# BD Single Cell Genomics Analysis Setup User Guide

Doc ID: 47383 Rev. 7.0

23-21333-00
02/2019

**Becton, Dickinson and Company**
**BD Biosciences**
2350 Qume Drive
San Jose, CA 95131 USA
Tel 1.877.232.8995

bdbiosciences.com
ResearchApplications@bd.com

### Regulatory information

For Research Use Only. Not for use in diagnostic or therapeutic procedures.

### History

| Revision | Date | Change made |
|---|---|---|
| Doc ID: 47383 Rev. 1.0 | 09/2017 | Initial release. |
| Doc ID: 47383 Rev. 2.0 | 11/2017 | —Added setup information for multiplex runs.<br>—Rebranded document. |
| Doc ID: 47383 Rev. 3.0 | 12/2017 | —Updated BD™ Data View content to latest version v1.1.<br>—Moved note to ensure use of correct CWL files under Requirements.<br>—Updated Define App Settings in Seven Bridges Genomics and local installation chapters. |
| Doc ID: 47383 Rev. 4.0 | 04/2017 | Added chapter on a customer service. |
| Doc ID: 47383 Rev. 5.0 | 07/2018 | —Removed chapter on a customer service.<br>—Updated to BD™ Data View v1.2.<br>—Added content to set up for analysis of experiments with BD™ AbSeq Ab-Oligos. |

| Revision | Date | Change made |
|---|---|---|
| Doc ID: 47383 Rev. 6.0 | 10/2018 | —In the requirements for local installation, clarified that Microsoft® Windows® is not supported and specified that Python 2.7.15 or later is required.<br>—For CWL-runner on a local installation, added a recommendation of ≥32 GB memory limit.<br>—Clarified that local installation is supported by most Unix-like operating systems. |
| Doc ID: 47383 Rev. 7.0<br>23-21333-00 | 02/2019 | Added reference to the BD™ Mouse Immune Single-Cell Multiplexing Kit. |

# Contents

# 1

# Introduction

# About this guide

**Introduction**

This guide provides detailed instructions on how to set up and run the BD Rhapsody™ Analysis pipeline for sequencing and clustering analyses on the Seven Bridges Genomics platform or on a local installation. While sequencing analysis is required before clustering analysis, clustering analysis can be performed independently. Output from the analysis pipeline can be visualized using BD™ Genomics Data View, which is run locally.

For references, including third-party tools, see the *BD Single Cell Genomics Bioinformatics Handbook* (Doc ID: 54169).

Genomics technical publications are available for download from the BD Genomics Resource Library at bd.com/genomics-resources.

# 2

# Requirements

# Seven Bridges Genomics platform

**Introduction**

Create an account only if you will analyze sequencing data on the Seven Bridges Genomics platform.

**Seven Bridges Genomics account**

1. Go to sbgenomics.com/bdgenomics.
2. Click **Request Access**. In the request access window, enter your email address so that you can receive an email invitation to the Seven Bridges Genomics platform within 24 hours.
3. Click the link in the email invitation, and complete the registration. Seven Bridges Genomics displays the dashboard with the demo projects.

# Local installation

**Introduction**

The system that runs BD Rhapsody™ analyses must meet certain minimum requirements. See Minimum system requirements.

The software applications required for analysis have specific software tools. To ensure that these tools are always available, the analysis is run in a self-contained environment called a docker container. The docker container is obtained by "pulling" or downloading a docker image to your local computer. The docker container has all of the libraries and settings required by the pipeline to run the analysis. In the portable docker container, the analysis can be run reproducibly wherever it is deployed, whether on a local installation or the Seven Bridges Genomics platform. CWL-runner is the tool that manages docker containers to complete the pipeline run. CWL-runner uses two inputs: a CWL workflow file and a YML input specification file. The CWL workflow file describes each step in the pipeline and how each docker container should run to complete the step. The YML file tells CWL-runner where to find the pipeline inputs, such as the sequencer read files and gene panel reference. When the pipeline run is finished, CWL-runner obtains the final outputs in the docker containers and adds them to a designated output folder on your computer.

**Minimum system requirements**

- Operating system: macOS® X or Linux®. Microsoft® Windows® is not supported.
- 8-core processor (>16-core recommended)
- 32 GB RAM (128 GB recommended)
- 250 GB free disk space

## Software requirements

### Docker

Install the community edition at store.docker.com.

### Python 2.7.15 or later

1. Check to see if Python 2.7.15 or later is already installed by running at the command line:

   ```
   $ python2 --version
   ```

2. Ensure that you are using a local installation of Python and not a system version. Run:

   ```
   $ which python
   ```

   This should return the path to a local installation and not to a system path (usually /usr/bin/python).

   **Using a system installation of python might not give you sufficient permissions to install the required packages.**

3. If Python 2.7.15 or later is not installed, download and install it from python.org/downloads.

4. If pip is not installed, go to pip.pypa.io/en/stable/installing, and follow the instructions.

5. Update pip before installing cwlref-runner by using the command:

   ```
   $ pip install -U pip
   ```

**CWL-runner**

1. Install the package from PyPi. Enter:

   ```
   $ pip install cwlref-runner
   ```

2. Ensure that cwl-runner is in your path. Type:

   ```
   $ cwl-runner
   ```

3. If the command is not found, add the install location of the pip packages to $PATH.

   a. Find where cwlref-runner is installed by entering:

      ```
      $ pip show cwlref-runner
      ```

   b. Add the above path to $PATH. For example:

      ```
      $ export PATH=$PATH:/Library/Frameworks/
      Python.framework/Versions/2.7/lib/python2.7
      ```

   c. Restart the command line utility.

---

**CWL and YML files**

**Ensure that you are using the correct CWL files with your pipeline, or the analysis might fail. To determine your pipeline version, see Pipeline image (page 14).**

1. If necessary, create a Bitbucket account. Go to bitbucket.org/CRSwDev/cwl.
2. In the left pane, click **Downloads > Download Repository**. The CWL and YML files are downloaded.
3. Unzip the archive. Each folder within the archive is named after the pipeline version it corresponds to.

---

**Pipeline image**

1. Ensure that docker is running.
2. Download (pull) the docker image by entering:

   ```
   $ docker pull bdgenomics/rhapsody
   ```

   **Note:** The pull command automatically downloads the most current pipeline version. To download an earlier version, specify the version number. For example:

   ```
   $ docker pull bdgenomics/rhapsody:v1.0
   ```

3. Confirm the pipeline image by entering:

   ```
   $ docker images
   ```

   **Note:**

   – bdgenomics/rhapsody appears under the repository column.

