Setup and usage guide for UniverSC GUI app

Check for the following before you start

Make sure that…

- The computer has sufficient space for the sample reads, reference genome, output data.
- The computer has **32GB of RAM** or more.
  UniverSC uses STAR aligner for aligning reads to a reference. 27GB of RAM was used to map reads to human genome. So, a comfortable execution of the app will need 32GB or RAM.
- The computer has **500G or free storage** or more.
  As a part of its process to format your input files, UniverSC app will unzip the already-large .fastq.gz files. These files can quickly reach over hundreds of GB. If the app is used on Windows or Mac (when a Docker image is used), make sure that there is enough space where the Docker container will be created.
- The OS of the computer is **Ubuntu 22.04** (for Linux), **Monterey 12.4** (for Mac), or **Windows 10** (for Windows).
  While it may very well work on other versions, UniverSC has been specifically tested on these versions of operating systems.

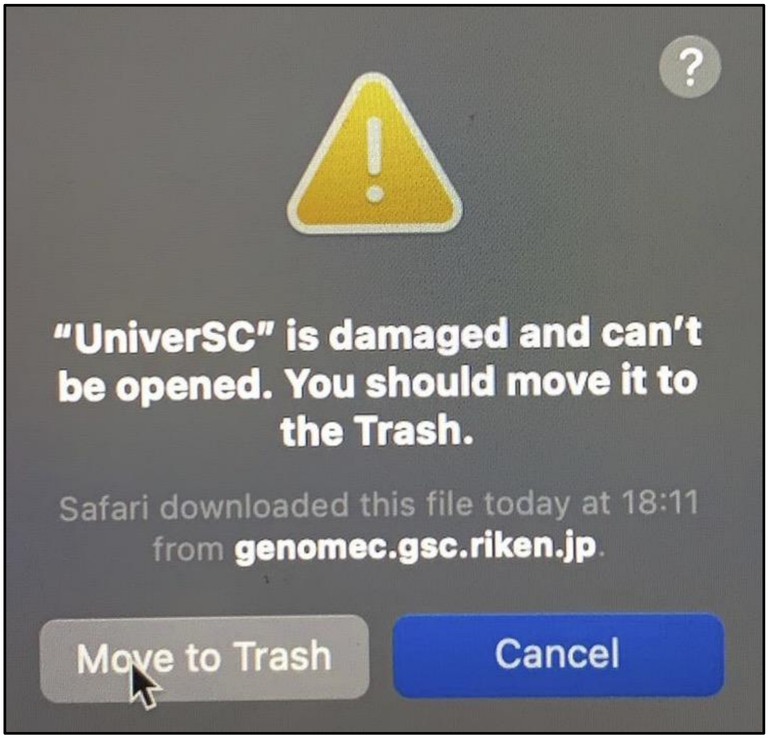
Guide for setting up UniverSC GUI app

1. Download and unzip the UniverSC GUI app for the appropriate platform from the following link:

   https://genomec.gsc.riken.jp/gerg/UniverSC/UniverSC_app_release

(If you are not a Mac user, skip to Step 3.)

2. Clear the extended attribute.
   Depending on the version of the OS, Mac can give you the following error:



“UniverSC” is damaged and can’t be opened. You should move it to the Trash.

Safari downloaded this file today at 18:11 from **genomec.gsc.riken.jp**.

This is due to Mac's incompatibilities with some of the extended attributes of UniverSC GUI app. When this occurs, click on “Cancel” DO NOT MOVE TO TRASH. Instead, open the terminal and enter the following command to clear all extended attributes of the app.

```
xattr -cr /SOME/PATH/TO/YOUR/APP/UniverSC.app
```

Alternatively, change your working directory to where UniverSC GUI app is at by using the ‘cd’ command and then enter the following.

```
xattr -cr UniverSC.app
```

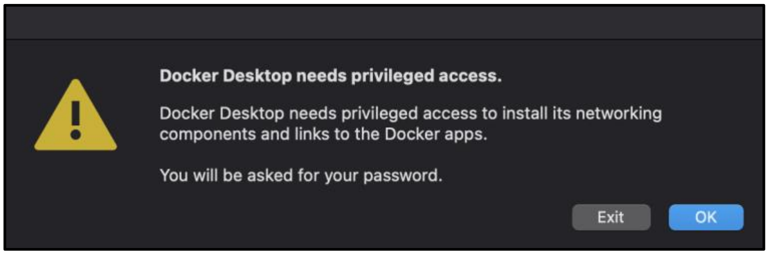
3. Execute the UniverSC GUI app by double clicking on the icon to confirm that the app opens properly. It will take some time for the app to open up.

(If you are a Linux user, skip to Step 9. Docker is not needed.)

4. Download Docker for desktop installer from their official website:

https://www.docker.com/products/docker-desktop

5. Install Docker.
For Mac: Copy Docker icon to the Application folder as instructed. Following window will appear. Press OK (it will take a moment to start up).

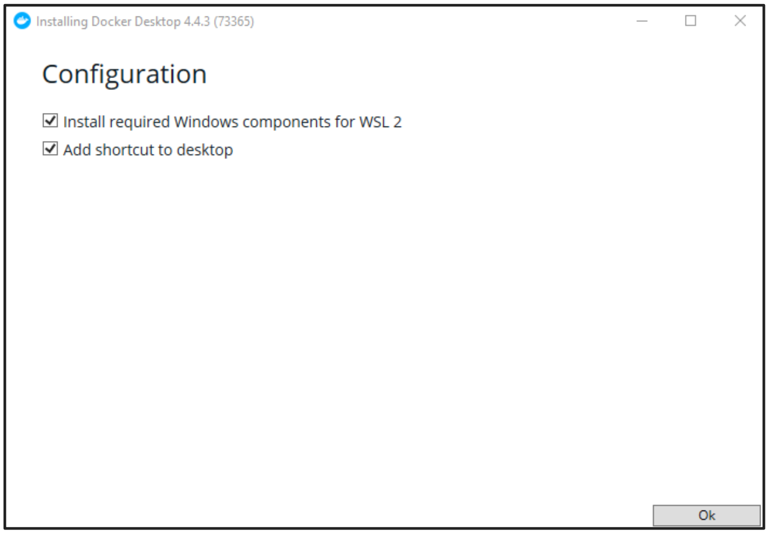


Docker Desktop needs privileged access.

