Running Puffin on Vhub

Introduction

Puffin is a coupling of the programs Puffin (also known as Bent) and Puff. Puffin is a theoretical model of a volcanic plume based on applying the equations of motion in a plume-centered coordinate system. Puff simulates the transport, dispersion and sedimentation of volcanic ash. Puff also prepares some scripts to turn the cdf files generated into images. ImageMagick then creates an animated gif using these images, so we can see how ash spreads over time.

Required Preparations

The tools we use require large databases, too large to be stored on Vhub. For this reason, you will need to mount onto the irods database in order to be able to run Puffin. Additionally, for large runs you have the option of sending larger jobs to the cluster.

First of all, you need to join the **network** group. By default, the hub does not allow outgoing network access. You will need to request an administrator to be added, and then it may take an hour for access to be granted.

To set up irods, look up the topics page for installing irods to get details on how to set up irods. For the purposes of running Puffin, I recommend having the irods directory be in your home directory and to stay logged in as the public user. All necessary databases are in the public folder.

Example Run

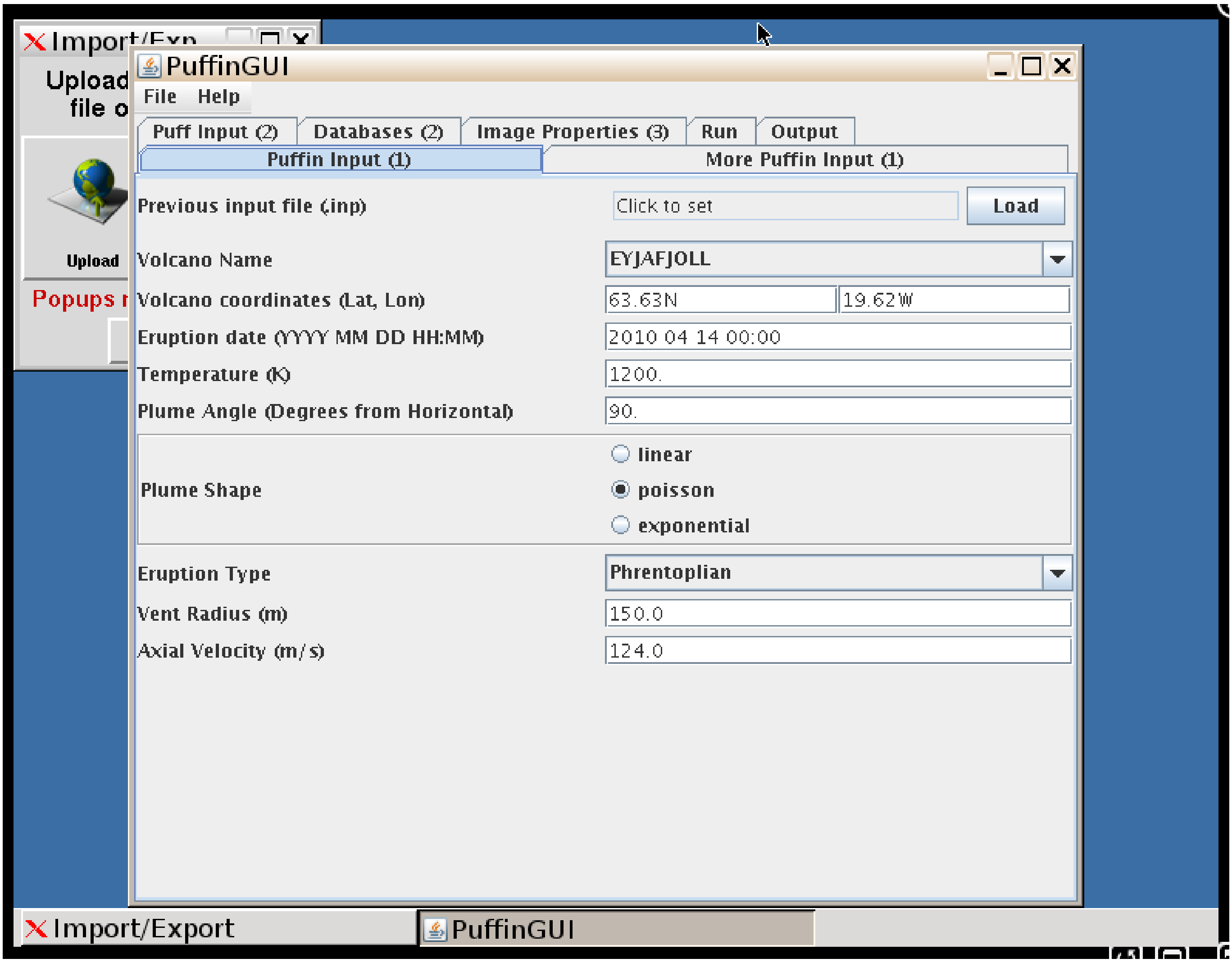
When you start Puffin, most of the entries will start out as filled. The only entries you may need to change for a test run are the locations of the reanalysis databases on the Database tab. If you set up irods to be in your home directory, it will automatically set up the locations in the gui.

In order to run the test run, you just need to press the three buttons on the **run** tab in order. The first button runs puffin, which fills out some information about the plume on the puff input tab. The second button runs puff, which will create the cdf files representing the ashes. The third button creates the animations, which will then display on the **output** tab. You can view these various animations by clicking the buttons on the bottom of the output tab.

