

# Roswell OnDemand

## A One-Stop Shop for High Performance Computing



**L. Shawn Matott, PhD**  
**Senior Systems Analyst**  
**IT Research Computing**

# Training Objectives

- To provide an overview of Roswell Park's IT Research Computing (IT RC) group
- To provide an introduction to Roswell Park's HPC OnDemand interface

# About IT Research Computing

- Mission Statement

*To promote the use of advanced computing and computational science to enable and accelerate research throughout the Roswell Park campus.*

- Expertise and Services

- HPC assets
- Training
- Grant Support
- IT Referrals

# Computing Assets

- Roswell Park HPCC

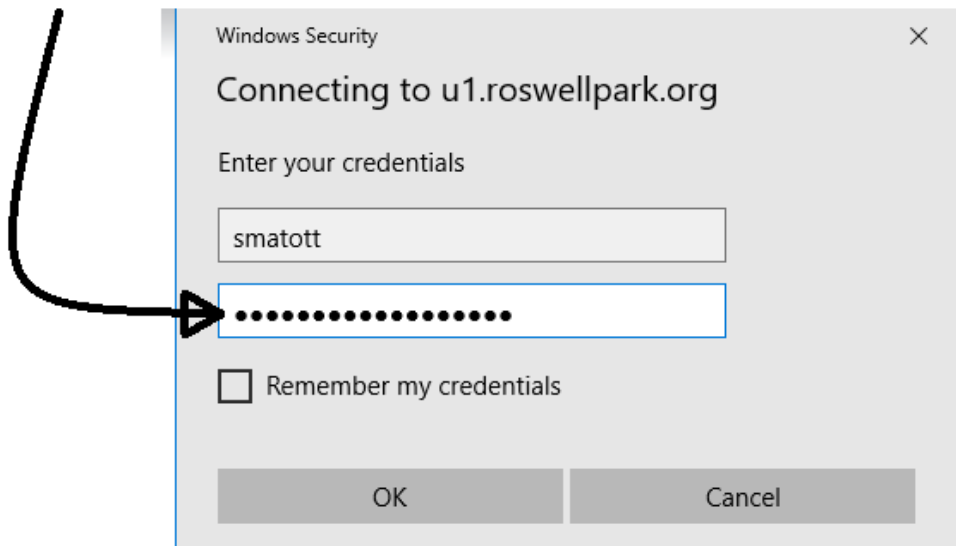
- 1,600 processors, 450 TB Lustre storage
- Software modules include: R/Bioconductor, Python, Keras, and TensorFlow, many bio-informatics packages
- OnDemand web interface  
<http://u1.roswellpark.org/pun/sys/dashboard/>
- e-mail [ServiceDesk@RoswellPark.org](mailto:ServiceDesk@RoswellPark.org) to request access