Docker Desktop needs privileged access to install its networking components and links to the Docker apps.

You will be asked for your password.

For Windows: Execute the installer by double clicking the installer.
NOTE: If asked, make sure to configure required Windows components for WSL 2.

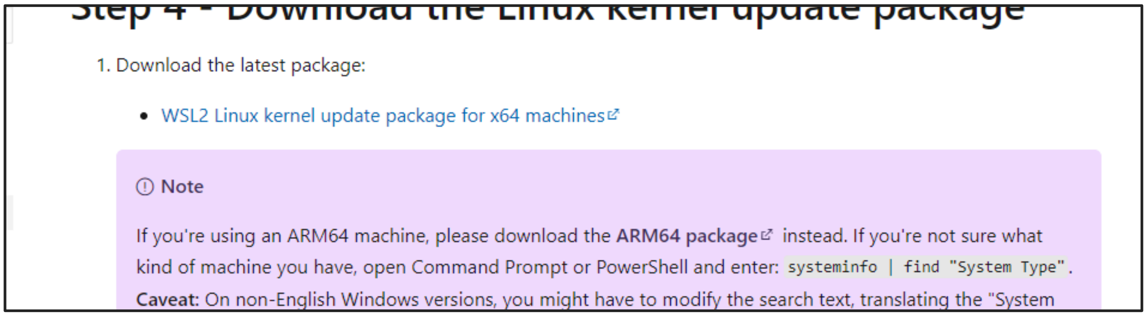


Installing Docker Desktop 4.4.3 (73365)

Configuration

☑ Install required Windows components for WSL 2
☑ Add shortcut to desktop

For Windows: Install the Linux kernel manually. Follow the link to the kernel to download and install the Linux kernel. This will complete the process to install Docker.



Step 4 - Download the Linux kernel update package

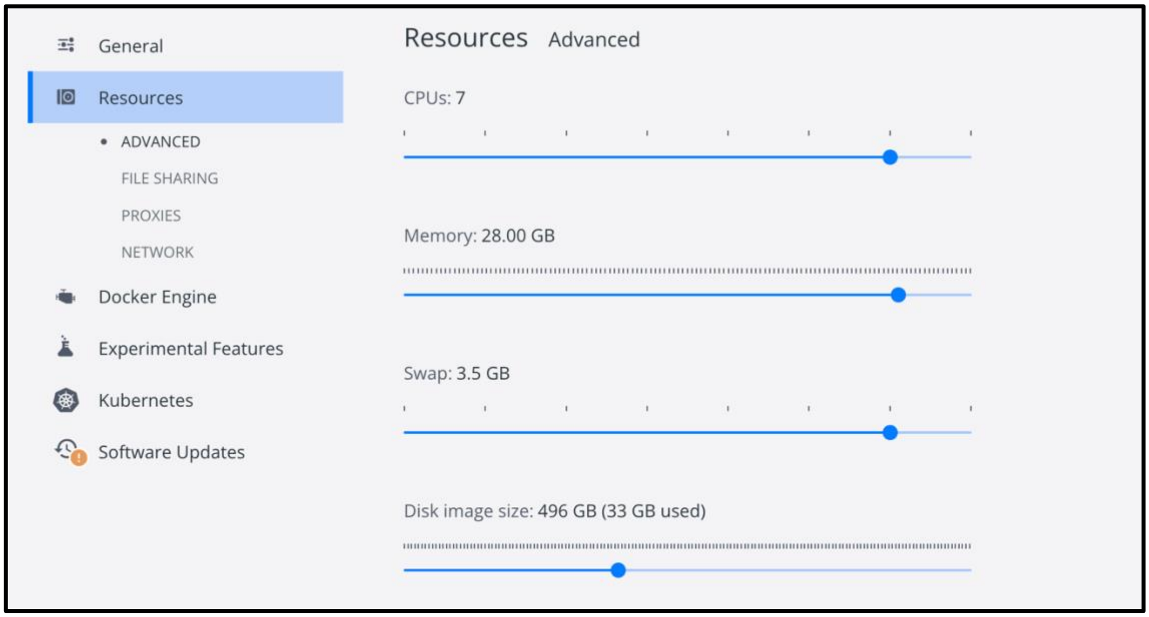
1. Download the latest package:

- WSL2 Linux kernel update package for x64 machines

ⓘ Note

If you're using an ARM64 machine, please download the **ARM64 package** instead. If you're not sure what kind of machine you have, open Command Prompt or PowerShell and enter: `systeminfo | find "System Type"`. **Caveat:** On non-English Windows versions, you might have to modify the search text, translating the "System

6. If needed, allocate sufficient resources to Docker.



Resources Advanced

CPUs: 7

Memory: 28.00 GB

Swap: 3.5 GB

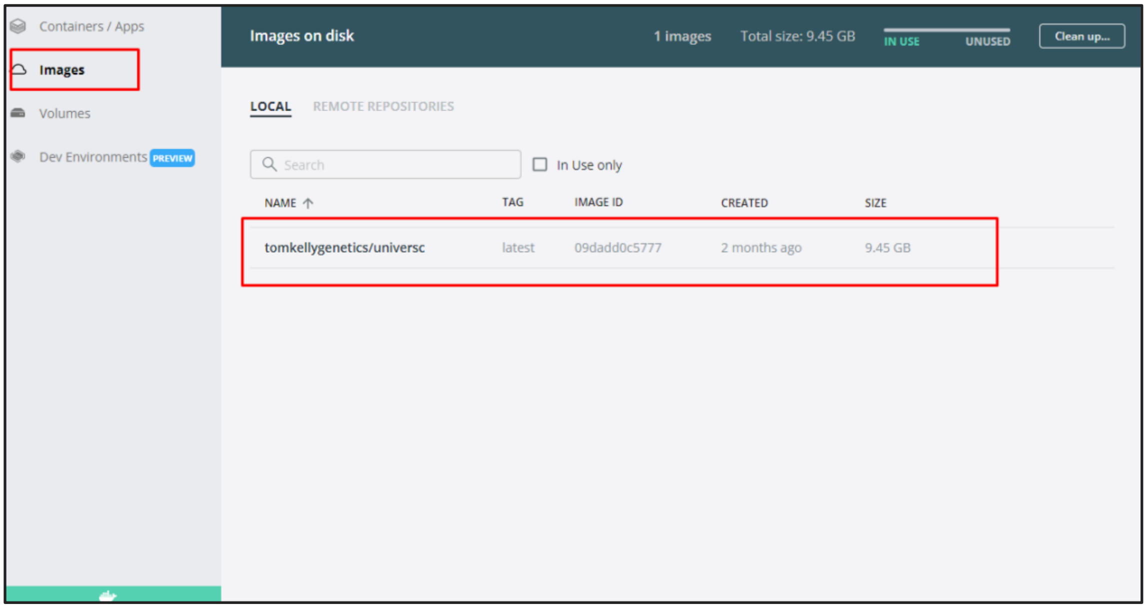
Disk image size: 496 GB (33 GB used)