The prepared run simulates a volcanic plume in Iceland. The sample runs only 10000 particles over a 72 hour period, so it runs rather quickly. Running puffin should take less than a minute. Running puff should take less than five minutes. Creating the animations should take less than a minute.

Discussing each tab in more detail:

Puffin Input



Previous input file (.inp) Click to set

Volcano Name EYJAFJOLL

Volcano coordinates (Lat, Lon) 63.63N 19.62W

Eruption date (YYYY MM DD HH:MM) 2010 04 14 00:00

Temperature (K) 1200.

Plume Angle (Degrees from Horizontal) 90.

Plume Shape ○ linear ● poisson ○ exponential

Eruption Type Phrentoplian

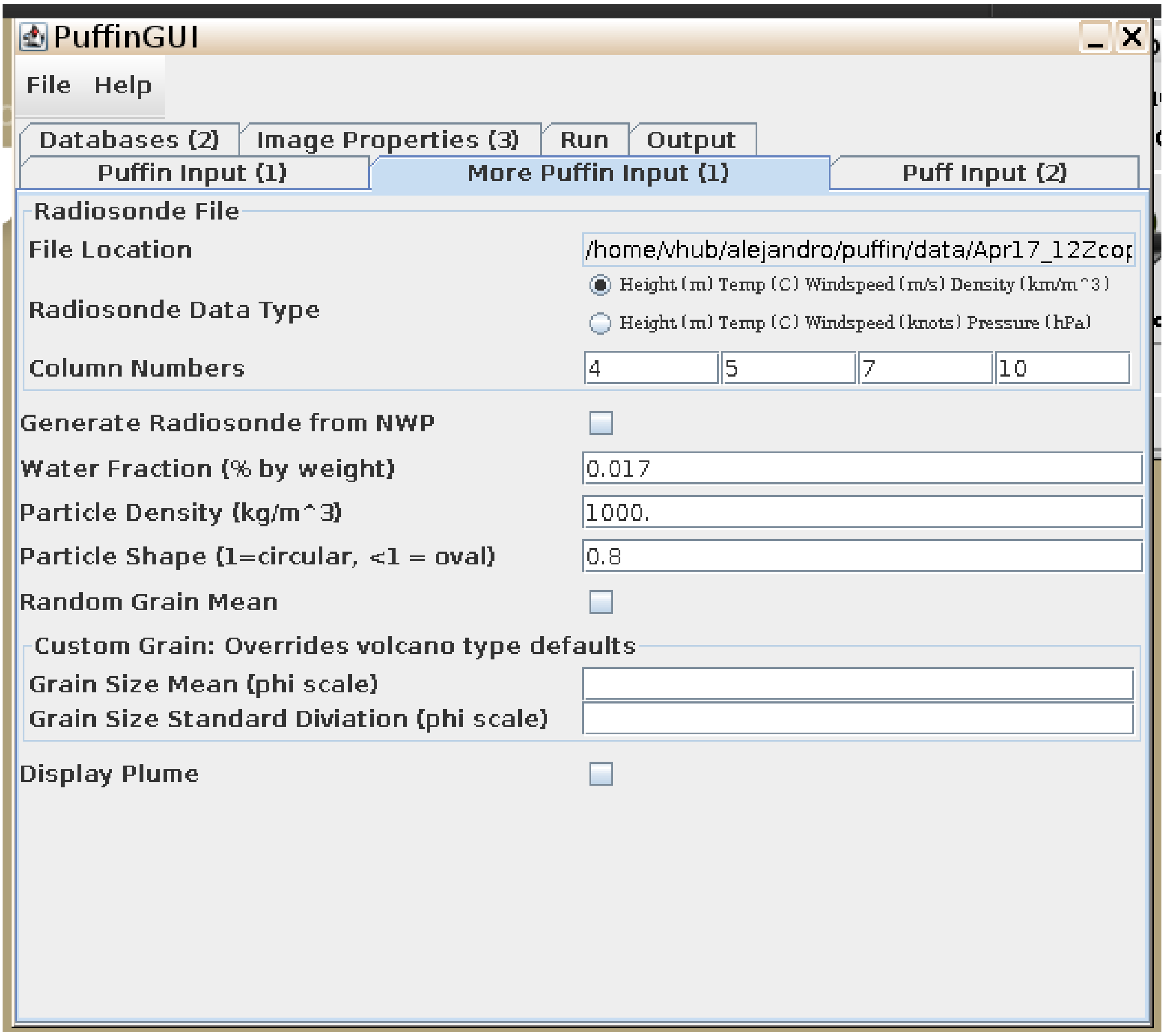
Vent Radius (m) 150.0

Axial Velocity (m/s) 124.0

All the entries here are pretty self expanatory.

If you have the .inp from a previous run, you can load the information from a previous run. Just select the .inp file, and press load. All relevant entries will be updated.

More Puffin Input



Radiosonde File

File Location: /home/vhub/alejandro/puffin/data/Apr17_12Zcop

Radiosonde Data Type:
● Height (m) Temp (C) Windspeed (m/s) Density (km/m^3)
○ Height (m) Temp (C) Windspeed (knots) Pressure (hPa)

Column Numbers: 4 | 5 | 7 | 10

Generate Radiosonde from NWP □

Water Fraction (% by weight): 0.017

Particle Density (kg/m^3): 1000.