*A look inside Roswell's HPC data center.*

# OnDemand Login

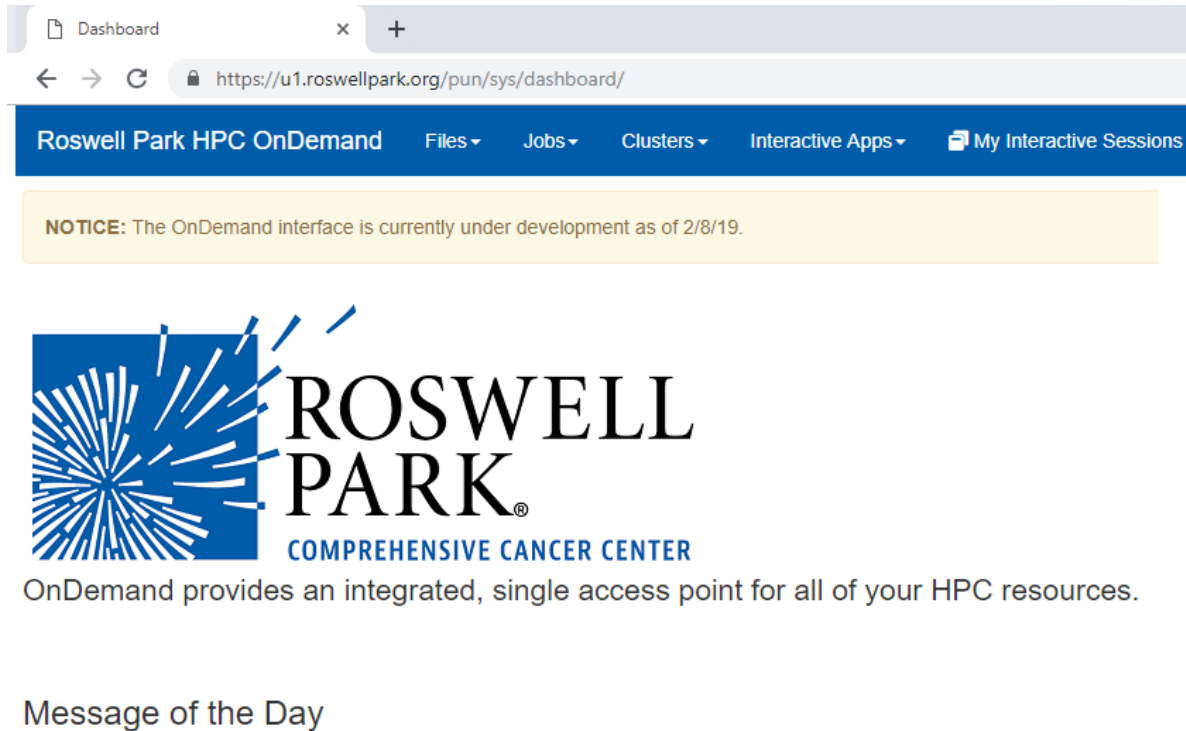
<http://u1.roswellpark.org/pun/sys/dashboard/>



- Note: *your OnDemand user ID and password may not be the same as your main Roswell ID and password*

# OnDemand Dashboard

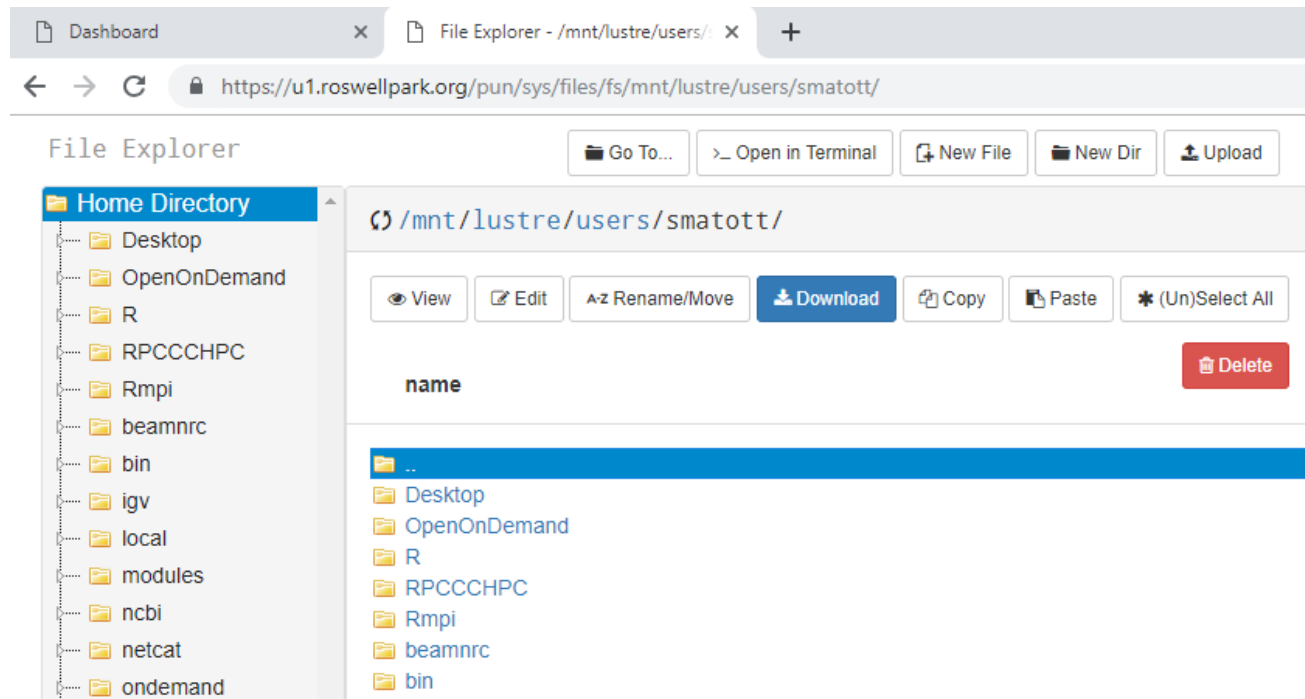
- Files
  - Upload, download, edit, move, delete etc.
- Jobs
  - Configure, submit, & monitor jobs
- Clusters
  - Linux CLI
- Interactive Apps
  - Launch programs with GUIs



