RCBio: easy, quick HPC pipeline builder & runner

HMS Research Computing

Lingsheng Dong: Lingsheng_Dong@hms.harvard.edu
Kathleen Keating: Kathleen_Keating@hms.harvard.edu
RC Help: rchelp@hms.harvard.edu

What is RCBio?

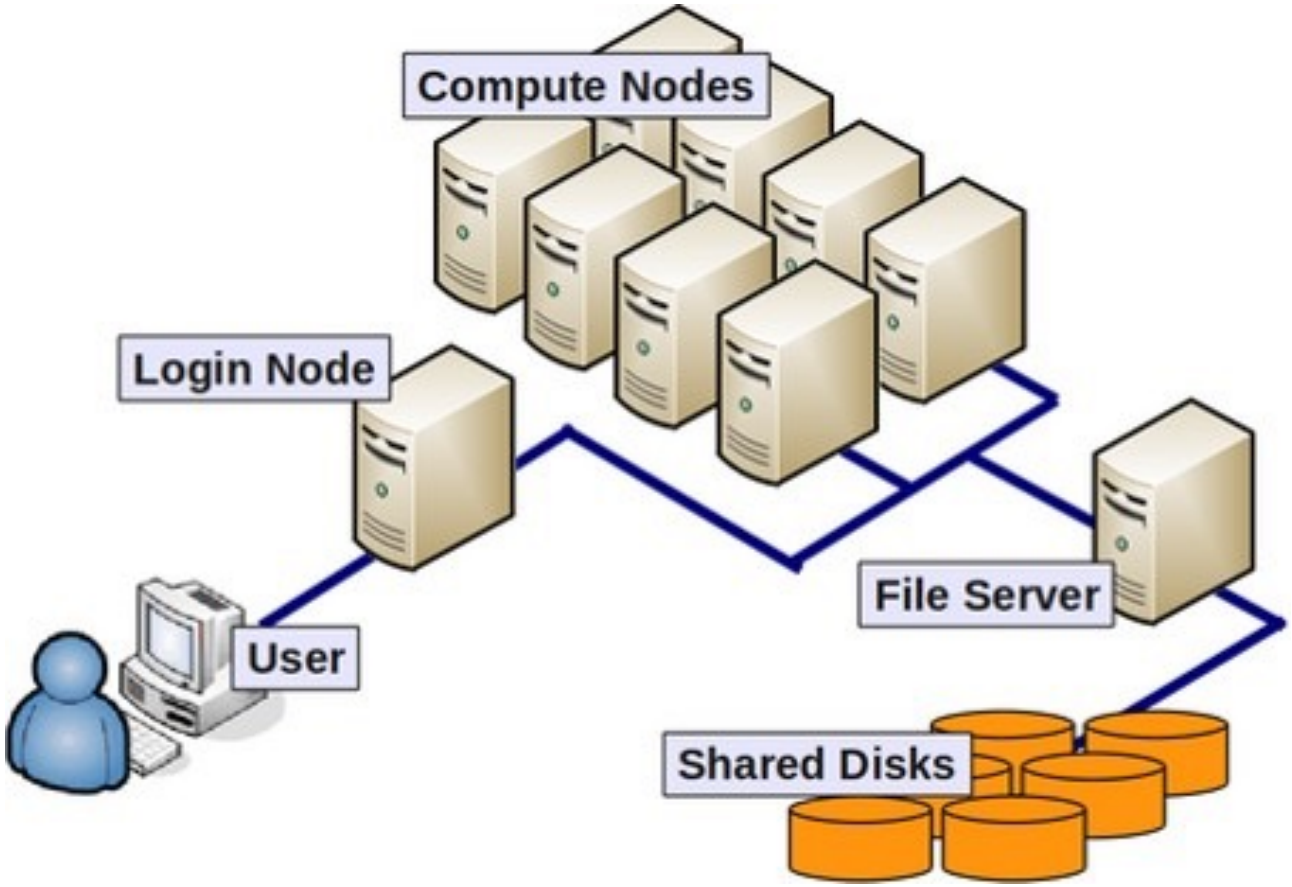
RCBio is a pipeline runner and workflow tool developed and supported by HMS RC consultants.

User comments about RCBio

"PS-- **this pipeline runner is THE most amazing thing**. Super organized, super easy to rerun failed jobs, and so easy to debug right from my email because the failure emails are so informative. Thank you for creating this!!"