Particle Shape (1=circular, <1 = oval): 0.8

Random Grain Mean □

Custom Grain: Overrides volcano type defaults

Grain Size Mean (phi scale):

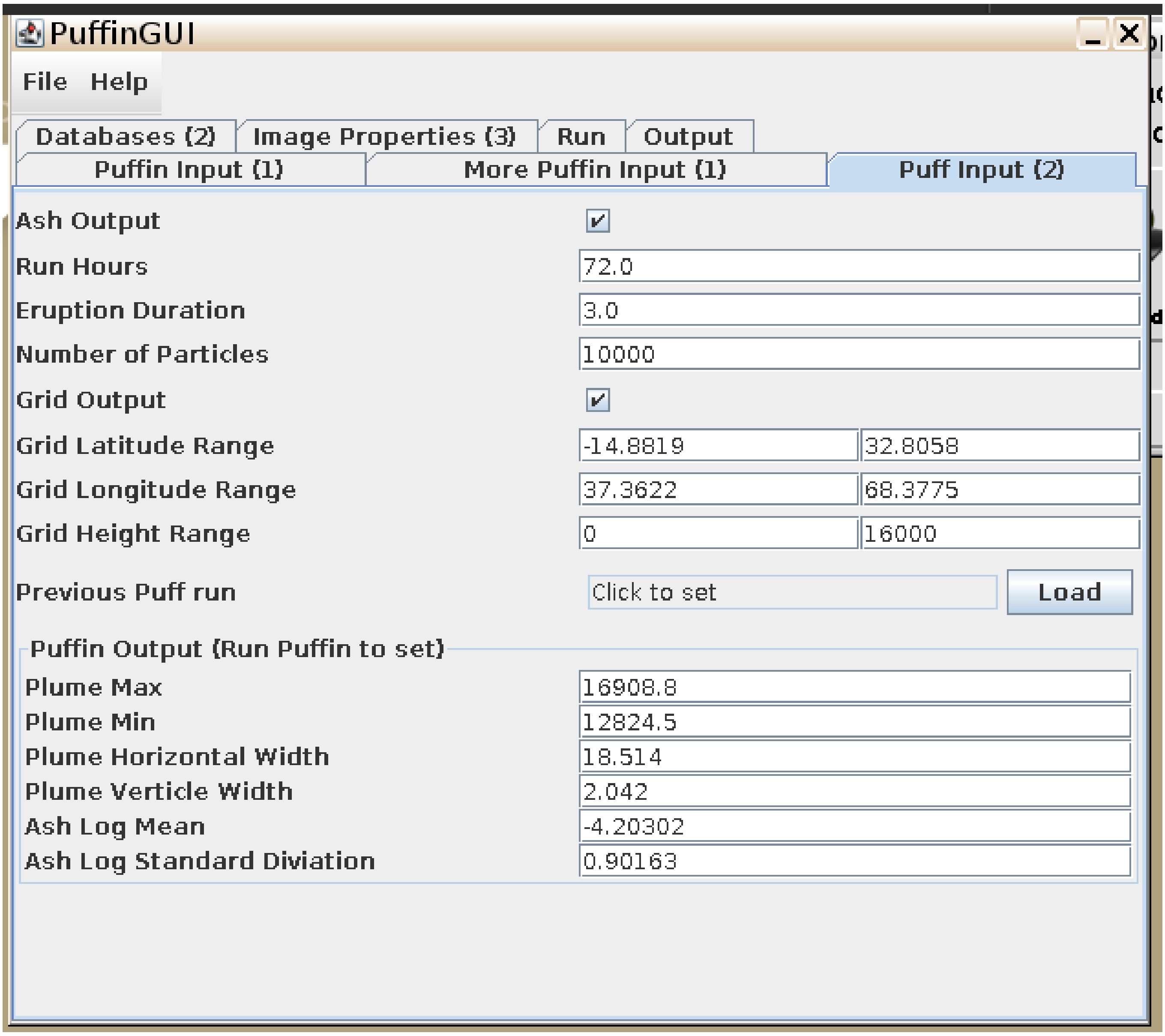
Grain Size Standard Diviation (phi scale):

Display Plume □

In order to simulate how the ash will move up into the atmosphere, you need to give information about the air conditions, given by a weather balloon file. If you have one already, just mark in where the file is, what are the relevant columns, and what the relevant columns represent. If you do not have one, select generate weather file in order to automatically generate one from the databases. Note that this requires all 3 databases from the database tab to be filled out.

Additionally, you have the choice to specify the mean and standard diviation, or just select grain random.

Puff Input



Ash Output	☑	
Run Hours	72.0	
Eruption Duration	3.0	
Number of Particles	10000	
Grid Output	☑	
Grid Latitude Range	-14.8819	32.8058
Grid Longitude Range	37.3622	68.3775
Grid Height Range	0	16000
Previous Puff run	Click to set	

Puffin Output (Run Puffin to set)