The screenshot shows a web browser window with the title 'Dashboard' and a single tab. The address bar displays the URL <https://u1.roswellpark.org/pun/sys/dashboard/>. The dashboard header is a dark blue bar with the text 'Roswell Park HPC OnDemand' and navigation links for 'Files', 'Jobs', 'Clusters', 'Interactive Apps', and 'My Interactive Sessions'. Below the header is a yellow notice box stating: 'NOTICE: The OnDemand interface is currently under development as of 2/8/19.' The main content area features the Roswell Park Comprehensive Cancer Center logo, which consists of a blue square with white radiating lines and the text 'ROSWELL PARK' in large black letters, with 'COMPREHENSIVE CANCER CENTER' in smaller blue letters below it. Under the logo, a text line reads: 'OnDemand provides an integrated, single access point for all of your HPC resources.' At the bottom of the dashboard, the text 'Message of the Day' is visible.

# OnDemand - Files

- File Explorer
  - Transfers via upload/download or drag and drop
- Editor
  - Language aware coloring (e.g. Python comments, etc.)



# OnDemand – Compute Jobs

- Composer
  - Create and configure a new job
  - Use templates to minimize effort
  - Monitor or alter current jobs
- Active Jobs
  - Check job queues

The screenshot displays the 'Job Composer' interface for Roswell Park HPC OnDemand. The browser address bar shows the URL <https://u1.roswellpark.org/pun/sys/myjobs>. The navigation bar includes 'Roswell Park HPC OnDemand', 'Job Composer', 'Jobs', and 'Templates'. The main heading is 'Jobs'. Below this, there are buttons for '+ New Job', 'Edit Files', 'Job Options', 'Open Terminal', 'Submit', 'Stop', and 'Delete'. A search bar is also present. The job list table shows three completed jobs:

Created	Name	ID	Cluster	Status
April 25, 2019 10:47am	BiocParallel_multicore	736401	Roswell HPC Cluster	Completed
March 20, 2019 1:36pm	hello	728989	Roswell HPC Cluster	Completed
February 22, 2019 11:20am	Keras_TensorFlow_CIFAR10	725194	Roswell HPC Cluster	Completed



# OnDemand – Job Templates

- Templates
  - Searchable
  - IT RC (i.e. Shawn) can create new templates on request
  - Users can also create and manage their own templates

The screenshot shows a web browser window with two tabs: 'Dashboard' and 'Job Composer'. The address bar shows the URL <https://u1.roswellpark.org/pun/sys/myjobs/workflows/new>. The page title is 'Templates'. Below the title, a subtitle reads: 'To create a new job, select a template to copy, fill out the form to the right, and click "Create New Job".'

At the top of the main content area, there are two buttons: '+ New Template' and 'Copy Template'.

