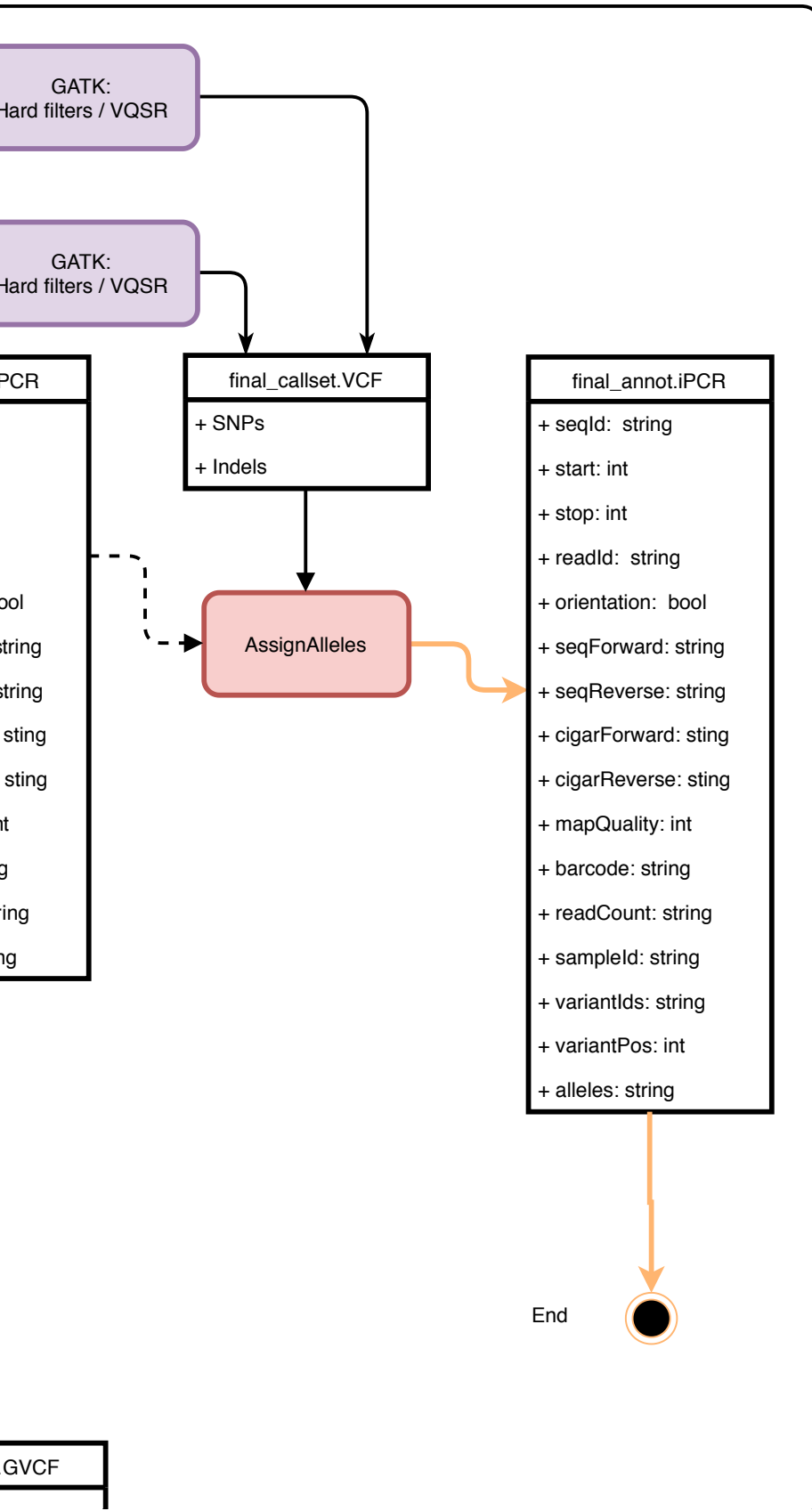
    subgraph BED_Outputs
        CeD_regions_bed["CeD_regions.bed  
+ sequence: string  
+ start: int  
+ end: int"]
    end

    dbSNP_138_VCF --> GATK_GVCFs1
    frag_rev_info --> GATK_GVCFs1
    frag_forw_info --> GATK_GVCFs1
    sorted_bam --> GATK_GVCFs1
    GATK_GVCFs1 --> raw_callset_VCF
    raw_callset_VCF --> GATK_SelectVariants1
    GATK_SelectVariants1 --> raw_callset_indels_VCF
    GATK_SelectVariants1 --> raw_callset_snps_VCF
    raw_callset_snps_VCF --> GATK_GVCFs2
    GATK_GVCFs2 --> no_alleles_info
    raw_callset_indels_VCF --> GATK_SelectVariants2
    GATK_SelectVariants2 --> regions_bam
    regions_bam --> GATK_GVCFs3
    GATK_GVCFs3 --> ready_bam
    ready_bam --> GATK_GVCFs3
    GATK_GVCFs3 --> SNPs

    sorted_bam --> iPCR_bwa_stats
    sorted_bam --> iPCR_extract_regions
    sorted_bam --> MergeWithBarcodes
    MergeWithBarcodes --> DupRM
    DupRM --> MergeSamples
    MergeSamples --> regions_bam
    sorted_bam --> CeD_regions_bed
    CeD_regions_bed --> iPCR_extract_regions
    CeD_regions_bed --> MergeWithBarcodes
    CeD_regions_bed --> MergeSamples
    CeD_regions_bed --> GATK_GVCFs3
```



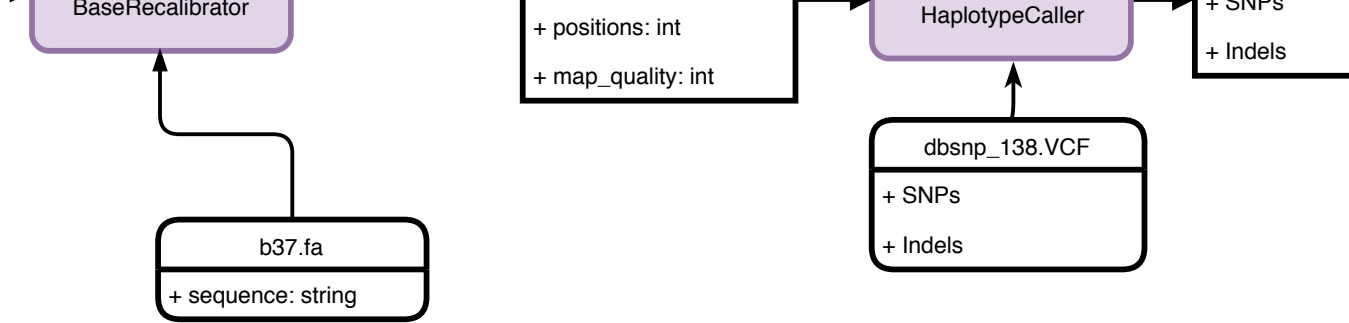
# cDNA





+ positions: int  
+ map\_quality: int

Marking of duplicates may not be ideal since the complexity of the library is limited due to the enrichment of target fragments. For more details:  
<https://gatkforums.broadinstitute.org/gatk/discussion/6747/how-to-mark-duplicates-with-markduplicates-or-markduplicateswithmatecigar>



End