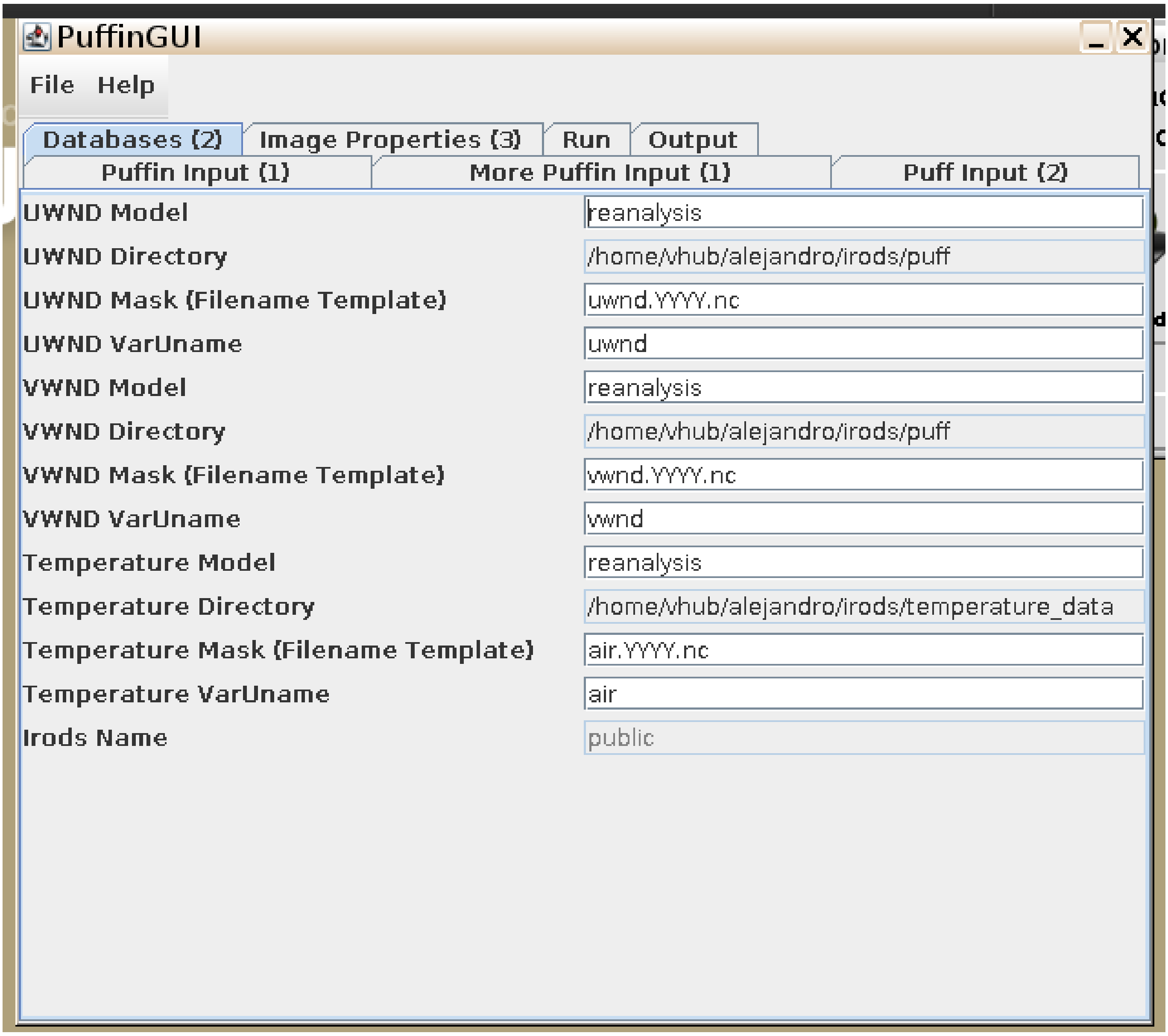
Plume Max	16908.8
Plume Min	12824.5
Plume Horizontal Width	18.514
Plume Verticle Width	2.042
Ash Log Mean	-4.20302
Ash Log Standard Diviation	0.90163

First of all, having ash output selected is very important. With no intermediary ash outputs, no images can be made for the animation.

Second, the more particles you simulate, the more computation time the program will take. Because of this, you are limited in how many particles you can simulation. If running locally, you are capped at 100,000. If you submit to the u2-grid, you are capped at 10,000,000.

Also, note that the bottom six entries will be filled out by running puffin. You do not need to fill them out yourself if you are doing a full run through.

Databases



UWND Model	reanalysis
UWND Directory	/home/vhub/alejandro/irods/puff
UWND Mask (Filename Template)	uwnd.YYYY.nc
UWND VarUname	uwnd
VWND Model	reanalysis
VWND Directory	/home/vhub/alejandro/irods/puff
VWND Mask (Filename Template)	vwnd.YYYY.nc
VWND VarUname	vwnd
Temperature Model	reanalysis
Temperature Directory	/home/vhub/alejandro/irods/temperature_data
Temperature Mask (Filename Template)	air.YYYY.nc
Temperature VarUname	air
Irods Name	public

There are three databases that may need to be set up.

The wind u and wind v databases (which is in miles per hour) are needed to run puff.

The temperature database (which is in Celcius) is only needed for generating a weather balloon file for puffin.

For each database you need to set:

* MODEL- The name of the model you are using.