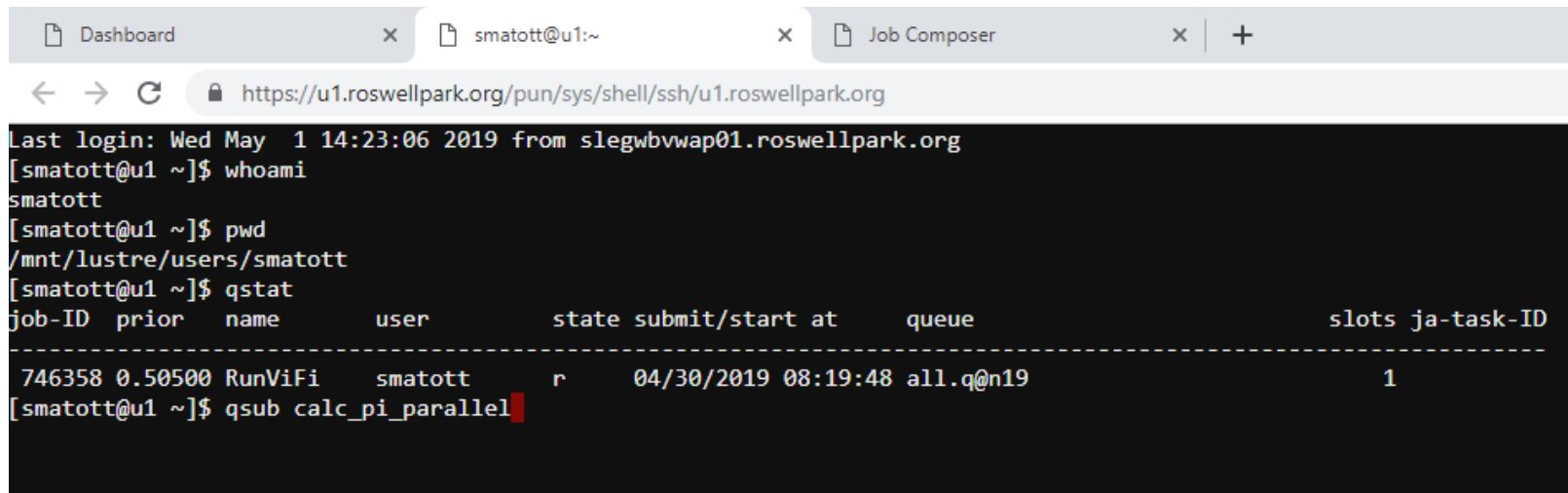
Below these buttons is a section for managing templates. It includes a 'Show 10 entries' dropdown, a 'Search:' input field, and a 'Delete' button. A table lists the available templates:

Name	Cluster	Source
BlocParallel_multicore	Hpc	System Templates
BlocParallel_Rmpi	Hpc	System Templates
BlocParallel_snow	Hpc	System Templates
hello	Hpc	System Templates
hello_array	Hpc	System Templates
Keras_TensorFlow_charseq	Hpc	System Templates

To the right of the table is a form titled 'Create New "BlocParallel\_multicore"'. The form includes a description: 'This illustrates using BlocParallel to parallelize a set of "busy" calculations across processors on a single node.' It has three sections: 'Job Name:' with a text input field containing 'BlocParallel\_multicore'; 'Cluster:' with a dropdown menu showing 'Roswell HPC Cluster'; and 'Script Name:' with a text input field containing 'sge-bloc-parallel-multicore'. At the bottom of the form are two buttons: 'Create New Job' and 'Reset'.

# OnDemand – Cluster CLI

- Provides a bash command line interface on front-end node (u1)
- Handy for those who are familiar with conventional HPC access paradigm (e.g. putty, ssh)



```
Last login: Wed May 1 14:23:06 2019 from slegwbvwap01.roswellpark.org
[smatott@u1 ~]$ whoami
smatott
[smatott@u1 ~]$ pwd
/mnt/lustre/users/smatott
[smatott@u1 ~]$ qstat
job-ID prior  name      user      state submit/start at    queue      slots ja-task-ID
-----
746358 0.50500 RunViFi    smatott   r       04/30/2019 08:19:48 all.q@n19      1
[smatott@u1 ~]$ qsub calc_pi_parallel
```

# OnDemand – Interactive Apps

- Special compute jobs for launching interactive GUI-based applications
- Remote Desktop + Individual Apps
- OpenGL codes can benefit from GPU hardware acceleration (viz nodes)

The screenshot shows the Roswell Park HPC OnDemand web interface. The browser address bar displays the URL: [https://u1.roswellpark.org/pun/sys/dashboard/batch\\_connect/sys/igv/session\\_contexts/new](https://u1.roswellpark.org/pun/sys/dashboard/batch_connect/sys/igv/session_contexts/new). The navigation bar includes links for Files, Jobs, Clusters, Interactive Apps, and My Interactive Sessions. A yellow notice states: "NOTICE: The OnDemand interface is currently under development as of 2/8/19." The breadcrumb trail shows Home / My Interactive Sessions / IGV. On the left, a sidebar lists Interactive Apps (Desktops, Remote Desktop, Programs, Grace, IGV, Jupyter, ParaView, RGui, RStudio) and Interactive Apps [Sandbox] (xclock). The main content area for IGV includes a description: "This app will launch IGV - the Integrative Genomics Viewer." It features a form for IGV Version (2.5.0), an Account field, a Queue field (all.q), a Number of hours field (1), and a Number of cores field (1). There is a checkbox for "I would like to receive an email when the session starts" and a blue Launch button at the bottom right.