   – The pipeline version number appears under the tag column.

# FASTQ files

**Dataset size**

**BD Biosciences recommends analyzing datasets that are ≤1 TB in size. For datasets (compressed FASTQ FILES from all libraries) >1 TB, contact BD Biosciences technical support at researchapplications@bd.com.**

**Read 1 and Read 2 sequencing files**

For the Seven Bridges Genomics platform and local installation, obtain Read 1 and Read 2 sequencing files, and ensure that the FASTQ file names follow these rules:

- An underscore on each side of R1 or R2 (_R1_ and _R2_).
- The <sample> name should be the same for R1 and R2.
- Convert uncompressed files to .gz format.

  **Example**
  <sample>_S1_L001_R1_001.fastq.gz

  <sample>_S1_L001_R2_001.fastq.gz

**Do not use special characters or spaces in the filenames, or the analysis might fail. Use only letters, numbers, or hyphens.**

**Note:** If you are downloading the files from BaseSpace, follow these steps:

a. Choose the run to download in BaseSpace.

b. Click the download icon on the main screen.

c. If necessary, install the BaseSpace downloading application.

d. Click **Select all fastq files for this run**.

e. Download the files. This might take several minutes.

For more information, go to help.basespace.illumina.com.

# FASTA files

**Introduction**

Separate FASTA reference files are used to store the sequences of gene targets and BD™ AbSeq Ab-Oligos (antibody-oligonucleotides) that are used in a BD Rhapsody experiment.

**Obtaining pre-designed mRNA panels**

Obtain the FASTA panels from the Seven Bridges demo project or by contacting BD Biosciences customer support at researchapplications@bd.com.

**Designing supplemental or custom mRNA panels**

By providing a list of genes to BD Biosciences customer support, we can design custom mRNA targeted panels. Contact BD Biosciences customer support at researchapplications@bd.com.

**Downloading, preparing, and saving an AbSeq reference file**

If your experiment contains BD™ AbSeq Ab-Oligos, you are required to have an AbSeq reference file.

1. Download the FASTA file containing all of the BD Ab-Oligo (AbO) sequence. Go to bd-rhapsody-public.s3-website-us-east-1.amazonaws.com/AbSeq-references/BDAbSeq_allReference_latest.fasta.
2. Use a text editor such as Microsoft® Notepad or TextEdit to delete the sequence header and sequence pairs that will not be used in the experiment.

   **Do not use a word processor such as Microsoft® Word, which can add unintended special characters to the file.**

3. Ensure that the AbSeq reference file follows these rules:
   - File extension is .fa or .fasta
   - Format is:

```
>CD103|ITGAE|AHS0001|pAbO
AAATAGTATCGAGCGTAGTTAAGTTGCGTAGCCGTT
>CD161|KLRB1|AHS0002|pAbO
GTTATGGTTGTCGGTAGAGTATCGTGTTGCGTTAGT
```

   **Note:** BD Biosciences uses this format for its sequence header: <AntibodyName>|<GeneSymbol>|<SeqID>|pAbO.

4. Save as an .fa or .fasta file locally on your computer.

This page intentionally left blank

# 3

# Setting up sequencing analysis on the Seven Bridges Genomics platform

# Introduction

Whether analysis is performed on the Seven Bridges Genomics platform or locally, sequencing and clustering analyses use the BD Rhapsody™ Analysis pipeline. During the execution of the pipeline, sequencing analysis processes sequencing files to generate molecular counts per cell, read counts per cell, metrics, and an alignment file. Clustering analysis is based on single cell gene expression profiles. Sequencing analysis is required before clustering analysis, but clustering analysis can be run independently of sequencing analysis. See Setting up clustering analysis on the Seven Bridges Genomics platform (page 31) or Setting up clustering analysis on a local installation (page 45).

# Workflow

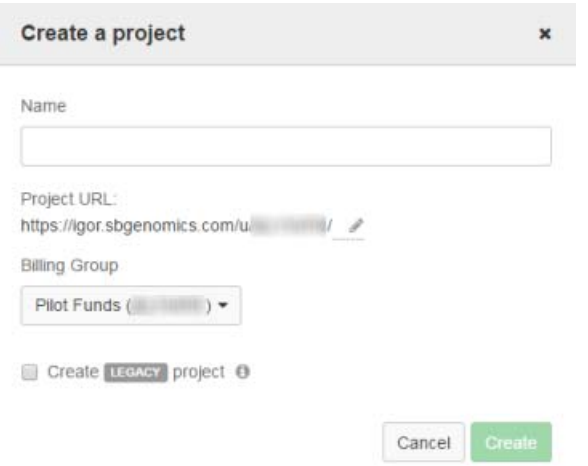
**During sequencing analysis, the BD Rhapsody™ Analysis pipeline analyzes only one cartridge per run. To analyze multiple cartridges, create a pipeline run (or task) for each cartridge. During clustering analysis, multiple cartridges can be merged and analyzed together.**

| Step | Purpose |
|---|---|
| 1 | Create a new project. |
| 2 | Import FASTQ files. |
| 3 | Import the reference file. |
| 4 | Import the BD Rhapsody Analysis pipeline. |
| 5 | Set up and run the pipeline. |
| 6 | Download the output files. |

# Creating a new project

**Procedure**

1. At the top of the dashboard, click **Projects > Create a project:**




2. On the Create a project dialog, enter the project name, and edit the project URL if necessary.

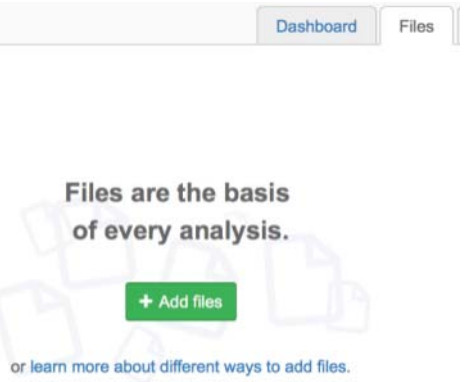
   **Do not click the checkbox by Create Legacy project.**

3. Click **Create**. Seven Bridges Genomics displays the new project dashboard.

# Importing FASTQ files

**Procedure**

1. On the project dashboard, click the **Files** tab, and then click **+Add files**:




2. In the top menu, select the source of the files, such as **Public files**, **Projects**, or **FTP/HTTP**. Seven Bridges Genomics displays instructions on uploading the files. Follow the Seven Bridges Genomics instructions to import your files.