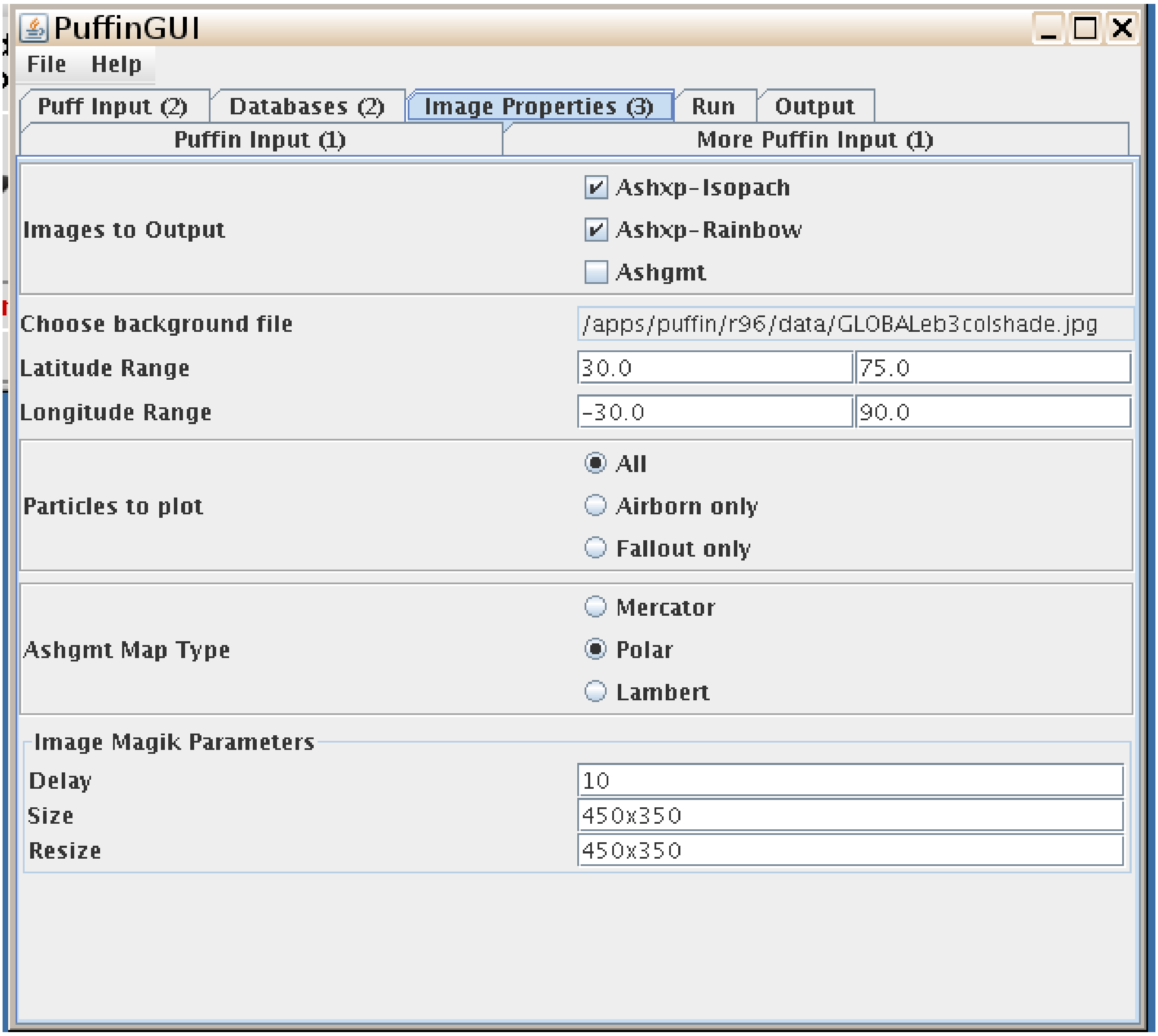
* DIR- The directory where your nc file is.

* MASK- The filenames of the data files. YYYY represents where the year would go.

* VARUNAME- The name of variable within the nc file that we use. This can be found by running ncdump -h on the file.

The irods username is only needed for remote runs. The Irods username is used to convert the directory paths of an irods mount into the directory paths on the cluster.

