# CDER Scientific Computing Genome Analysis Toolkit (GATK) Pipeline On CDRH HPC Cluster

SCD20

September 30, 2020

**Xiaoqing He** 





# CDER GATK Pipeline Programming Platform is specifically designed to promote Next Generation Sequencing (NGS) analytics on High Performance Computing (HPC) cluster

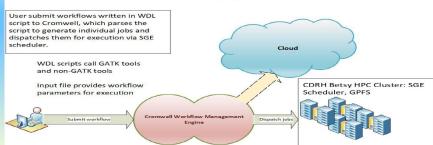
### **CDER GATK Pipeline in Detail**

The GATK Pipeline Programming Platform has been installed and configured on CDRH Betsy HPC cluster that consists of the following:

- · A workflow management engine (Cromwell)
- Workflow Description Language (WDL)
- Programming Cromwell Sun Grid Engine (SGE) backend for local HPC
- Programming GATK Best Practice workflows using GATK pipeline programming platform

### **Overview of GATK Pipeline Programming Platform**

# CDER GATK Pipeline Flow Chart



### **Pipelining GATK With Cromwell/WDL**

Cromwell - an open sourced Workflow Management System geared towards scientific workflows.

- Used by Broad Institute GATK Best Practice Pipelines.
- Executes tasks defined in WDL scripts.
- Uses input JSON file to fill in the values of inputs to commands in WDL script where appropriate
- Cromwell has a way to configure Sun Grid Engine(SGE) backend relying on CDRH Betsy High Performance Computing (HPC) frameworks, and with access to General Parallel File System (GPFS)

Cromwell tracks the execution of workflows

Command to run Cromwell: java –jar cromwell.jar run workflow.wdl –i inputs.json

### WDL Script for GATK Workflow

- WDL is a user-friendly scripting language
- WDL scripts defines tasks and call GATK tools to perform workflows.
- A sample WDL script for Sequence data format validation workflow

(validatebamWf.wdl)



### Input JSON File

- Provides input variables to the commands in WDL script, may include the following:
  - file names.
  - path to the input data,
  - the parameters for the GATK tools.
  - any other resources that needed for the workflows.
- Allows you to be able to customize input values from run to run.
- If input variables were hard-coded in WDL scripts, you will have to make a new copy and edit the inputs each time you run your script on a new batch of data -- which undermines the advantages of setting up a pipeline

"##Comment1":"Input"

"VaridiateRemovif.bem\_erroy": [
"/projects01/Cder-sc/gatk-pipeline/gatk-test-data/wgs\_bam/MA12878\_20k\_hg38/MA12878.bam"

"#EComment2":"Parameter"

"#EComment2":"Parameter"

### **Cromwell SGE Backend**

- Cromwell provides a way to configure SGE backend
- It launches jobs using gsub command
- It supports runtime attribute configurations, such as *queue*, *cpu* and *memory-unit*

(cromwell-sge.conf)



### **Sequence Data Format Validation Workflow**

- Purpose: To validate sequence data format
- Workflow Name: seq-format-validation
- Workflow Requirements:
- One or more SAM or BAM files to validate
- Explicit request of either SUMMARY or VERBOSE mode
- Outputs:
- Set of .txt files containing the validation reports, one per input file

### **GATK Best Practice Workflows**

- The GATK Best Practices provide step-by-step recommendations for performing variant discovery analysis
- There are several different GATK Best Practices workflows tailored to particular applications depending on the type
  of variation of interest and the technology employed.
- These are dependent on many factors including sequencing technology and the hardware infrastructure, soone may need to adapt their recommendations to one specific situation
- The following GATK best practice workflows has been uploaded to CDER GATK pipeline project repository
- gatk4-exome-analysis-pipeline.
- gatk4-germline-snps-indels
- gatk4-somatic-snvs-indels
   gatk4-somatic-cnvs
- seq-format-conversion
- gatk4-rnaseq-germline-snps-indels
   gatk4-genome-processing-pipeline
- gatk4-mitochondria-pipeline
- seq-format-validation
   gatk4-data-processing
- e gatk4-germline-cnvs
- gatk4-pathseq

## **Pre-Data Processing for Variant Discovery Workflow**

- Purpose: To read unmapped sequencing data and process the data for variant discovery analysis.
- Workflow Name: processing-for-variant-discovery-gatk4
- Workflow Requirements:
- Pair-end sequencing data in unmapped BAM (uBAM) format
- One or more read groups, one per uBAM file, all belonging to a single sample (SM)
- Input uBAM files must additionally comply with the following requirements:
- The contract deduction of the contract of the
- Filenames all have the same suffix (we use ".unmapped.bam")
- Files must pass validation by ValidateSamFile
- · Reads are provided in query-sorted order
- All reads must have an RG tag- Reference index files must be in the same directory as source (e.g. reference.fasta.fai in the same directory as reference.fasta)
- Outputs: A clean BAM file and its index, suitable for variant discovery analyses.
- Tools: GATK4 and non-GATK tools (BWA, Picard, Samtools 1.3.1 (using htslib 1.3.1), Python)