   **Use the Desktop Uploader to upload files from BaseSpace. Security permissions on your BaseSpace account prevent FTP/HTTP protocols from working.**

3. After import, the files are on the Files tab.

# Importing reference files

**Importing pre-designed mRNA panels**

1. On the **Files** tab of the project dashboard, click **+Add files**.
2. Click **Projects**, and then click on **BD Rhapsody Analysis pipeline** in the left panel.
3. Locate the appropriate FASTA file for your experiment, and click **Copy**.

**Importing supplemental or custom mRNA panels or AbSeq reference files**

1. On the project dashboard, click the **Files** tab, and then click **+Add files**.
2. In the top menu, select the source of the files, such as **Public files**, **Projects**, or **FTP/HTTP**. Seven Bridges Genomics displays instructions on uploading the files. Follow the Seven Bridges Genomics instructions to import your files.

   **Use the Desktop Uploader to upload files from BaseSpace. Security permissions on your BaseSpace account prevent FTP/HTTP protocols from working.**
3. After import, the files are on the Files tab.

# Importing the BD Rhapsody Analysis pipeline

**Importing the pipeline**

1. On the project dashboard, click the **Apps** tab, and then click **+Add app**.
2. Click **Public Apps**, and then enter **Rhapsody** to find the **BD Rhapsody Analysis pipeline**. Or, copy the workflow from the Demo project.
3. Click **Copy** on the app window, select the project in the drop-down menu, and then click **Copy** again.
4. Navigate to the Apps tab to confirm that the workflow was copied to the project.

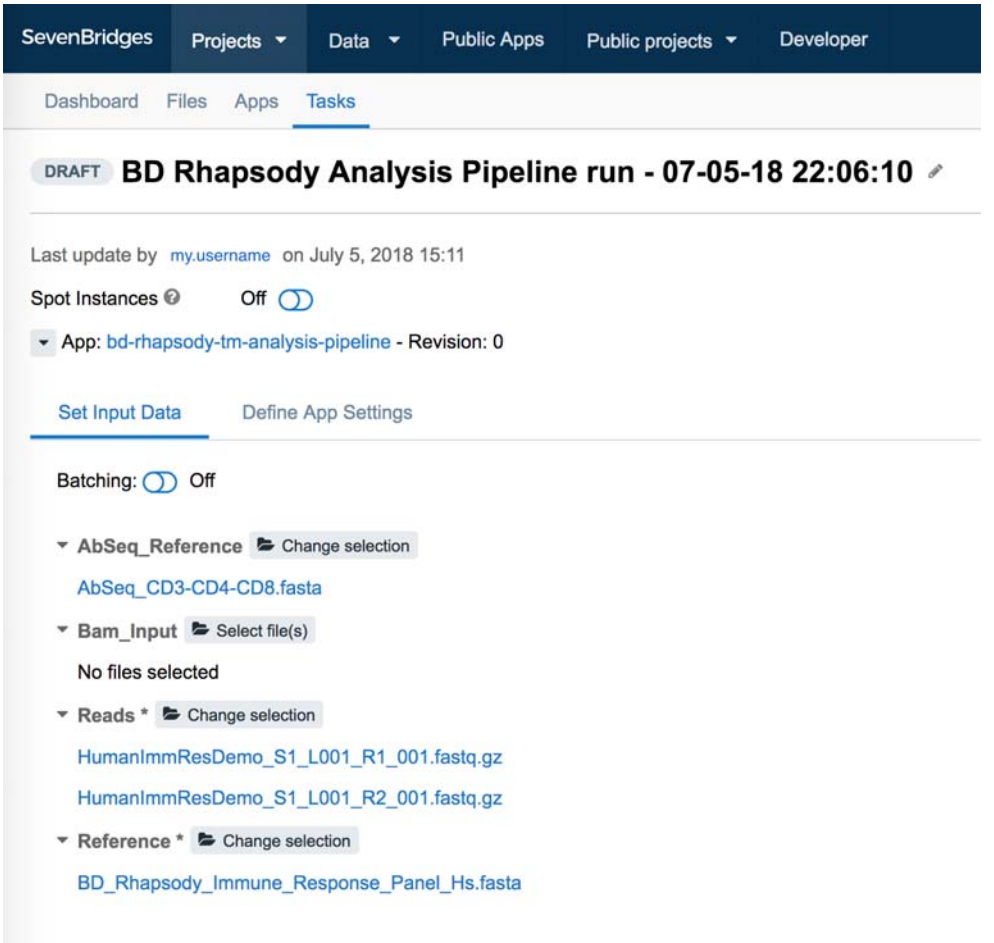
# Setting up and running the pipeline

**Procedure**

1. Click the **Apps** tab to view the apps.

   **Note:** If the app is highlighted in yellow, an update is available. Select the refresh icon to get the latest app version.

2. By the **BD Rhapsody Analysis pipeline**, click the green play button under Actions. The Set Input Data tab displays on the Tasks tab:



Complete all required fields, which appear in red.

| Input field | Input | Required? |
|---|---|---|
| AbSeq _Reference | FASTA AbSeq reference file generated from Importing supplemental or custom mRNA panels or AbSeq reference files (page 23).<br>**Ensure that the AbSeq reference file contains the BD AbSeq Ab-Oligos that were used in the experiment; otherwise, the read mapping will be incorrect.** | Optional |
| BAM_Input[a] | The BAM file output from a previous analysis of the same library as the FASTQ files. | Optional |
| Reads | R1 reads and R2 reads. Ensure to include all FASTQ sequencing data from the experiment, including R1 and R2 files for the targeted RNA library, and, if applicable, the Sample Tag and BD™ AbSeq libraries. | Yes |
| Reference | This is an mRNA reference file. Select the FASTA reference file. This is a pre-designed, supplemental, or custom panel.<br>**Ensure that the reference matches the species and panel used for the experiment; otherwise, read mapping will not be correctly aligned.** | Yes |
| Sample_Tags _Version | For a multiplexed samples run only. Specifies the Sample Tags used:<br>Single-Cell Multiplex Kit—Human<br>Single-Cell Multiplex Kit—Mouse | Required for multiplexed samples |
| Subsample_ _Sample _Tags | For a multiplexed samples run only. Any number of reads >1 or a fraction of reads between 0<n<1 to indicate the percentage of reads to subsample per Sample Tag. | Optional for multiplexed samples |