Image Properties



Images to Output: ☑ Ashxp-Isopach ☑ Ashxp-Rainbow □ Ashgmt

Choose background file: /apps/puffin/r96/data/GLOBALeb3colshade.jpg

Latitude Range: 30.0 75.0

Longitude Range: -30.0 90.0

Particles to plot: ● All ○ Airborn only ○ Fallout only

Ashgmt Map Type: ○ Mercator ● Polar ○ Lambert

Image Magik Parameters

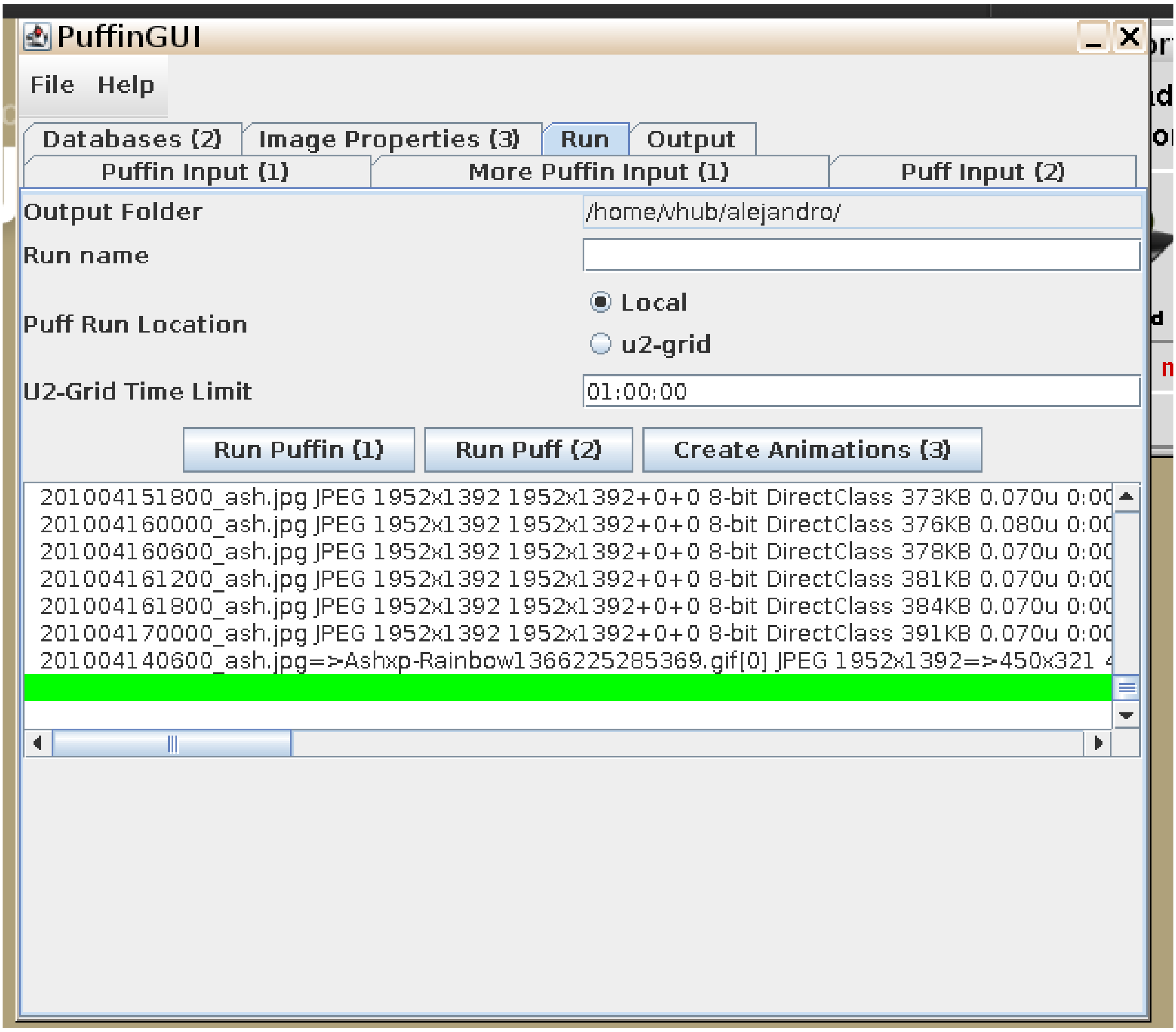
Delay: 10

Size: 450x350

Resize: 450x350

The images to output button selects which images to create. Warning: Ashgmt currently requires some files that are not on Vhub. Ashgmt will not be able to create an image.

Run



Output Folder: /home/vhub/alejandro/

Run name:

Puff Run Location: ● Local ○ u2-grid

U2-Grid Time Limit: 01:00:00

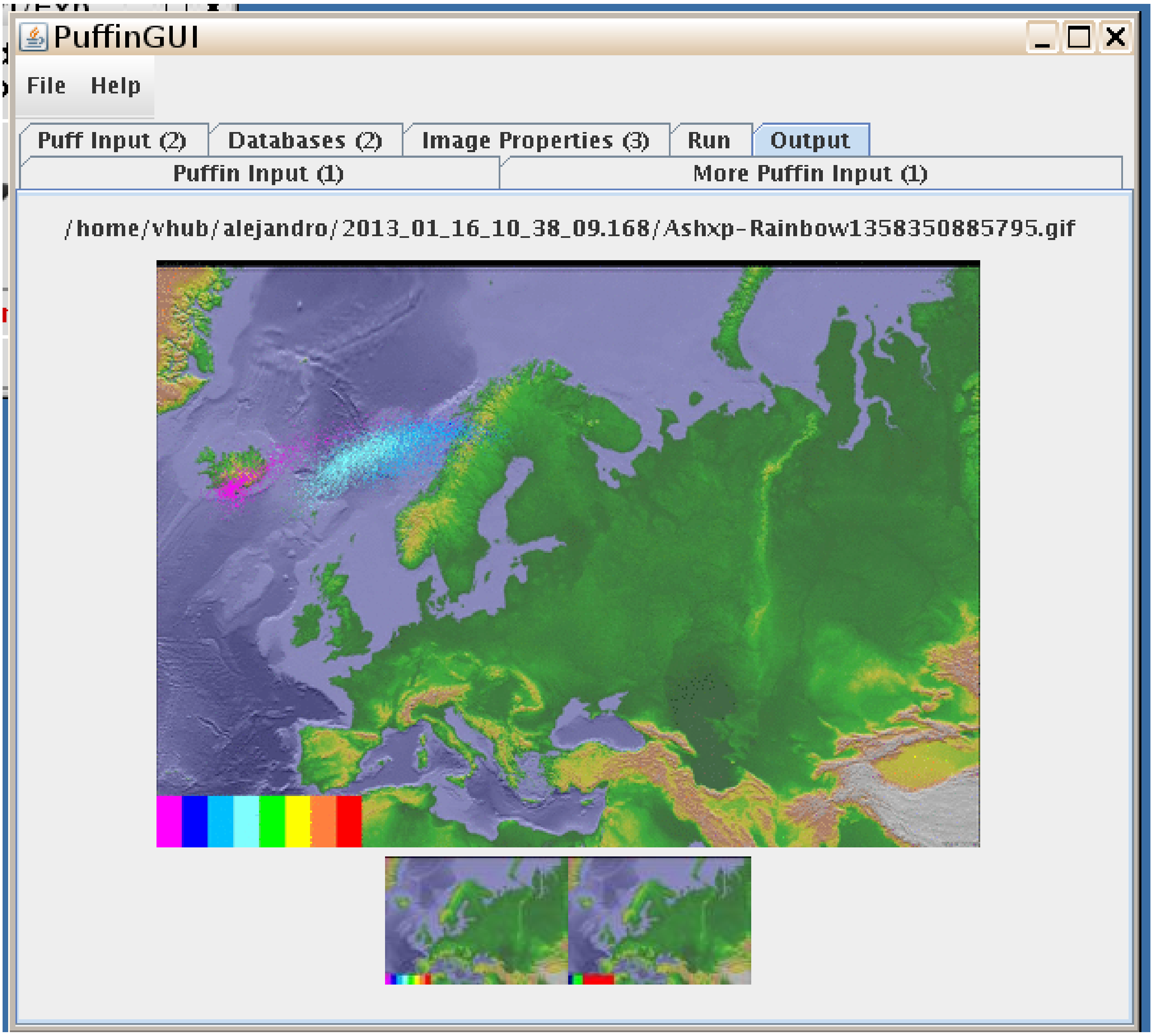
```
201004151800_ash.jpg JPEG 1952x1392 1952x1392+0+0 8-bit DirectClass 373KB 0.070u 0:0
201004160000_ash.jpg JPEG 1952x1392 1952x1392+0+0 8-bit DirectClass 376KB 0.080u 0:0
201004160600_ash.jpg JPEG 1952x1392 1952x1392+0+0 8-bit DirectClass 378KB 0.070u 0:0
201004161200_ash.jpg JPEG 1952x1392 1952x1392+0+0 8-bit DirectClass 381KB 0.070u 0:0
201004161800_ash.jpg JPEG 1952x1392 1952x1392+0+0 8-bit DirectClass 384KB 0.070u 0:0
201004170000_ash.jpg JPEG 1952x1392 1952x1392+0+0 8-bit DirectClass 391KB 0.070u 0:0
201004140600_ash.jpg=>Ashxp-Rainbow1366225285369.gif[0] JPEG 1952x1392=>450x321
```