NOTE: The screen shown above will NOT be available if using Windows WSL2. See Appendix 1 for hints on how to allocate recourses when using Windows WSL2.

7. When Docker installation is complete, open the command prompt (on Windows) or the terminal (on Mac), and enter the following command to pull the Docker image for the most updated version of UniverSC.

```
docker pull tomkellygenetics/universc:latest
```

8. When the command runs to completion successfully, an image of UniverSC should appear on the Docker application under ‘Images’.



NAME ↑	TAG	IMAGE ID	CREATED	SIZE
tomkellygenetics/universc	latest	09dadd0c5777	2 months ago	9.45 GB

(If you are not a Linux user, skip to Step 11. Command-line UniverSC is not needed.)

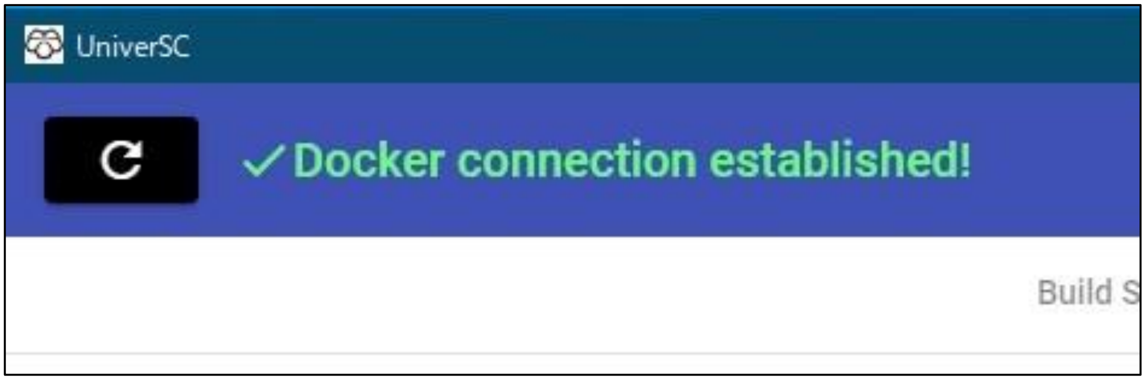
9. Download and unzip the most updated UniverSC command line version from the following link:

https://github.com/minoda-lab/universc/releases

10. Download and install Cell Ranger 3.0.2.

https://github.com/10XGenomics/cellranger

11. Execute the UniverSC GUI app by double clicking the app. You should see the green message in the top left corner signaling that the Docker connection is established.



✓Docker connection established!

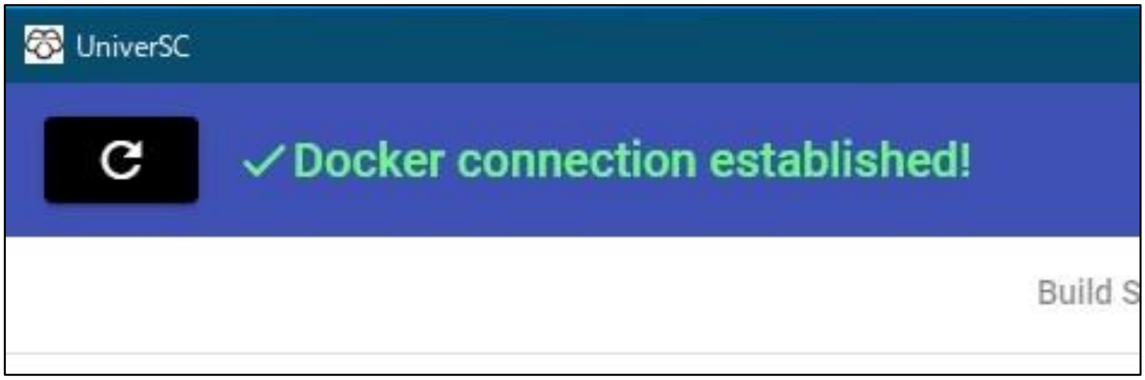
Guide for using UniverSC GUI app

NOTE: This guide is based on the processing of test data provided at https://genomec.gsc.riken.jp/gerg/UniverSC