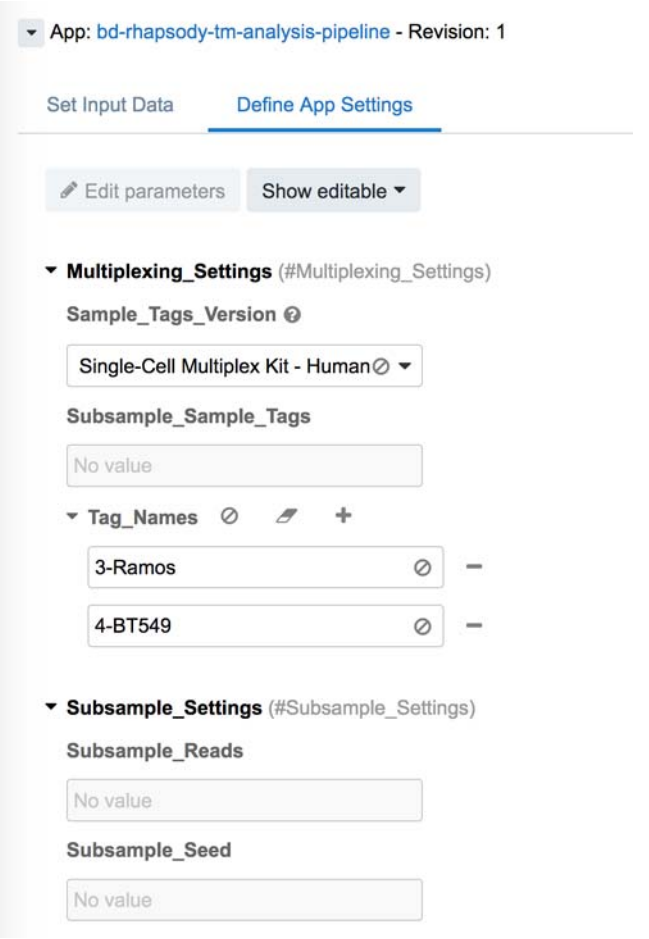
| Input field (continued) | Input | Required? |
|---|---|---|
| Tag_Names | For a multiplexed samples run only. To enter a new sample, click **+** to add a row. Enter one tag name per row by following this format, using a hyphen; **no spaces or forward slashes allowed**:<br>**Sample Tag number-sample name**<br>Example: 3-Ramos<br><br>**Note:** Until the tag name is in the correct format, a red *expected type* warning message is displayed. | Optional for multiplexed samples |
| Subsample _Reads | Any number of reads >1 or a fraction between 0< n<1 to indicate the percentage of reads to subsample. | Optional |
| Subsample _Seed | For use when replicating a previous subsampling run only. Obtain the seed generated from the log file for the SplitFastQ node. To obtain the log file, see Downloading the log file from Seven Bridges Genomics (page 68). **Entering the seed ensures that the same reads are subsampled to reproduce the results. If no seed is needed, leave blank and the pipeline will generate one randomly.** | Optional |

a. Use BAM output from a previous analysis and a FASTQ file from the current sequencing run to achieve greater sequencing depth.

3. On the Set Input Data tab, import your files for analysis according to these requirements:
   - For every R1 .fastq.gz file, import the paired R2 .fastq.gz file.
   - Multiple R1 and R2 reads can be run together as long as they are from the same library, but the files can be generated from different sequencer runs.

4. If necessary, set the options on the Define App Settings tab. For example:

   **When using a BD™ Single-Cell Multiplexing Kit, be sure to select the Sample_Tags_Version (Single-Cell Multiplex Kit - Human or Mouse) from the drop-down menu.**



5. Click **Run**. Seven Bridges Genomics displays the app running on the Tasks tab.
6. If you enabled email notifications, look for notification of the completed run.

# Downloading the output

**Procedure**

See Downloading output files on the Seven Bridges Genomics platform (page 54).

This page intentionally left blank

# 4

# Setting up clustering analysis on the Seven Bridges Genomics platform

# Workflow

## Introduction

The BD Rhapsody™ Clustering Analysis app clusters gene expression profiles of cells and is part of the BD Rhapsody™ Analysis pipeline. While sequencing analysis is required before clustering analysis, clustering analysis can be performed independently. Standalone clustering analysis is particularly useful for analysis across multiple cartridges.

## Workflow

| Step | Purpose |
|---|---|
| 1 | Obtain the required files. |
| 2 | Import the required files. |
| 3 | Import the app. |
| 4 | Set up and run the app. |
| 5 | Download the output files. |

# Obtaining the required files

**Procedure**

The required file for clustering analysis is DBEC_MolsPerCell.csv or Expression_Data.st. Multiple CSV or ST files can be used but only of one file type. Each file is generated from sequencing analysis with the BD Rhapsody Analysis pipelines. See Sequencing analysis output files (page 55).

# Importing the required file

**Procedure**

Import the required DBEC_MolsPerCell.csv or Expression_Data.st file:

- Using an existing project: The required file is already available as an output file to select for running the clustering analysis.
- Creating a new project: Click the **Files** tab, and then click **+Add files**. Click the project containing the sequencing analysis. Select the DBEC_MolsPerCell.csv or Expression_Data.st file, and then copy to the new project.

# Importing the app

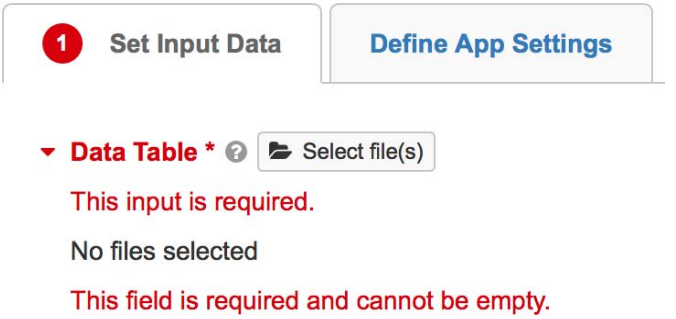
**Procedure**

1. On the project dashboard, click the **Apps** tab, and then click **+Add app**.
2. Click **Browse Public Apps**, and then enter **Rhapsody** to find the BD Rhapsody Clustering Analysis app. Or, copy the app from the Demo project.
3. Click **Copy** on the app window, select the project in the drop-down menu, and then click **Copy** again.
4. Navigate to the Apps tab to confirm that the app was copied to the project.