First of all, you specify where you want to output in the output folder, and whether you want to create a new folder to keep the information in with run name. Every time puffin run, a new folder will be created based on the current time.

Normally, you will want to press run puffin, then run puff, then make animations.

You have the option to send the job to the cluster. Puffin runs (when you hit the first button) will always be done locally. You can send Puff runs (second button) and the animation runs (third button) to the cluster.

Output



/home/vhub/alejandro/2013_01_16_10_38_09.168/Ashxp-Rainbow1358350885795.gif

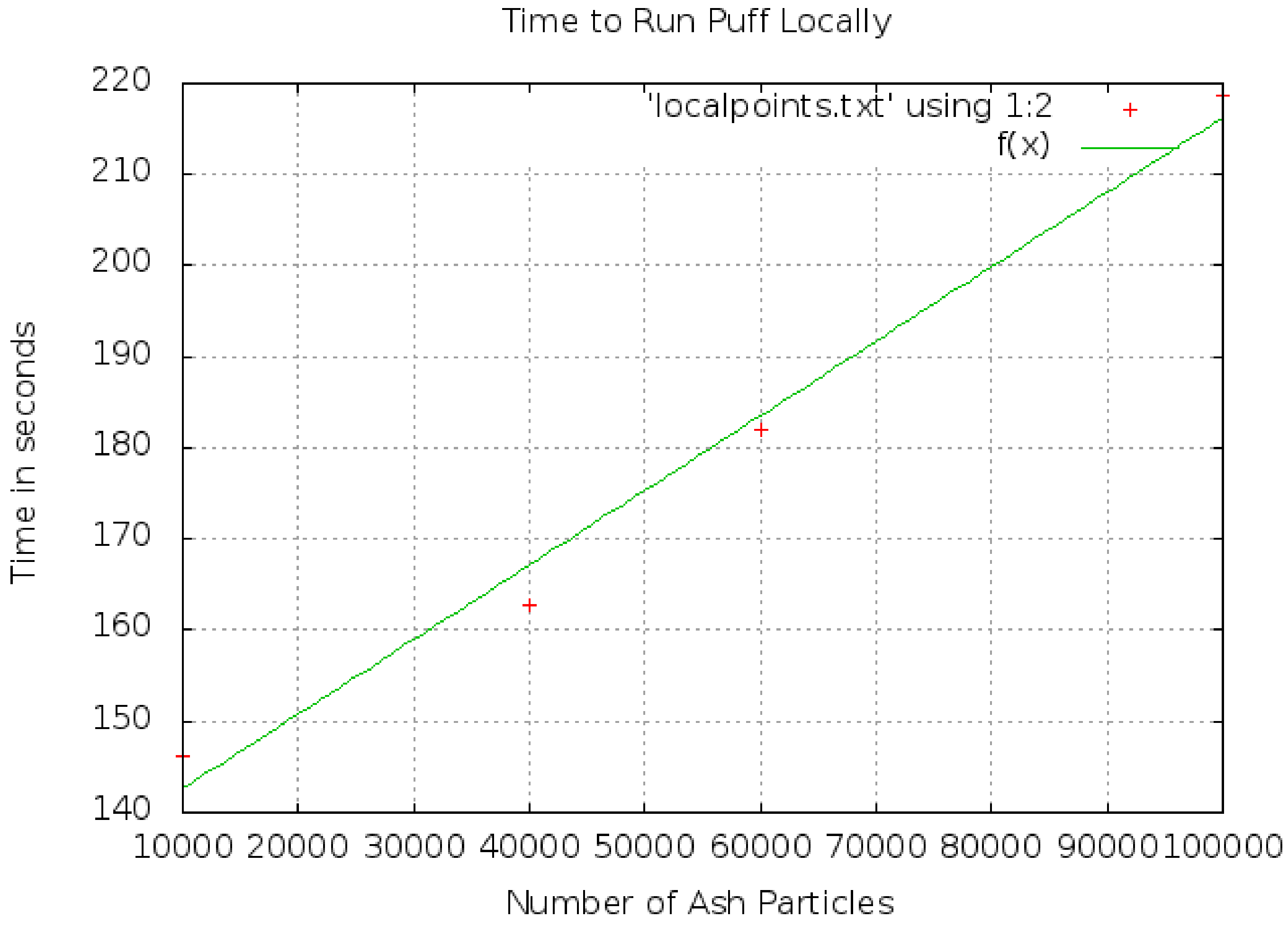
If successful, there will be images on the output tab after make animations is run. Click the thumbnails on the bottom in order to cycle through the images made.