I. Making a genome reference

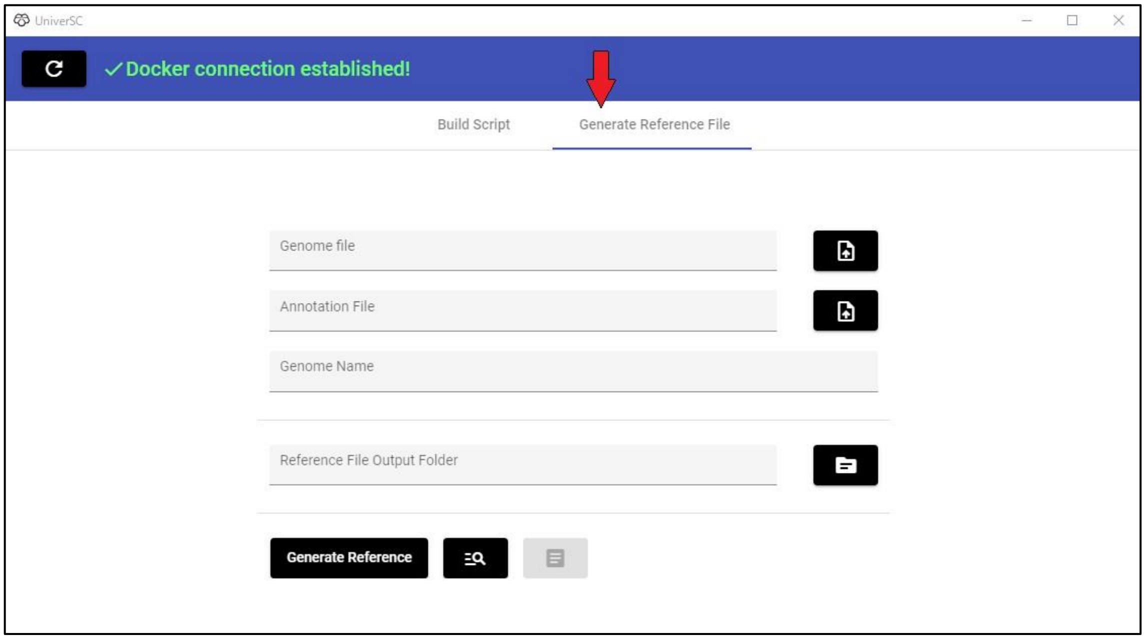
A reference genome file(s) must be formatted prior to analyzing a sample.

1. Execute the UniverSC GUI app by double clicking the app. You should see the green message in the top left corner signaling that the Docker connection is established.



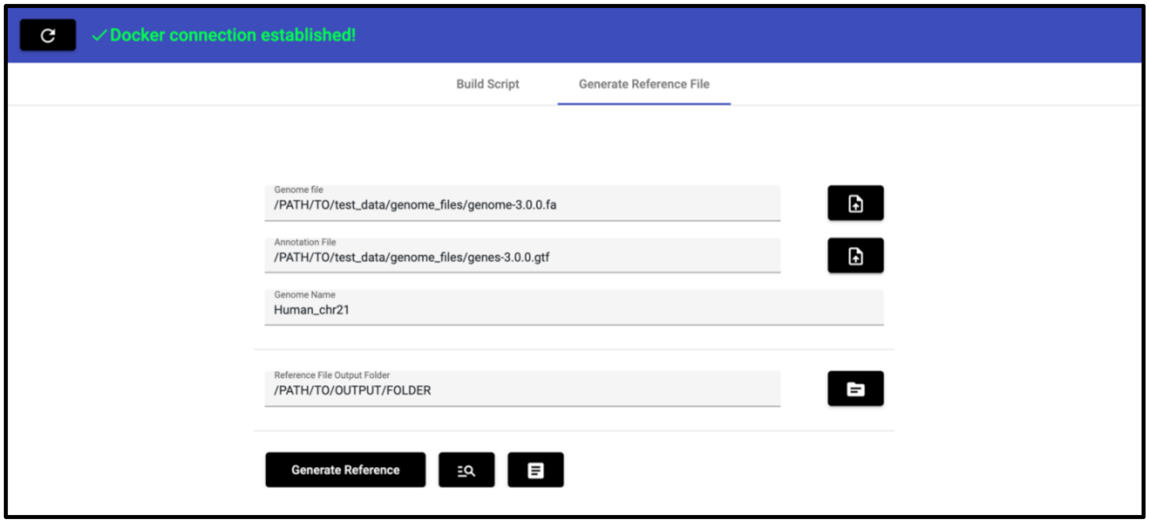
✓Docker connection established!

2. Choose the “Generate Reference File” tab.



✓Docker connection established!

3. Select the genome file, the annotation file, a name for the newly generated reference file, and an output folder.



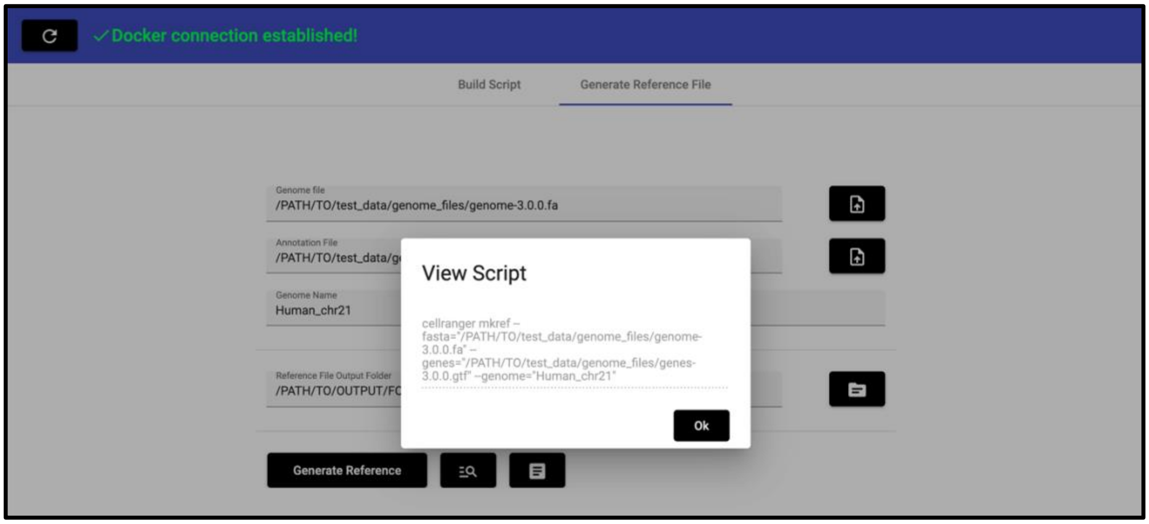
Genome file: /PATH/TO/test_data/genome_files/genome-3.0.0.fa

Annotation File: /PATH/TO/test_data/genome_files/genes-3.0.0.gtf

Genome Name: Human_chr21

Reference File Output Folder: /PATH/TO/OUTPUT/FOLDER

(If you wish to see the actual command passed, click on the “View Script” icon.)