# Setting up and running the app

**Procedure**

1. Click the **Apps** tab to view the apps. If the app is highlighted in yellow, an update is available. Select the refresh icon to get the latest app version.
2. By the BD Rhapsody Clustering Analysis app, click the green play button under Actions. The Set Input Data tab displays on the Tasks tab (asterisk means required input):




3. On the Set Input Data tab, input your file(s) for analysis. Fields in red are required. You can select one or more data table files to run in the same analysis.
4. Skip the Define App Settings tab.

5. Click **Run**. Seven Bridges Genomics displays the app running on the Tasks tab.
6. If you enabled email notifications, look for notification of the completed run.

# Downloading the output

**Procedure**

See Downloading output files on the Seven Bridges Genomics platform (page 54).

# 5

# Setting up sequencing analysis on a local installation

# Workflow

**During sequencing analysis, the BD Rhapsody™ Analysis pipeline analyzes only one cartridge per run. To analyze multiple cartridges, create a pipeline run (or task) for each cartridge. During clustering analysis, multiple cartridges can be merged and analyzed together.**

| Step | Purpose |
|---|---|
| 1 | Set up the input specification file. |
| 2 | Run the pipeline using CWL-runner at the command line. |

# Setting up the input specification file

**Procedure**

The input specification file template.yml is downloaded from the CWL folder.

1. Obtain the FASTQ files. See Read 1 and Read 2 sequencing files (page 15).
2. Obtain the mRNA reference file from BD Biosciences technical support at **researchapplications@bd.com**.
3. If your experiment contains BD™ AbSeq Ab-Oligos, obtain the AbSeq Reference file. See Downloading, preparing, and saving an AbSeq reference file (page 16).
4. Specify the desired file paths in the YML file for Reads and Reference with the exact input field listed in the table. (Optional) Define BAM input, subsample, and subsample seed input fields:

| Input field | Input | Required? |
|---|---|---|
| Reads | R1 reads and R2 reads. Ensure to include all FASTQ sequencing data from the experiment, including R1 and R2 files for the targeted RNA library, and, if applicable, the Sample Tag and BD™ AbSeq libraries. | Yes |
| Reference | Select the FASTA reference file. This is a pre-designed, supplemental, or custom panel. | Yes |
| AbSeq Reference | FASTA AbSeq reference file generated from Importing supplemental or custom mRNA panels or AbSeq reference files (page 23).<br>**Ensure that the AbSeq reference file contains the BD™ AbSeq Ab-Oligos that were used in the experiment; otherwise, the read mapping will be incorrect.** | Optional |
| BAM_Input[a, b] | The BAM file output from a previous analysis of the same library as the FASTQ files. | Optional |

| Input field (continued) | Input | Required? |
|---|---|---|
| Subsample[a] | Any number of reads >1 or a fraction between 0< n<1 to indicate the percentage of reads to subsample. | Optional |
| Subsample_seed[a] | For use when replicating a previous subsampling run only. Obtain the seed generated from the log file for the SplitFastQ node. To obtain the log file, see Downloading the log file from Seven Bridges Genomics (page 68).<br>**Entering the seed ensures that the same reads are subsampled to reproduce the results. If no seed is needed, leave blank, and the pipeline will generate one randomly.** | Optional |
| Sample_Tags _Version[a] | For a multiplexed samples run only. Specifies the Sample Tags used: human (hs), mouse (mm). | Required for multiplexed samples |
| Subsample_Tags[a] | For a multiplexed samples run only. Any number of reads >1 or a fraction of reads between 0<n<1 to indicate the percentage of reads to subsample per Sample Tag. | Optional for multiplexed samples |
| Tag_Names[a] | For a multiplexed samples run only. Associate a name with each Sample Tag, which will appear in the output files. Within square brackets, enter a comma-separated list of Sample Tag numbers and associated names. For each sample, follow this format, using a hyphen; **no spaces or forward slashes allowed**:<br>**Sample Tag number-sample name**<br>Example: Tag_Names: [3-Ramos, 4-BT549] | Optional for multiplexed samples |

a. If BAM input, subsampling, or multiplex options are not needed, the corresponding fields can be deleted from the YML file.

b. Use BAM output from a previous analysis and a FASTQ file from the current sequencing run to achieve greater sequencing depth.

5. If necessary, specify multiple R1 and R2 reads under **Reads** by including additional file objects and following the nomenclature for each file. For example:

```
-class: File
location: "path/to/additional_R1_fastq.gz"
```

For example:

**YML file example showing a pair of FASTQ files and a panel reference file as input**

```
#!/usr/bin/env cwl-runner
cwl:tool: Rhapsody

Reads:
 - class: File
   location: path/to/mySample_R1_.fastq.gz
 - class: File
   location: path/to/mySample_R2_.fastq.gz

Reference:
 - class: File
   location: path/to/reference.fasta
AbSeq_Reference:
 - class: File
   location: path/to/abseq_reference.fasta
```

**YML file example showing optional BAM input and 50% subsampling of the reads**

```
#!/usr/bin/env cwl-runner
cwl:tool: Rhapsody

Reads:
 - class: File
   location: "test/mySample2_R2_.fastq.gz"
 - class: File
   location: "test/mySample2_R1_.fastq.gz"


Reference:
 - class: File
   location: "test/Immune_Response_Panel_Hs_with_Phix.fasta"


Subsample: 0.5

Bam_Input:
 - class: File
   location: "test/mySample1.final.BAM"
```

**YML file example showing choice of human Sample Tags, 50% subsampling of reads per Sample Tag, and Sample Tag naming**

```
#!/usr/bin/env cwl-runner
cwl:tool: mist

Reads:
 - class: File
   location: /path/to/mySample_R1_.fastq.gz
 - class: File
   location: /path/to/mySample_R2_.fastq.gz
 - class: File
   location: /path/to/mySampleTag_R1_.fastq.gz
 - class: File
   location: /path/to/mySampleTag_R2_.fastq.gz

Reference:
 - class: File
   location: /path/to/targeted_sampleTags.fasta

Sample_Tags_Version: human

Subsample_Tags: 0.5

Tag_Names: [4-mySample, 9-myOtherSample, 6-alsoThisSample]
```

6. Save the modified template YML file.

# Running the pipeline

**Procedure**

See Running a pipeline using CWL-runner (page 49).

This page intentionally left blank

# 6

# Setting up clustering analysis on a local installation

# Workflow

**Introduction**

The BD Rhapsody™ Clustering Analysis app clusters gene expression profiles of cells and is part of the BD Rhapsody™ Analysis pipeline. While sequencing analysis is required before clustering analysis, clustering analysis can be performed independently. Standalone clustering analysis is particularly useful for analysis across multiple cartridges.