"I was able to code these steps up in a single bash script with rcbio pipeline decorators; all tracking and job dependencies were taken care of and I received emails for successful and failed jobs. -- Shilpa Kobren, a postdoc from HMS DBMI

Generic Cluster Architecture



Batch Job submission

```
#!/bin/bash
#SBATCH -p short #partition
#SBATCH -t 0-01:00 #time days-hr:min
#SBATCH -c X #number of cores
#SBATCH --mem=XG #memory per job (all cores), GiB
# put any module load commands here
# put any analysis commands you want to run here
```

Save above as myjobscript.sh, then run:

```
sbatch myjobscript.sh
```

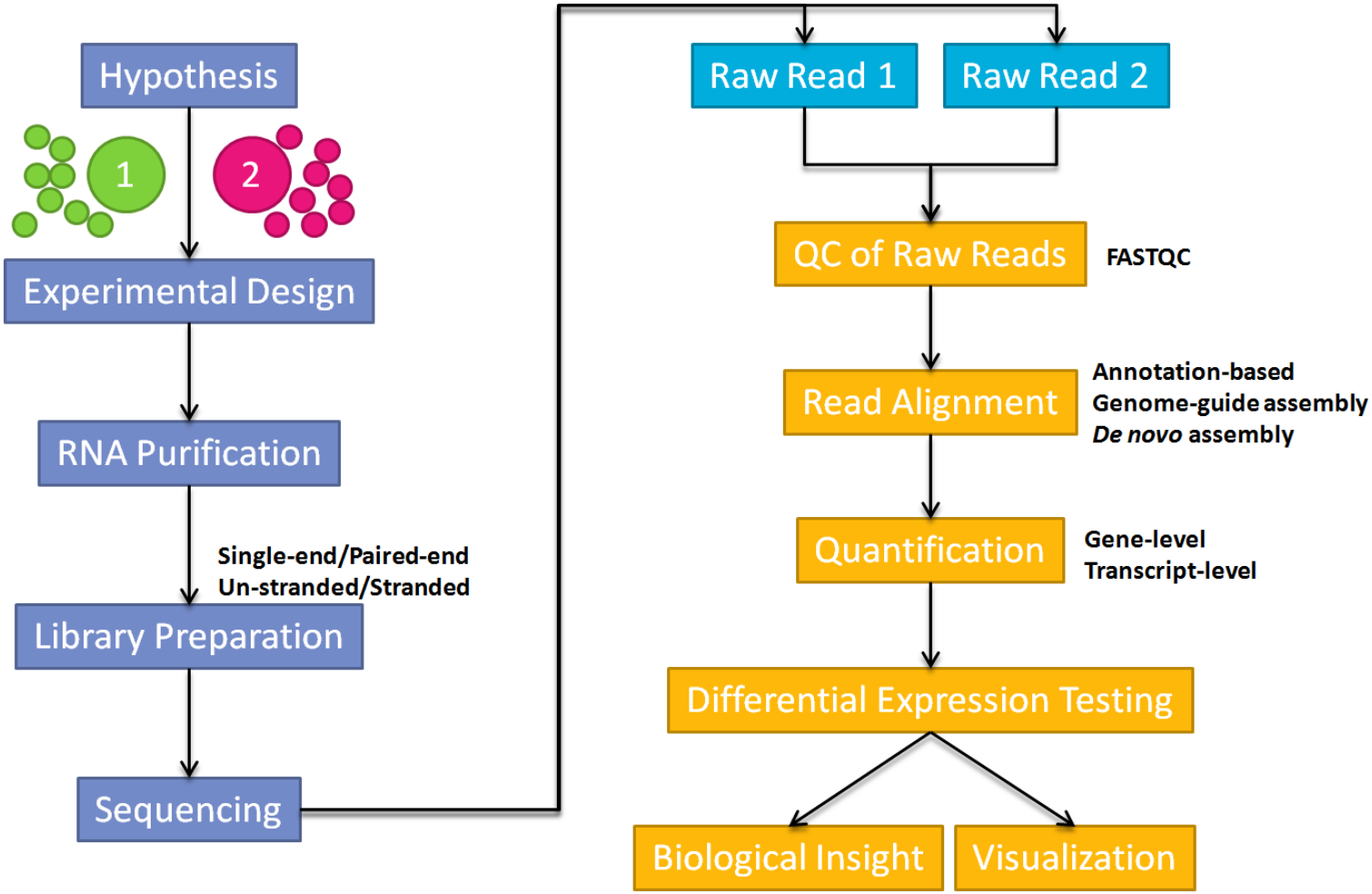
Also you can using –wrap:

```
sbatch –p short –t 0-01:00 -c x –mem 2G –wrap="software…"
```

Job Dependencies

With the –d option, the Slurm scheduler allows you to run jobs with dependencies. That means one job waits to run until another job finished. You can also set a job to run only if the dependency failed, or only if it succeeded.