Genome file: /PATH/TO/test_data/genome_files/genome-3.0.0.fa

Annotation File: /PATH/TO/test_data/g

Genome Name: Human_chr21

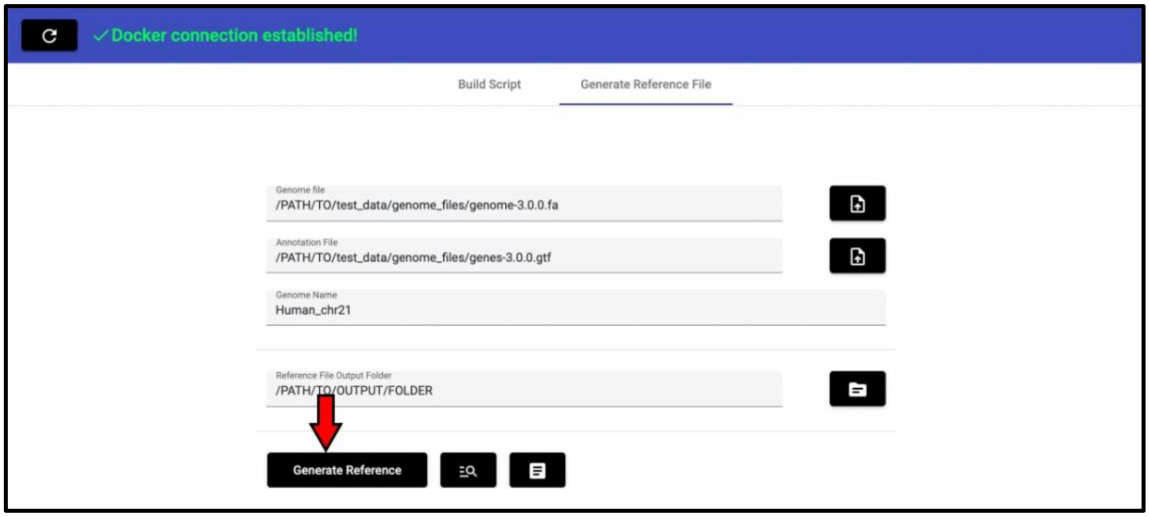
Reference File Output Folder: /PATH/TO/OUTPUT/FC

View Script

```
cellranger mkref –
fasta="/PATH/TO/test_data/genome_files/genome-
3.0.0.fa" –
genes="/PATH/TO/test_data/genome_files/genes-
3.0.0.gtf" –genome="Human_chr21"
```

4. Generate a new reference by clicking on “Generate Reference” icon.

   NOTE: This step should take about a minute with test data.



Genome file
/PATH/TO/test_data/genome_files/genome-3.0.0.fa

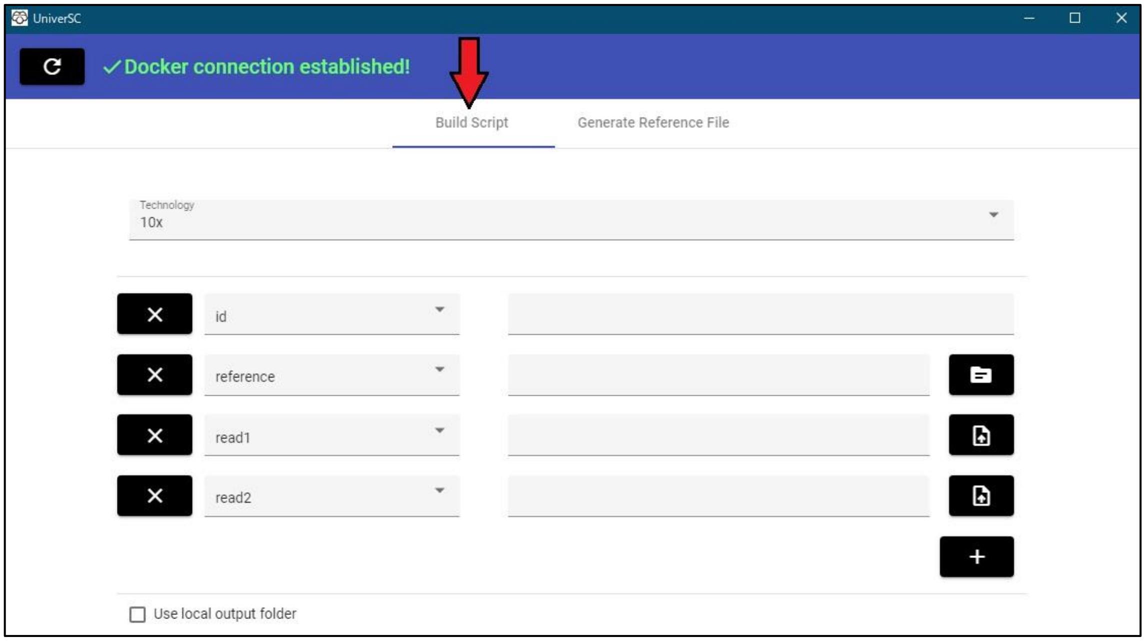
Annotation File
/PATH/TO/test_data/genome_files/genes-3.0.0.gtf