**Workflow**

| Step | Purpose |
|---|---|
| 1 | Obtain the required files. |
| 2 | Set up the input specification file. |
| 3 | Run the pipeline using the CWL-runner at the command line. |

# Obtaining the required files

**Procedure**

Use either DBEC_MolsPerCell.csv or Expression_Data.st for clustering analysis. Multiple CSV or ST files can be used but only of one file type. The files are generated from sequencing analysis with the BD Rhapsody analysis pipelines. See Sequencing analysis output files (page 55).

# Setting up the input specification file

**Procedure**

Modify the ClusteringAnalysis-template.yml with your desired input files path. You can specify one or more data tables. For example:

**YML file example showing two samples being analyzed together**

```
#!/usr/bin/env cwl-runner
cwl:tool: ClusteringAnalysis

DataTable:
 - class: File
   location: "data/mySample1_DBEC_MolsPerCell.csv"
 - class: File
   location: "data/mySample2_DBEC_MolsPerCell.csv"
```

# Running the pipeline

**Procedure**

See Running a pipeline using CWL-runner (page 49).

This page intentionally left blank

# 7

# Running a pipeline using CWL-runner

# Running CWL-runner on a local installation

**Procedure**

Local installation is supported by most Unix-like operating systems such as macOS X or Linux. Minimum system requirements must be met. See Local installation (page 11).

To run the pipeline on macOS X, perform these additional configuration steps:

1. To enable CWL-runner to set up volumes, run the command:

   ```
   $ export TMPDIR=/tmp/docker_tmp
   ```

2. To increase the memory available to docker:

   a. Click the docker icon in the menu bar to open the docker menu.

   b. Click **Preferences**, and navigate to the Advanced tab.

   c. Use the slider to increase the memory limit. BD Biosciences recommends ≥32 GB. Lower limits are sufficient for smaller datasets.

   d. Click **Apply & Restart** at the bottom of the window.

**Running CWL-runner**

1. In the terminal, ensure that you are in a directory that contains the CWL files that were downloaded from the Bitbucket repository. The edited YML file for input specifications must also be present in this directory. See Setting up sequencing analysis on a local installation (page 37) or Setting up clustering analysis on a local installation (page 45).

2. Run the pipeline by entering the command:

   ```
   $ cwl-runner workflow.cwl input.yml
   ```

   If running the sequencing analysis pipeline, the workflow is the file rhapsody.cwl, and the input specification file is the edited template.yml.

   If running the clustering analysis pipeline, the workflow is the file ClusteringAnalysis.cwl, and the input specification file is the edited ClusteringAnalysis-template.yml.

3. If desired, you can specify the output directory for the analysis using the flag `--outdir`

   An example command:

   ```
   $ cwl-runner --outdir
   /path/to/results_folder rhapsody.cwl my_sample.yml
   ```

   **Note:** The output directory must be an existing directory. If no output directory is specified, files are output to the working directory.

4. Confirm that the following message displays after the pipeline is completed:

   ```
   Final process status is success.
   ```

5. Access the output files. All output files are found in the output directory specified in the CWL-runner command. If no output directory is specified, the files are output to the directory from which the command was called. See Reviewing output files (page 53).

This page intentionally left blank

# 8

# Reviewing output files

# Downloading output files on the Seven Bridges Genomics platform

**Procedure**

1. Select the project from the Projects drop-down menu to view output files.
2. Click the **Tasks** tab to view the list of tasks.
3. Click the name of the completed task to view Outputs on the right of the screen.
4. Click **Download** to download and save the output file. To download all files at once, click the **Files** tab, click the checkboxes by the files to download, and then click **Download**.
5. View the output files. See Sequencing analysis output files (page 55) and Clustering analysis output files (page 58).

# Sequencing analysis output files

Most output files contain a header summarizing the pipeline run. Headers contain all of the information needed to rerun the pipeline with the same settings.

| Output | File | Content |
|---|---|---|
| Metrics summary | <sample_name>_Metrics_Summary.csv | Report containing sequencing, molecules, and cell metrics |
| BAM | <sample_name>.final.BAM | Alignment file of R2 and associated R1 annotations |
| Data tables[a] | <sample_name>_RSEC_MolsPerCell.csv<br><sample_name>_RSEC_ReadsPerCell.csv<br><sample_name>_DBEC_MolsPerCell.csv<br><sample_name>_DBEC_ReadsPerCell.csv | Reads per gene per cell and molecules per gene per cell, based on RSEC or DBEC |
| | <sample_name>_RSEC_MolsPerCell_Unfiltered.csv.gz<br><sample_name>_RSEC_ReadsPerCell_Unfiltered.csv.gz<br><sample_name>_DBEC_MolsPerCell_Unfiltered.csv.gz<br><sample_name>_DBEC_ReadsPerCell_Unfiltered.csv.gz | Unfiltered tables containing all cell labels of ≥5 reads |
| Expression data[a] | <sample_name>_Expression_Data.st | The expression sparse matrix, a table of counts in sparse format |
| | <sample_name>_Expression_Data_Unfiltered.st.gz | Compressed file containing all cell labels of ≥5 reads |

| Output (continued) | File | Content |
|---|---|---|
| Cell label filtering | <sample_name>_Cell_Label_Filter.png | Visualization of cell label filtering results |
| Second derivative curve | <sample_name>_Cell_Label _Second_Derivative_Curve.png | |
| Putative cells origin | <sample_name>_Putative_Cells _Origin.csv | Algorithm that found the putative cell: basic or refined |
| Unique Molecular Identifier metrics | <sample_name>_UMI_Adjusted_Stats.csv | Metrics from RSEC and DBEC Unique Molecular Identifier adjustment algorithms on a per-gene basis |
| Clustering analysis | ClusteringAnalysis.zip | See Clustering analysis output files (page 58) |

a. For a multiplexed samples run, the tables contain counts for putative cells from all samples combined.

