Implementing a Scalable JGI-RQC Pipeline on the Cloud

Date: July 30th, 2021
Presenter: Sandra Chacon
Mentors: Bryce Foster & Zhong Wang
JGI is Producing Big Sequencing Data

280 Terabases

A STANDARD STORAGE BOX

2,000 PAGES
10 MB

28 Million Boxes

18,230 Trucks

Dimensions: 15-by-12-by-10-inch

26 Foot Penske Truck Basics
What is RQC Filtering and why is it important?

Short answer: garbage in, garbage out

Legend

<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Unchecked</td>
</tr>
<tr>
<td></td>
<td>Sequence</td>
</tr>
<tr>
<td></td>
<td>Valid</td>
</tr>
<tr>
<td></td>
<td>Trimmed</td>
</tr>
<tr>
<td></td>
<td>Sequence</td>
</tr>
</tbody>
</table>

Clean Up

By Filtering

1. Adapter filtering

2. Valid Sequence

3. Trimmed Sequence

4. Removed Adapter

Discarded Sequence
Genomic Data can have extra information

An example of contamination

Unfiltered Sequencing Data

Filtered Sequencing Data
Current RQC Pipeline Runs on a Single Server

For large datasets it will take a long while

RQC filter Pipeline Runtime Histogram

- Frequency
- Run Time (Hours)
Big Data Industry has a Solution: Apache Spark

Data Parallel Processing
How Does Our Python Program Work?

Test file:

@H100 read 1: ACCATCTC
@H100 read 2: CATGCATG
@H101 read 1: TTCGAGTC
...
@H5000 read 2: CATCATGA

worker evaluates sequence:
H100 read 1

quality test
n test
kmer test

*reads one line at a time
processes one test at a time
Outcomes of Python Program

Big File: HCTUP2.fastq.gz

- Raw Sequence Data
- Filtered Sequence Data
Test file:

@H100 read 1: ACCATCTC
@H100 read 2: CATGCATG
@H101 read 1: TTCGAGTC

... 

@H5000 read 2: CATCATGA

Spark Data Frame

<table>
<thead>
<tr>
<th>Index</th>
<th>DNA Sequence</th>
<th>Quality</th>
<th>Q Test</th>
</tr>
</thead>
<tbody>
<tr>
<td>@H100 read 1</td>
<td>ACCATCTC..</td>
<td>FFFF:::,,</td>
<td>PASS</td>
</tr>
<tr>
<td>@H100 read 2</td>
<td>CATGCATG..</td>
<td>FF,FF:FF;F</td>
<td>PASS</td>
</tr>
<tr>
<td>@H101 read 1</td>
<td>TTCGAGTC..</td>
<td>FF::FF;;</td>
<td>FAIL</td>
</tr>
</tbody>
</table>

Example of tasks being distributed between all available CPUs
Outcomes of Python & Databricks

Comparison of Filtering Programs

Run time (mins)

- Python
- Old Pyspark
- Pyspark

Filtering type
Outcomes of Proposed Solution

What happened with Scala?

ERROR: Failed with Error
......ClassNotFound
Future Directions

Pipeline Improvements

- Improvement in how contamination filtering done
- Research how to make the contamination reference file quick to load for each run.
- Convert more user defined functions into PySpark functions
- PySpark filter pipeline write to local disk
Acknowledgements:

Dr. Zhong Wang  
Department of Energy Joint Genome Institute

Bryce Foster  
Department of Energy Joint Genome Institute

Brian Bushnell  
Department of Energy Joint Genome Institute

Chen Zhang  
Shanghai University (Graduate Student)
THANK YOU TO OUR SPONSORS
THANK YOU FOR LISTENING

QUESTIONS? COMMENTS?