Genome Name
Human_chr21

Reference File Output Folder
/PATH/TO/OUTPUT/FOLDER

II. Processing your sample

1. Choose the “Build Script” tab.



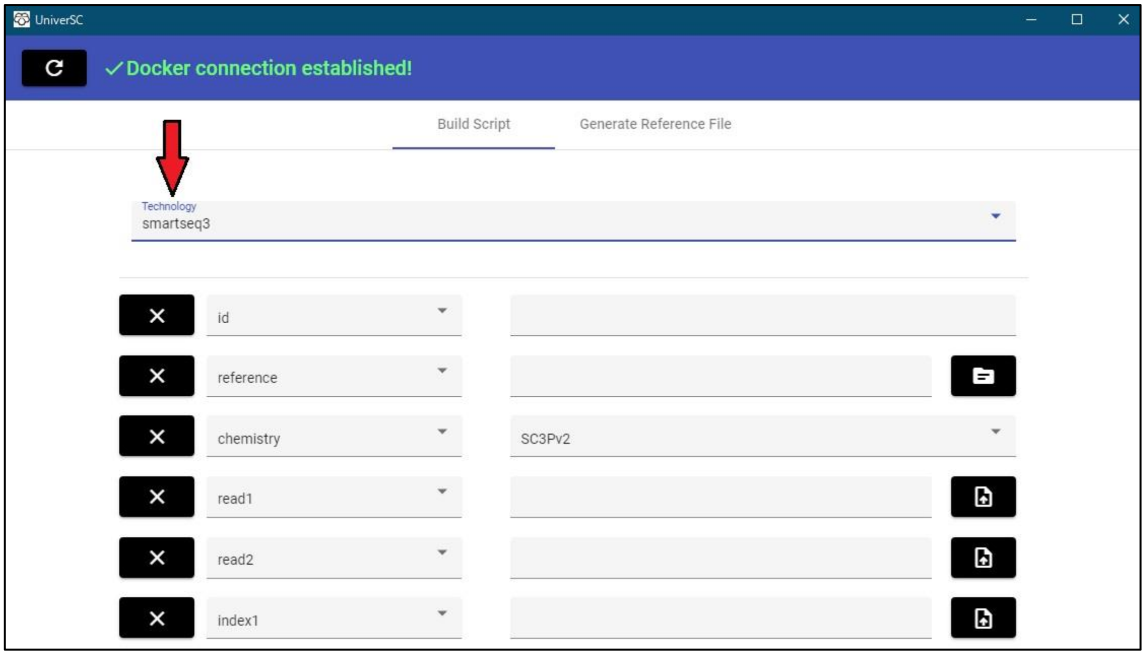
✓Docker connection established!

Technology
10x

□ Use local output folder

2. Select the appropriate technology from a pulldown list. All the required additional parameters will automatically appear.

   NOTE: Chose 10x to process test data.



Technology
smartseq3

id

reference

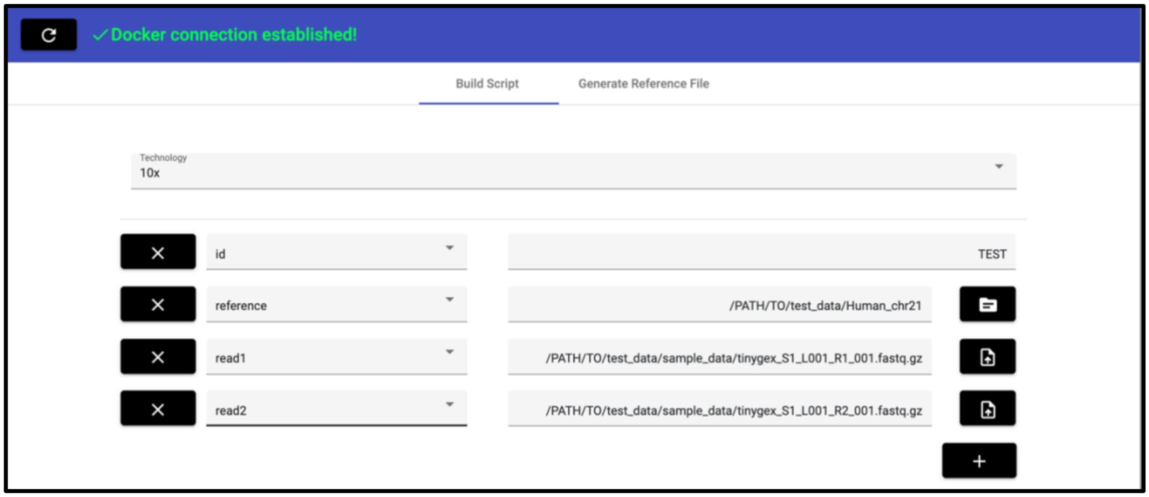
chemistry SC3Pv2

read1

read2

index1

3. Fill out the ID name, select reference and sample files.



Technology
10x

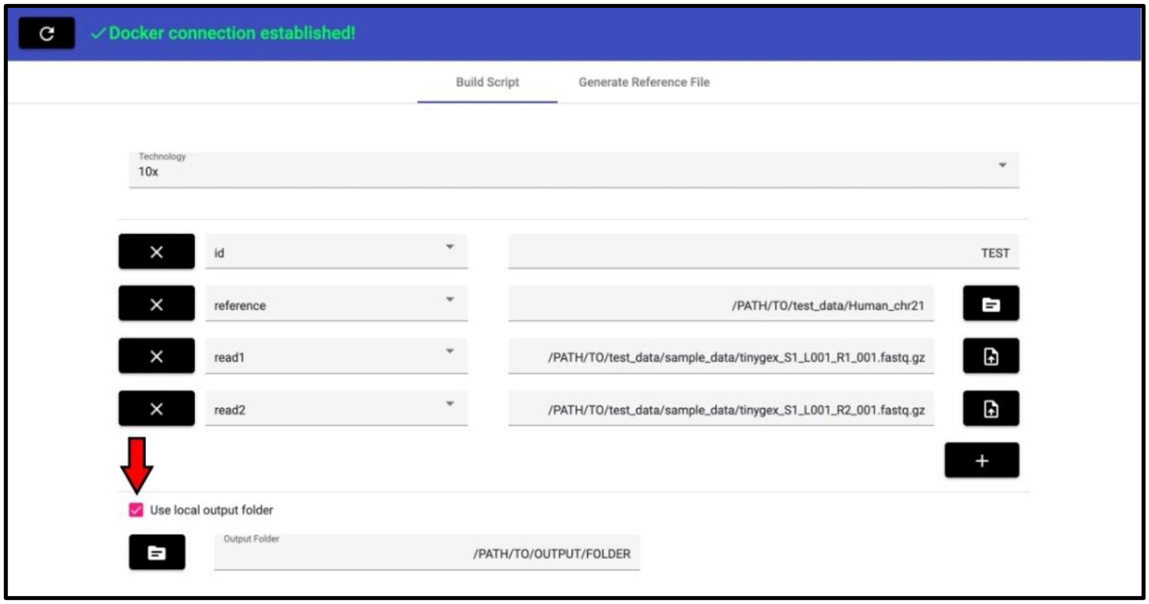
id TEST

reference /PATH/TO/test_data/Human_chr21

read1 /PATH/TO/test_data/sample_data/tinygex_S1_L001_R1_001.fastq.gz

read2 /PATH/TO/test_data/sample_data/tinygex_S1_L001_R2_001.fastq.gz

4. Click on the “Use local output folder” box to output the results to your computer outside of Docker. Otherwise, the output files will remain within the Docker container and cannot be directly accessed from the local computer.