If the multiplex option was selected, the following outputs are generated:

| Output | File | Content |
|---|---|---|
| Sample Tags metrics | <sample_name>_Sample_Tag_Metrics.csv | Metrics from the sample determination algorithm |
| Sample Tag calls | <sample_name>_Sample_Tag_Calls.csv | Assigned Sample Tag for each putative cell |
| Per-sample folder | <sample_name> _Sample_Tag<number>.zip<br><sample_name>_Multiplet_and _Undetermined.zip | Data tables, expression matrix, and clustering analysis files for a particular sample.<br><br>**Note:** For putative cells that could not be assigned a specific Sample Tag, a Multiplet_and _Undetermined.zip file is also output. |

# Clustering analysis output files

The BD Rhapsody™ Clustering Analysis app outputs one or more sets of four output files (cluster labels, t-SNE projection labelled by clusters, cluster features, and pairwise cluster features) that describe levels of clustering:

| Output | File | Content |
|---|---|---|
| t-SNE coordinates | <sample_name> _bh-tSNEcoordinates.csv | Coordinates of the t-SNE projection |
| Cluster labels | <sample_name>_<num_clusters> _Labels.csv | Cluster membership per cell |
| t-SNE plot | <sample_name>_<num_clusters> _tSNE.png | Visualization of the t-SNE projection with cells colored by cluster labels |
| Over-represented genes in each cluster to all clusters | <sample_name>_<num_clusters> _Cluster_Features.csv | Top 50 statistically over-represented genes compared to all clusters |
| Over-represented genes in each cluster to every other cluster | <sample_name>_<num_clusters> _Pairwise_Cluster_Features.csv | Top 50 statistically over-represented genes compared to every other cluster |
| (Optional) Concatenated data tables | < sample names>_MolsPerCell.csv or <sample names>_Expression_Data.st | Combined data table; output only if multiple inputs specified |
| (Optional) Sample IDs | SampleIDs.csv | Table of sample IDs; output only if multiple inputs specified |

# Reviewing output files

See the *BD Single Cell Genomics Bioinformatics Handbook* (Doc ID: 54169).

Genomics technical publications are available for download from the BD Genomics Resource Library at bd.com/genomics-resources.

This page intentionally left blank

# 9

# Installing BD™ Data View

# Launching BD Data View

**Introduction**

BD Data View is a software application for visualization of high-dimensional gene expression data derived from single cells.

BD Genomics Data View v1.1 and later requires MATLAB Runtime. For detailed information about the MATLAB Runtime and the MATLAB Runtime installer, see the MathWorks Documentation.

**Software requirements**

- Windows only: If you do not have an unzipping application, obtain an unzipping program, such as Winzip. Go to winzip.com/win/en/downwz.html.
- MATLAB Runtime R2017b (9.3) for Windows® or Macintosh. Go to mathworks.com/products/compiler/mcr.html.

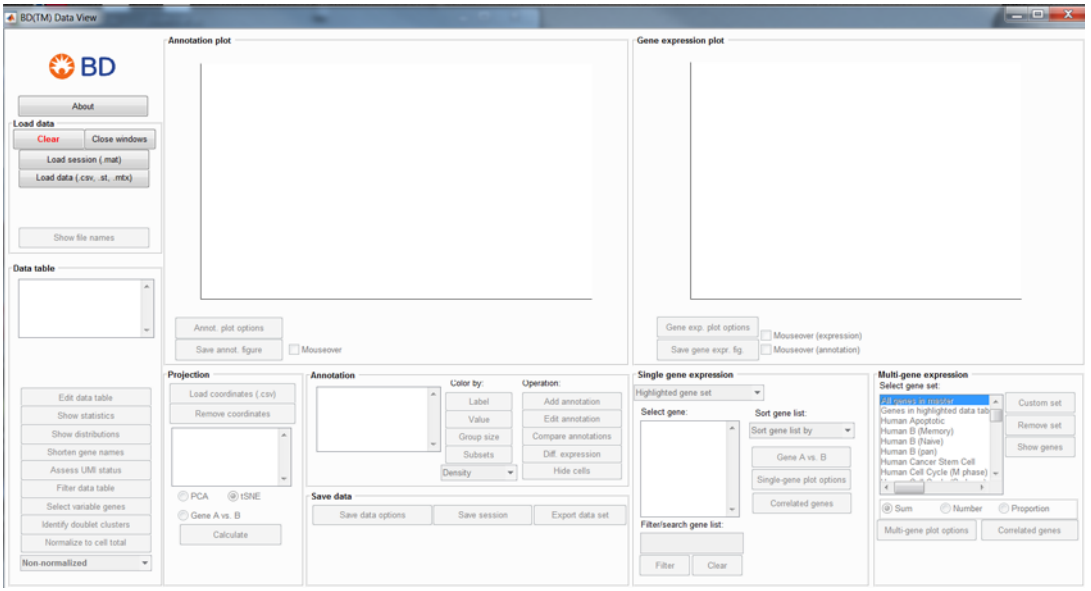
  **Install the specified MATLAB runtime version.**

**Download BD Data View**

1. Go to bitbucket.org/CRSwDev/dataview.
2. In the left pane, click **Downloads > Download Repository**. The application is downloaded.
3. Unzip the downloaded folder.

### Launching BD Genomics Data View on the Microsoft® Windows® operating system

Navigate to the folder that contains the application, and then double-click the application icon to launch the application. The BD Data View dashboard is displayed:




**Note:** The terminal window runs in the background. It shows the application status. You can monitor the progress of computationally intensive tasks and view any error messages that might help with troubleshooting. Do not close the window.