Use the app window at left to select an app and submit request via “Launch” button.

# OnDemand – Interactive Sessions

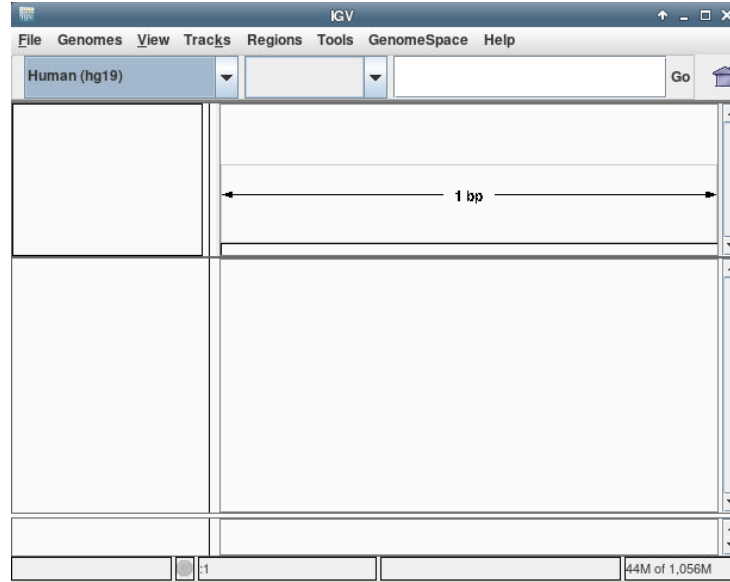
- Interactive apps are compute jobs!
- Can close browser window and reconnect without loss of work.

The screenshot displays the Roswell Park HPC OnDemand web interface. The browser address bar shows the URL [https://u1.roswellpark.org/pun/sys/dashboard/batch\\_connect/sessions](https://u1.roswellpark.org/pun/sys/dashboard/batch_connect/sessions). The navigation bar includes links for Files, Jobs, Clusters, Interactive Apps, and My Interactive Sessions. A yellow notice banner states: "NOTICE: The OnDemand interface is currently under development as of 2/8/19." Below this, a green message box confirms: "Session was successfully created." The main content area shows a sidebar with categories: Interactive Apps, Desktops, and Programs. Under Interactive Apps, the IGV application is listed. The main panel displays details for the IGV session (ID: 746925), which is running on 1 core. It includes the host name (n41), creation time (2019-05-02 15:37:15 EDT), and session ID (39c06ada-0c6e-41db-8cdb-7a661e9191a4). A "Launch IGV" button is visible, along with a "Delete" button and a "View Only (Share-able Link)" button.

Once the job starts the window will refresh and offer a “Launch [app]” button. Click it to open the app in a browser window.

# OnDemand – Interactive Sessions

- Interactive apps are compute jobs!
- Can close browser window and reconnect without loss of work.



In this example, a remote desktop is opened and pre-populated with an instance of the IGV genome viewer.

# About Bioconductor

- A set of 1700+ R packages for high-throughput sequence analysis
  - RNA-seq, ChIP-seq, Annotation, etc.
- [www.Bioconductor.org](http://www.Bioconductor.org)
- Open source and extensively documented and supported
  - Vignettes, user-groups, mailing lists, etc.
- The Bioconductor project is led by Roswell's own Dr. Martin Morgan!

Selected *Bioconductor* packages for high-throughput sequence analysis.