✓ Docker connection established!

Technology: 10x

id: TEST

reference: /PATH/TO/test_data/Human_chr21

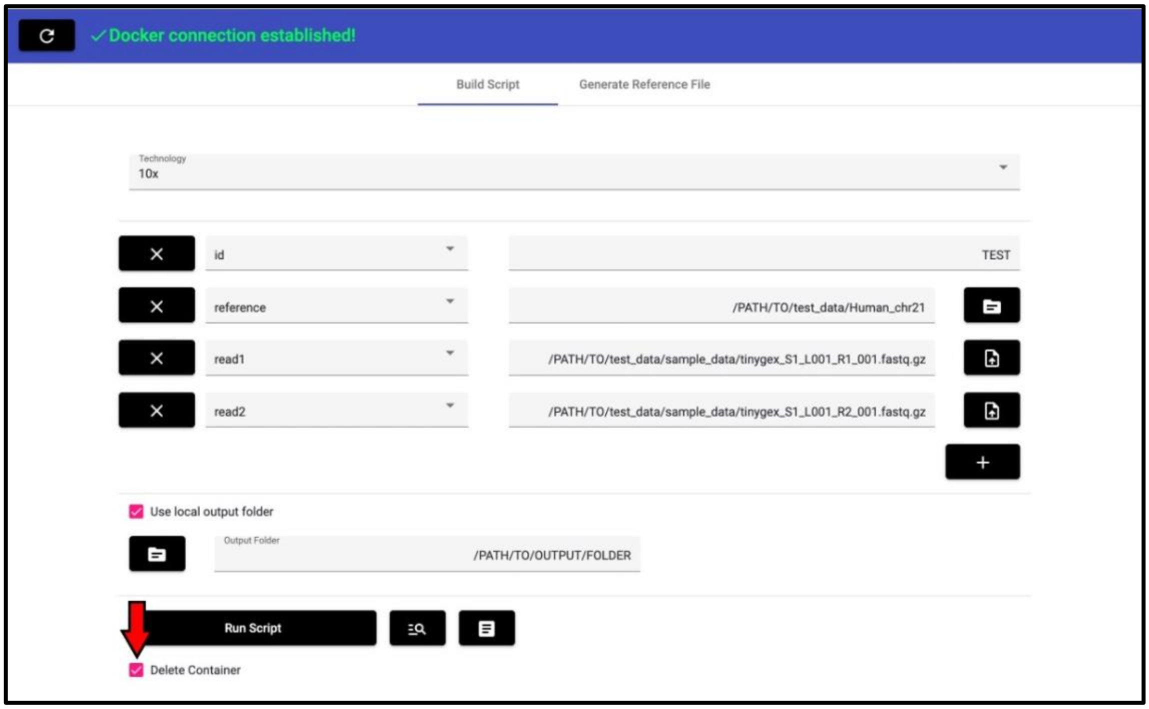
read1: /PATH/TO/test_data/sample_data/tinygex_S1_L001_R1_001.fastq.gz

read2: /PATH/TO/test_data/sample_data/tinygex_S1_L001_R2_001.fastq.gz

☑ Use local output folder

Output Folder: /PATH/TO/OUTPUT/FOLDER

5. Click on the “Delete Container” box to remove the Docker image made during the process to save space on your computer.



✓ Docker connection established!

Technology: 10x

id: TEST

reference: /PATH/TO/test_data/Human_chr21

read1: /PATH/TO/test_data/sample_data/tinygex_S1_L001_R1_001.fastq.gz

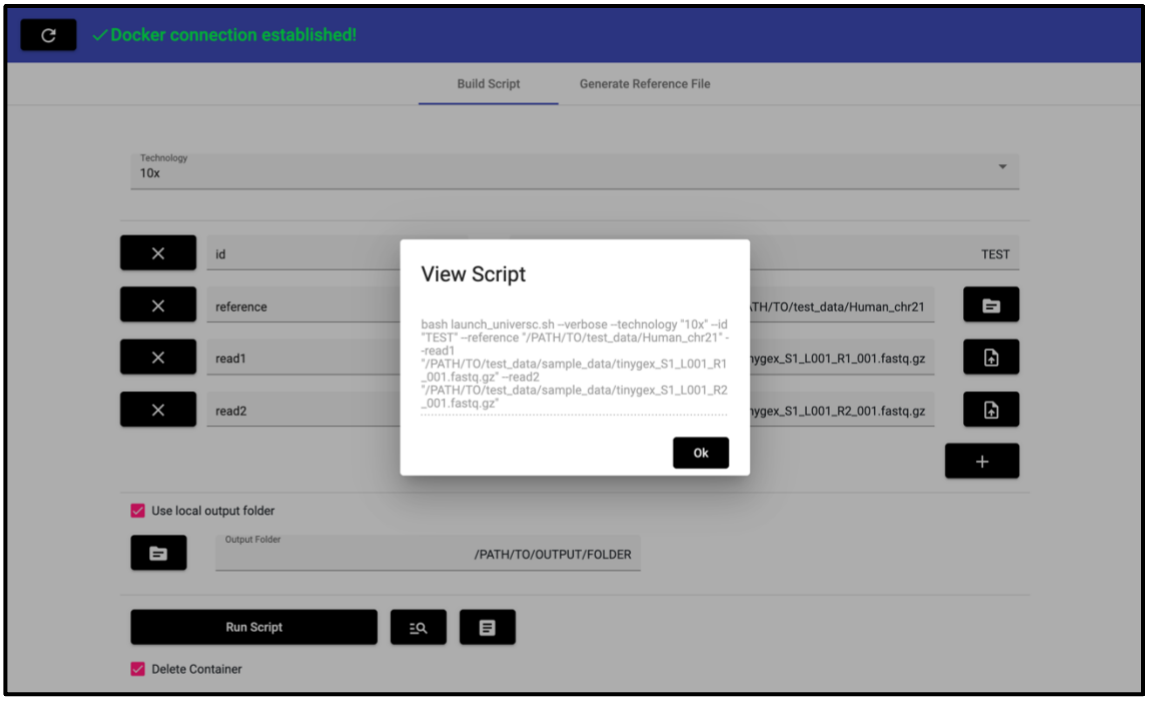
read2: /PATH/TO/test_data/sample_data/tinygex_S1_L001_R2_001.fastq.gz

☑ Use local output folder

Output Folder: /PATH/TO/OUTPUT/FOLDER

☑ Delete Container

(As in the case of making a new reference, if you wish to see the actual command passed, click on the “View Script” icon.)



