Instructions

1. To populate our visualizations, we will have to create a directory in which we will work in. To do this, we will type into our terminal “mkdir your directory name”
2. Next, we will install docker which we can do directly off https://github.com/nilsleiffischer/gwpv.
3. To produce these images, we will also need to install paraview. This can be done on https://www.paraview.org/download/ and we could download the most recent program.
4. Our final program we will install is visual studio code, as this will allow us to edit some of the files we will be rendering.
5. After we have all our programs installed, our next step will be to download what files will need to be rendered from nilslieffischer. To do this, we will need to download the file GWPV from https://github.com/nilsleiffischer/gwpv
6. When all the required files are downloaded, the next step is to ensure docker Is running. To run docker, our command is `docker run “nilsleiffischer/gwpv:latest”`
7. This will render what images we need, Our next command will be to run “`docker run –v $PWD:/out nilsleiffischer/gwpv:latest scene Examples/Rainbow/Still.yaml –o /out`” This will download the waveforms that we need
8. After this command runs, we will see that we have the files we need and can see them by typing the command “ls”. The yaml files we need will be in our directory. In order to produce the images and clips from Blackholes.org we will need to edit the yaml files so

that they can produce different simulations. To do this, we will choose a simulation, find the simulation at zenodo.org, open the Datasources.yaml file in visual studio code (it should be saved under our directory name), and copy and paste our identification number for our simulation from zeondo into our file.

9. We will then run our command which will look like this “docker run -v $PWD:/out nilsleiffischer/gwpv:latest scene File:https://zenodo.org/record/3314848/files/Lev3/rhOverM_Asymptotic_GeometricUnits_CoM.h5/Still.yaml -o /out”
10. Our next command will be “`% docker run –v $PWD:/out nilsleiffischer/gwpv:latest scene /out/Rainbow.yaml –o /out”.` This will Render our files and if we list what is in our directory, we will find that our frames have generated.
11. From the data sources yaml file we will see that we can change multiple different parameters for our images. In this example, we will leave as is. Once we have all our frames downloaded, we can then run these images on our supercomputer. To do this we log into our supercomputer, make a directory preferably with the same name we have been working in, and our frames will appear in our super computer
12. After all of our frames have been opened, we can open our frames one by one and scroll through them, or we can turn them into s short animation to do this we will have to download ffmpeg from https://www.ffmpeg.org/ .This command allows us to turn our frames into a movie. Ffmpeg is capable of more, but for our example this is what we will be using

13. After we have downloaded it our command to run will be

```
/home/samuel/TestingGround/ffmpeg-4.4-amd64-static/ffmpeg -r 60 -f image2 -start_number 0000 -s 1920x1080 -i frame.%06d.png -vcodec libx264 -crf 20 -pix_fmt yuv420p test.mp4
```

14. This command will allow us to turn our images into a short movie seen as “test.mp4”

15. For our command the only part that we need to change is our path. In this example, our path is set as “Samuel”. To change this path, we need to copy our command into our home directory, and set the path

16. Once we have completed this, we will find that our video has generated.