Concept	Packages
Data representation	<i>IRanges</i> , <i>GenomicRanges</i> , <i>GenomicFeatures</i> , <i>Biostrings</i> , <i>BSgenome</i> , <i>girafe</i> .
Input / output	<i>ShortRead</i> (fastq), <i>Rsamtools</i> (bam), <i>rtracklayer</i> (gff, wig, bed), <i>VariantAnnotation</i> (vcf), <i>R453Plus1Toolbox</i> (454).
Annotation	<i>GenomicFeatures</i> , <i>ChIPpeakAnno</i> , <i>VariantAnnotation</i> .
Alignment	<i>Rsubread</i> , <i>Biostrings</i> .
Visualization	<i>ggbio</i> , <i>Gviz</i> .
Quality assessment	<i>qrqc</i> , <i>seqbias</i> , <i>ReQON</i> , <i>htSeqTools</i> , <i>TEQC</i> , <i>Rolexa</i> , <i>ShortRead</i> .
RNA-seq	<i>BitSeq</i> , <i>cqn</i> , <i>cummeRbund</i> , <i>DESeq</i> , <i>DEXSeq</i> , <i>EDASeq</i> , <i>edgeR</i> , <i>gage</i> , <i>goseq</i> , <i>iASeq</i> , <i>tweeDEseq</i> .
ChIP-seq, etc.	<i>BayesPeak</i> , <i>baySeq</i> , <i>ChIPpeakAnno</i> , <i>chipseq</i> , <i>ChIPseqR</i> , <i>ChIPsim</i> , <i>CSAR</i> , <i>DiffBind</i> , <i>MEDIPS</i> , <i>mosaics</i> , <i>NarrowPeaks</i> , <i>nucleR</i> , <i>PICS</i> , <i>PING</i> , <i>REDseq</i> , <i>Repitools</i> , <i>TSSi</i> .
Motifs	<i>BCRANK</i> , <i>cosmo</i> , <i>cosmoGUI</i> , <i>MotIV</i> , <i>seqLogo</i> , <i>rGADEM</i> .
3C, etc.	<i>HiTC</i> , <i>r3Cseq</i> .
Copy number	<i>cn.mops</i> , <i>CNAnorm</i> , <i>exomeCopy</i> , <i>segmentSeq</i> .
Microbiome	<i>phyloseq</i> , <i>DirichletMultinomial</i> , <i>clstutils</i> , <i>manta</i> , <i>mcaGUI</i> .
Work flows	<i>ArrayExpressHTS</i> , <i>Genominator</i> , <i>easyRNASeq</i> , <i>oneChannelGUI</i> , <i>rnaSeqMap</i> .
Database	<i>SRadb</i> .

# OnDemand – Biostrings example

- Dashboard → Jobs → Job Composer → Templates
- Search for “Biostrings”
- Select the “Biostrings-gc-content” template
  - Compute GC content of multiple samples
- Click “Create New Job”
  - Drops you into the “Jobs” interface
- Explore the job details area
- Click on the “.R” file to open in editor
  - contains the Bioconductor code

```
1 |
2 # load packages
3 suppressPackageStartupMessages({
4   library(Biostrings)
5 })
6
7 # extract arguments
8 args = commandArgs(trailingOnly=TRUE)
9 fname = args[1]
10
11 # read sequence
12 myseq = unlist(readDNAStringSet(fname))
13
14 # compute GC and display result
15 af = alphabetFrequency(myseq, baseOnly=TRUE, as.prob=TRUE)
16 gc = as.numeric(af[2] + af[3])
17 cat(sprintf("%s : %f\n", fname, gc))
18
19
```

# OnDemand – Biostrings example

- Open the sge-R-biostrings script in the editor
- Notice the coarse “file-level” parallelization technique
  - Applicable any pipeline that must operate on many independent files
  - Good practice tip: start small, build up from there
- Job Composer → Submit
- Wait for job to complete
- Open gc\_values.out

```
1 #!/bin/bash
2 ##
3 ## output and error streams
4 $$ -j y -o gc_value.out
5 ##
6 ## job name
7 $$ -N R-biostrings
8 ##
9 ## who to e-mail
10 $$ -m e -M your.user.name@RoswellPark.org
11 ##
12 ## set working directory
13 $$ -cwd
14 ##
15 ## request processors
16 $$ -pe slots 16
17 ##
18 ## use general (all) queue
19 $$ -q all.q
20 ##
21
22 module load R
23
24 # process files in parallel using background processes (&)
25 count=0
26 for sample in data/*.fasta; do
27     count=`expr $count + 1`
28     Rscript biostrings-example.R $sample &
29     # wait for processing to complete
30     if [ "$count" == "$NSLOTS" ]; then
31         wait
32         count=0
33     fi
34 done
35 wait
```