Technology
10x

id TEST

reference TH/TO/test_data/Human_chr21

read1 ygex_S1_L001_R1_001.fastq.gz

read2 ygex_S1_L001_R2_001.fastq.gz

View Script

```
bash launch_universc.sh --verbose --technology "10x" --id
"TEST" --reference "/PATH/TO/test_data/Human_chr21" -
-read1
"/PATH/TO/test_data/sample_data/tinygex_S1_L001_R1
_001.fastq.gz" --read2
"/PATH/TO/test_data/sample_data/tinygex_S1_L001_R2
_001.fastq.gz"
```

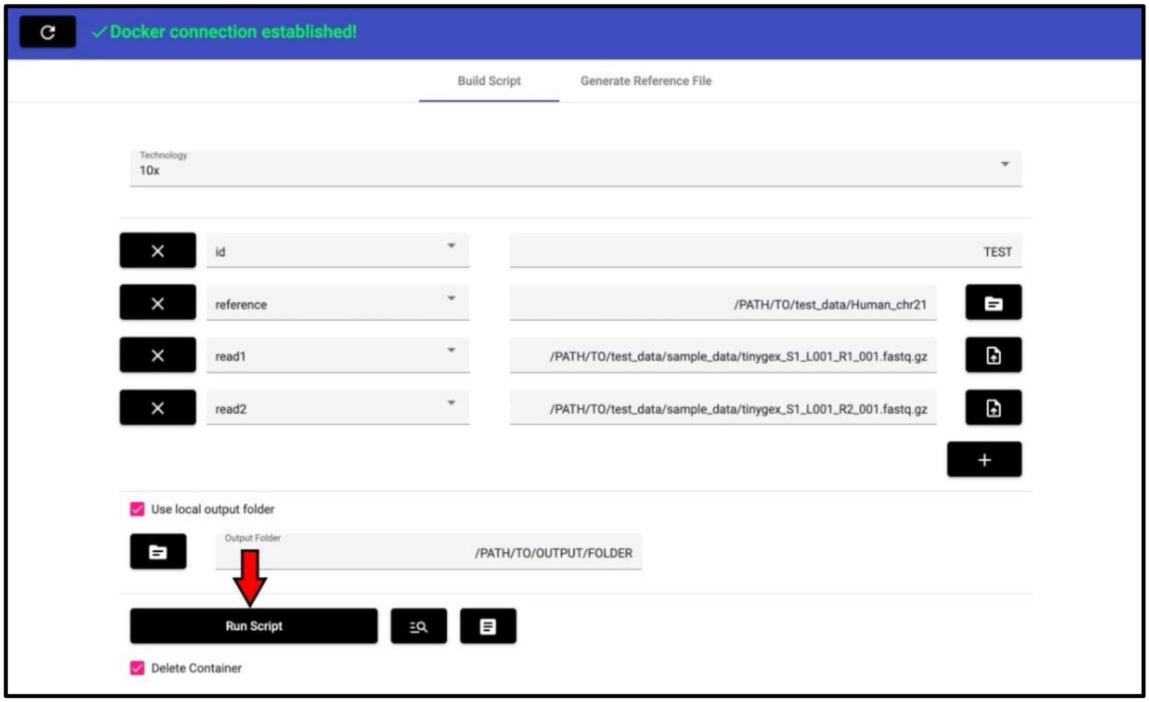
☑ Use local output folder

Output Folder
/PATH/TO/OUTPUT/FOLDER

☑ Delete Container

6. Process your data by clicking on “Run Script” icon.

   NOTE: This step should take about 5-10 minutes with test data.



✓Docker connection established!

Technology: 10x

id: TEST

reference: /PATH/TO/test_data/Human_chr21

read1: /PATH/TO/test_data/sample_data/tinygex_S1_L001_R1_001.fastq.gz

read2: /PATH/TO/test_data/sample_data/tinygex_S1_L001_R2_001.fastq.gz

☑ Use local output folder

Output Folder: /PATH/TO/OUTPUT/FOLDER

☑ Delete Container

7. Check if your output matches the output provided within test data.

Appendix 1: Hints for adding resources to in Docker (WSL2)

UniverSC app will likely require the Docker container to have more recourses allocated than its default values. Here are “hints” on how to increase these values.

NOTE: As some of these steps may differ from computer to computer, consider the following information as “hints”.

- Number of threads, RAM, and swap memory
  To increase these values, modifications must be made on the “.wslconfig” file. This file is usually located “C:\Users\<User>”. Within the .wslconfig file, each recourses can be set as below:

  ```
  [wsl2]
  memory=50GB
  swap=32GB
  swapFile=C:\Users\Administrator
  processors=28
  ```

- Storage
  The default storage size for a Docker container is set at 256GB. This needs to be increased through a multi-step process like below:

  i. Each Docker container is a virtual hard drive. The first step is to determine the current size of the container and where these containers are mounted. Determine this by building a new UniverSC container, open its terminal. Type in “df -h” to determine the disk size. The disk named “overlay” shows the container size and “/dev/sdX” (usually sdc) is the virtual hard drive.
  ii. To confirm that such virtual hard drive exists, open Ubuntu command-line and type in “mount | grep “/dev/sdX”, which should display the disk mounted somewhere.
  iii. The next step is to resize the virtual hard drive. In the Ubuntu command-line type in “sudo resize2fs /dev/sdX <SIZE>M” (size in MBs).