RNA-Seq Pipeline



-- https://databeauty.com/figures/2016-09-13-RNA-seq-analysis/rna_seq_workflow.png

Continue RNA-Seq pipeline

Assume we have RNA-Seq data files:

```
ld32@login02 /home/ld32/myRNASeq$ ls
sample1_1.fq  sample1_2.fq  sample2_1.fq
sample2_2.fq  sample3_1.fq  sample3_2.fq
sample4_1.fq  sample4_2.fq
```

RNA-Seq pipeline (single computer)

```
#!/bin/bash
for i in 1 2 3 4; do
    sp=sample${1}; r1=${sp}_1.fq; r2=${sp}_2.fq;

    qcCmd  -input1 ${r1}  -input2 ${r2} –output ${sp}.qc.txt

    alignCmd  -cpu 4 –input1 ${r1}  -input2 ${r2} -output   ${sp}.bam

    countCmd   -input   ${sp}.bam    -output     ${sp}.cnt
done

mergeCmd    -input    *.cnt    -output     allCount.txt

statCmd      -input      allCount.txt      -output      finalResult.txt
```

A simpler pipeline

```
for u in universityA.txt ; do

    grep -H John $u >> John.txt

    grep -H Nick $u >> Nick.txt

done

cat John.txt Nike.txt > all.txt
```

Converting to Slurm pipeline

```
for u in universityA.txt; do

    job1ID=$(sbatch -p short -t 5 --wrap "grep -H John $u >> John.txt")

    job2ID=$(sbatch -p short -t 5 --wrap "grep -H Nick $u >> Nick.txt")

done

sbatch -p short -t 5 -d afterok:$job1ID:$job2ID  --wrap "cat \
    John.txt Nike.txt > all.txt"
```

Also keep log and send emails

```
job1ID=$(checkIfItIsDone.sh ||
sbatch -p short -t 5 --mem 2G --
wrap “someCmd && touch
job1.success; sendBetterEmail.sh;”)
```

What are checkIfItIsDone.sh? And sendBetterEmail.sh?

A little more complex

```
for i in A     B     C     E     F;    do

    u=university${i}.txt

    grep -H John $u >> John.txt

    grep -H Nick $u >> Nick.txt

done

cat John.txt Nike.txt > all.txt
```

Can we automate it?

```
for i in A     B     C     E     F ; do
    u=university${i}.txt

    #@ Please submit this command as slurm job with 1G memory,
    #@ 5 minute run time and 1 CPU. And this job can start right away.
    grep -H John $u >> John.txt

    #@ Please submit this command as slurm job with 1G memory,
    #@ 5 minute run time and 1 CPU. And this job can start right away
    grep -H Nick $u >> Nick.txt
Done

#@ Please submit this command as slurm job with 2G memory,
#@ 5 minute run time and 1 CPU. And this job needs to wait for the other 10 jobs
cat John.txt Nike.txt > all.txt
```

RCBio decorator

#@3,1.2,merge,,sbatch -p short -t 5 –c 1 --mem 1G

cat John.txt Nick.txt >> all.txt

#@	→	This row is decorator for RCBio to parse
3	→	Step ID.
1.2	→	The step IDs this step depends on.
merge	→	Step name.
sbatch…	→	sbatch command to use

Can we automate it?

```
for i in A     B     C     E     F ; do
    u=university${i}.txt

    #@1,0,findJohn,u,sbatch -p short -t 5 –c 1 --mem 1G
    grep -H John $u >> John.txt

    #@2,0,findNick,u,sbatch -p short -t 5 –c 1 --mem 1G
    grep -H Nick $u >> Nick.txt
Done

#@3,1.2,merge,,sbatch -p short -t 5 –c 1 --mem 1G
cat John.txt Nike.txt > all.txt
```

Demo session

Go through the wiki page and run a simple pipeline

https://harvardmed.atlassian.net/wiki/spaces/O2/pages/1600651373/RC+workflows

For more direction

- **Email:** rchelp@hms.harvard.edu
- **O2 wiki**: https://harvardmed.atlassian.net/wiki/spaces/O2/overview
- **Website**: http://rc.hms.harvard.edu
- **RC Office Hours**: Wed 1-3p Gordon Hall 500
  - https://rc.hms.harvard.edu/office-hours/ for Zoom web conferencing during remote work