### Launching BD Genomics Data View on the Apple macOS® operating system

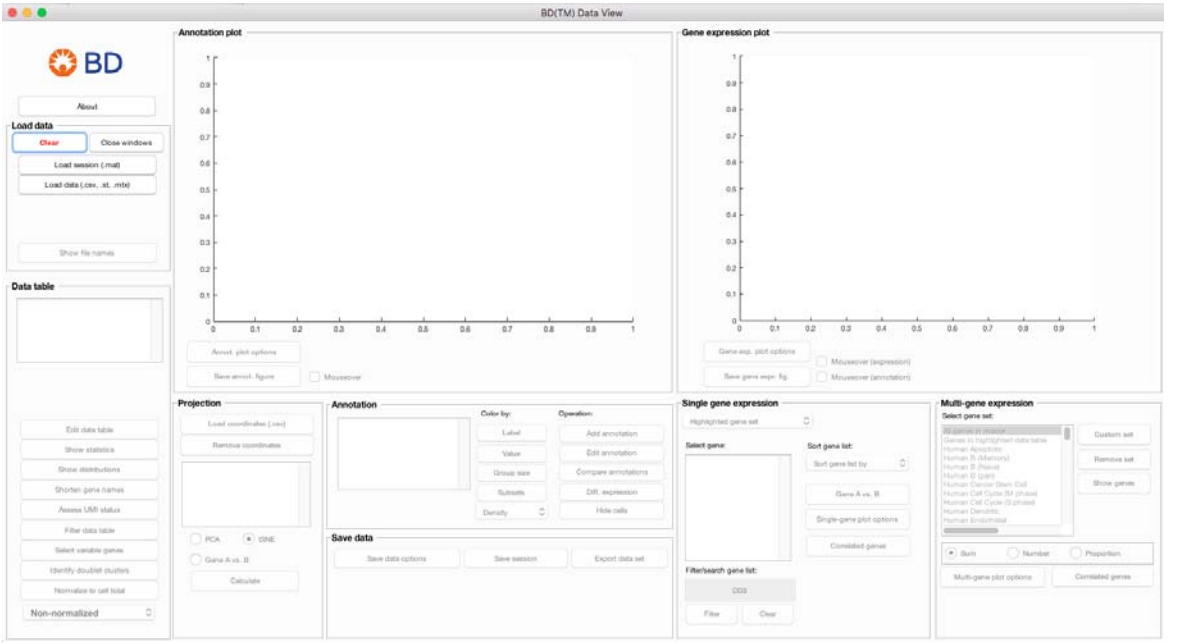
BD Biosciences recommends launching BD Data View from the application terminal window. In the terminal window, you can monitor the progress of computationally intensive tasks and view any error messages that might help with troubleshooting analysis.

1. Locate the MATLAB runtime folder. In most installations, MATLAB runtime is located in Applications/MATLAB/MATLAB_Runtime/v93.
2. Press the command ⌘ key + space bar, and then type `Terminal` to find and launch the terminal application.
3. At the terminal prompt, enter the appropriate path to navigate to the folder that contains the BD Data View application and .sh script file:
   `$ cd <path>`
4. At the prompt, run the Data View bash script followed by the path of your MATLAB runtime installation:
   `$ ./run_DataView.sh <path/to/matlab/install>`

   For example:

   ```
   $ ./run_DataView.sh Applications/MATLAB/
   MATLAB_Runtime/v93/
   ```

The BD Data View dashboard is displayed:



This page intentionally left blank

# 10

# Troubleshooting

# Analysis pipeline

**Introduction**

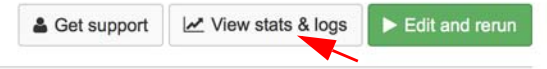
This topic describes how to respond to a task failure while running the BD Biosciences pipeline.

**Arranging BD Biosciences to join the project on Seven Bridges Genomics**

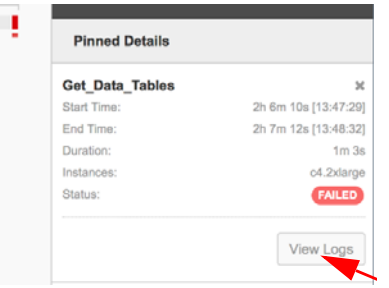
**If a task fails on the Seven Bridges Genomics platform, contact BD Biosciences technical support at researchapplications@bd.com to troubleshoot the issue. Tech support will provide you with instructions on inviting a support team member to your project. To troubleshoot the issue yourself, access the log files. See Downloading the log file from Seven Bridges Genomics.**

**Downloading the log file from Seven Bridges Genomics**

1. From within a failed task, click **View Stats & Logs** in the upper right corner:

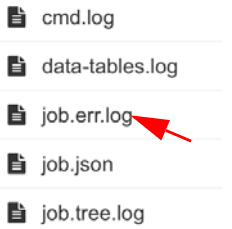



2. Locate the failed node in your pipeline run. Completed nodes are in green, and the failed node is in red. Click the failed node, and on the right, click **View Logs** for that node:




A list of files contained in the failed node are displayed.

3. Click **job.err.log** to display the log content and download it:




### Accessing the log file in a local installation

If a pipeline run completed successfully, all logs are collected in a Logs folder in your output directory. But if a pipeline run fails, the Logs folder is absent from the directory. You need to navigate to the *tmp* directory containing the intermediate files for that node to obtain the log files:

1. In the terminal STDOUT, find the failed node command call from CWL-runner. This is the most recent command call.
2. Locate the tmp folder name, which is in the format:

   ```
   [job Name_of_failed_node] /tmp/tmpb0kyIg $
   ```

3. Navigate to that directory. The log file will have the .log extension.
4. Send the log file to researchapplications@bd.com, or contact BD Biosciences technical support without it.

# BD™ Data View installation

**Introduction**

This topic describes possible problems and recommended solutions for setting up visualization analysis.

**BD Data View does not launch**

| Possible causes | Recommended solutions |
|---|---|
| Incorrect version of MATLAB Runtime installed for Windows® or Macintosh operating system | Install MATLAB Runtime R2017b (9.3) for Windows or Macintosh. |

**Application does not open after launching from terminal window**

| Possible causes | Recommended solutions |
|---|---|
| Copy error | Ensure that the .sh file is in the same folder as the application. |

# Glossary

# B

**BAM** An alignment file in binary format. A binary SAM file.

# C

**called cell** A putative cell that has been assigned a Sample Tag.

**CWL** Common workflow language. A way to describe commands and to connect them to create workflows.

# D

**data tables** Output of BD Rhapsody™ Analysis pipeline containing read count or molecule count per gene.

**DBEC** Distribution-based error correction.

# F

**FASTA** Text-based format that contains one or more DNA or RNA sequences.

**FASTQ** A file in standardized, text-based format that contains the output of base reads and per-base quality values from a sequencer.

# L

**library** A sequencing library derived through amplification of genomic material that had been captured by a collection Cell Capture Beads from a BD Rhapsody™ kit.

# P

**putative cell** A single cell determined to be putative by the cell label filtering algorithm.

# R

**R1 reads** Contains information about the cell label and molecular identifier.

**R2 reads** Contains information about the gene.

**RSEC** Recursive substitution error correction.

# S

**SAM** Tab-delimited text file with sequence alignment data.

**Sample Tag** Antibody-oligo tag that identifies a sample in a multiplexed run.

# T

**t-SNE** t-distributed stochastic neighbor embedding (t-SNE). An algorithm for dimensionality reduction. It allows for the representation of high-dimensional data (multiple genes across multiple cells) into a two-dimensional space, which can then be visualized in a scatter plot.

# Y

**YML** YAML: “YAML ain't markup language.” A data serialization language used for configuration files to various applications.

This page intentionally left blank