Run Time

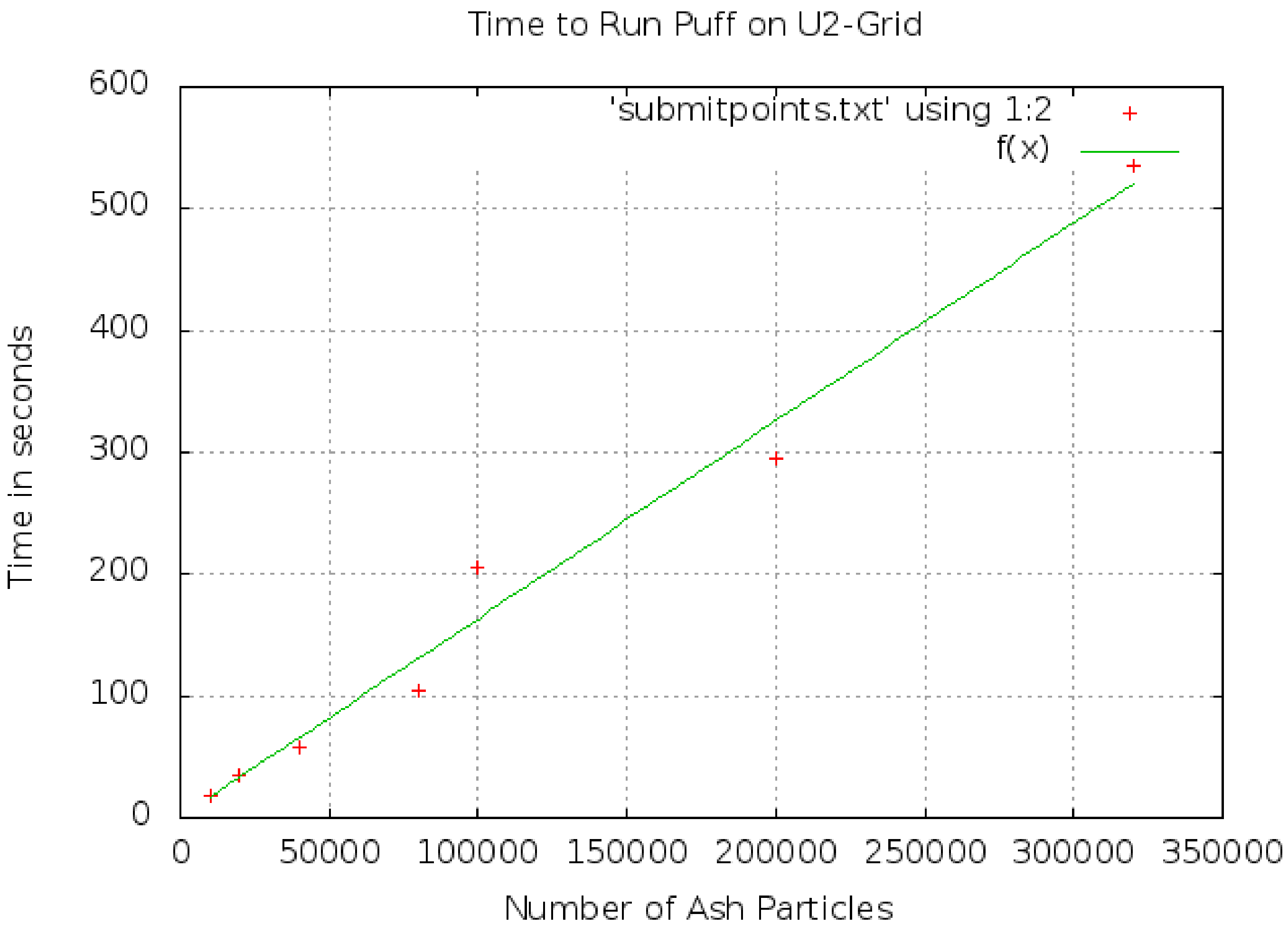
Running Puffin should take seconds, even if you are outputting the plume information.

If you generate a weather balloon file, it will take considerably longer because it has to load the databases. The entire calcuation should take on the order of 4 minutes.

Running Puff is much more computationally intensive, and takes time dependent on how many particles you want to simulate over how much time. You are restricted to only using 100,000 particles locally and 10,000,000 on the hub. Running locally, loading the databases takes about 2 minutes and most computations will take at most 2 minutes. From running several runs, here is a graph of how much time you can expect your runs to take.



When you submit puff to u2, since databases don’t have to be loaded the run will be a lot faster. However, often you will have to wait a couple minutes before your job is queued onto the server. As a result, runs on u2 tend to take just as long as local runs. Here is a graph of the computation time for some larger runs (but note that the time to queue is not included).



Creating the animations should also take on the order of a minute.