# OnDemand – Biostrings example

gc\_values.out

```
1 data/sample_11.fasta : 0.416994
2 data/sample_07.fasta : 0.416914
3 data/sample_02.fasta : 0.416384
4 data/sample_06.fasta : 0.416290
5 data/sample_08.fasta : 0.416822
6 data/sample_00.fasta : 0.416929
7 data/sample_05.fasta : 0.416967
8 data/sample_01.fasta : 0.416750
9 data/sample_12.fasta : 0.415907
10 data/sample_13.fasta : 0.416505
11 data/sample_10.fasta : 0.416936
12 data/sample_09.fasta : 0.416581
13 data/sample_14.fasta : 0.416234
14 data/sample_03.fasta : 0.417042
15 data/sample_04.fasta : 0.416873
16 data/sample_15.fasta : 0.416951
17 data/sample_17.fasta : 0.416478
18 data/sample_19.fasta : 0.416667
19 data/sample_20.fasta : 0.416304
20 data/sample_16.fasta : 0.417042
21 data/sample_22.fasta : 0.416966
22 data/sample_18.fasta : 0.417076
23 data/sample_25.fasta : 0.416499
24 data/sample_27.fasta : 0.416773
25 data/sample_21.fasta : 0.416749
```

```
26 data/sample_31.fasta : 0.416869
27 data/sample_23.fasta : 0.416551
28 data/sample_29.fasta : 0.416280
29 data/sample_26.fasta : 0.416916
30 data/sample_28.fasta : 0.416465
31 data/sample_24.fasta : 0.416965
32 data/sample_30.fasta : 0.417133
33 data/sample_40.fasta : 0.416440
34 data/sample_41.fasta : 0.417264
35 data/sample_33.fasta : 0.416794
36 data/sample_39.fasta : 0.416792
37 data/sample_47.fasta : 0.416272
38 data/sample_36.fasta : 0.416964
39 data/sample_34.fasta : 0.416772
40 data/sample_42.fasta : 0.416501
41 data/sample_35.fasta : 0.417028
42 data/sample_38.fasta : 0.417077
43 data/sample_37.fasta : 0.417142
44 data/sample_44.fasta : 0.416364
45 data/sample_43.fasta : 0.416558
46 data/sample_32.fasta : 0.417158
47 data/sample_45.fasta : 0.415817
48 data/sample_46.fasta : 0.416151
49 data/sample_48.fasta : 0.416726
50 data/sample_52.fasta : 0.416814
```

```
51 data/sample_50.fasta : 0.416599
52 data/sample_49.fasta : 0.416948
53 data/sample_51.fasta : 0.417097
54 data/sample_55.fasta : 0.417185
55 data/sample_53.fasta : 0.416749
56 data/sample_56.fasta : 0.416610
57 data/sample_57.fasta : 0.416914
58 data/sample_54.fasta : 0.416672
59 data/sample_60.fasta : 0.416513
60 data/sample_59.fasta : 0.417132
61 data/sample_58.fasta : 0.416621
62 data/sample_61.fasta : 0.416657
63 data/sample_62.fasta : 0.416432
64 data/sample_63.fasta : 0.416860
65 data/sample_69.fasta : 0.416245
66 data/sample_64.fasta : 0.417372
67 data/sample_65.fasta : 0.416539
68 data/sample_68.fasta : 0.416853
69 data/sample_71.fasta : 0.416137
70 data/sample_70.fasta : 0.416468
71 data/sample_66.fasta : 0.417003
72 data/sample_67.fasta : 0.416288
73 data/sample_73.fasta : 0.416393
74 data/sample_76.fasta : 0.416951
75 data/sample_79.fasta : 0.417166
```

# Questions?

- Contact IT RC through:
  - The ServiceNow portal (RequestIT, FixIT):  
[roswellpark.service-now.com](https://roswellpark.service-now.com)
  - e-mail Dr. Matott:  
[Loren.Matott@RoswellPark.org](mailto:Loren.Matott@RoswellPark.org)
  - The IT RC website:  
[roswellpark.org/research/departments/it-research-computing](https://roswellpark.org/research/departments/it-research-computing)