Table of Contents

**User Guide1**

Introduction1

GUI2

Start-Up Procedures7

Saving a Genome8

Loading a Genome10

The Params File11

Shut-Down Procedures14

System Error Messages15

System Recovery Guide17

**Introduction**

The WANTool implements an algorithm called NEAT (Neuro Evolution of Augmenting Topologies). This algorithm turns the neurons and synapses of a regular artificial neural network into genes which can be modified, evolved, and combined. The WANTool uses this algorithm to make predictions based on a set of inputs. The WANTool uses data that would be commonly used in regressions and attempts to simulate the predictions created by regression analysis. The goal of the WANTool is to achieve lower variance in predictions then traditional regression analysis.

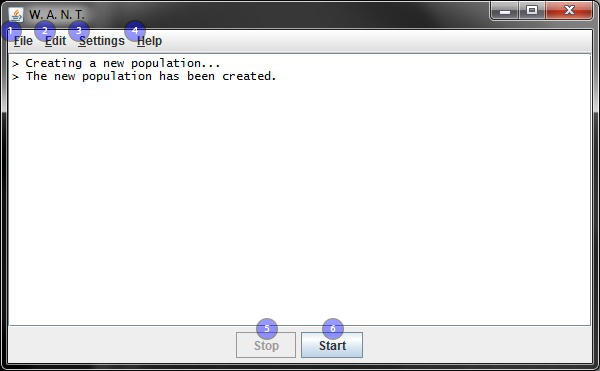
The main function of the WANTool is to take in a set of data with inputs and known outputs and use the algorithm described above to learn how the inputs correlate to the outputs. Once the network has been trained, the genome can then be loaded into the predictions console which will take in a dataset with unknown outputs and make predictions of the data. For example, a dataset describing the factors that affect whether someone will graduate college or not can be fed into the network. The data that might be fed into the network might be a student’s HS GPA, ACT score, distance from home, hours spent studying a week, etc. and then the network would train itself based on if it correctly predicts the given outputs, which in this case would be whether or not the student graduates. Once the network was trained, data about current students at a university could be fed into the prediction algorithm along with the trained network and predictions could then be made about those students. The output from the network could then be used to provide more assistance to those who are more likely to not graduate. The WANTool can be applied to many domains such as stock price predictor, starting salaries for new hires, college retention rates, etc.

Though the WANTool is very flexible and can be applied to many different domains, the tool is not without its limitations. One of the main limitations of the system is the number of inputs/outputs the network can handle. The WANTool takes a lot of computational power to run and the amount of computation needed to handle an additional input increases rather quickly. The recommended number of combined inputs and outputs is not more than 20-25, which in most cases is more than adequate to make good predictions.

**Graphical User Interface**

The GUI is how the user interacts with the WANTool. From the GUI, the user can do several things such as load a genome, start a new genome, or load the parameters file. These are only a few of the things a user can do from the GUI. The GUI will be covered in depth below.

**Main Console**

****

1 – File – From the File menu, the user can start a new population, open a saved population, save a population, print what is in the console, or exit the system.

2 – Edit – From the Edit menu, the user can start and stop training, as well as copying the console to the clipboard or clearing the console.

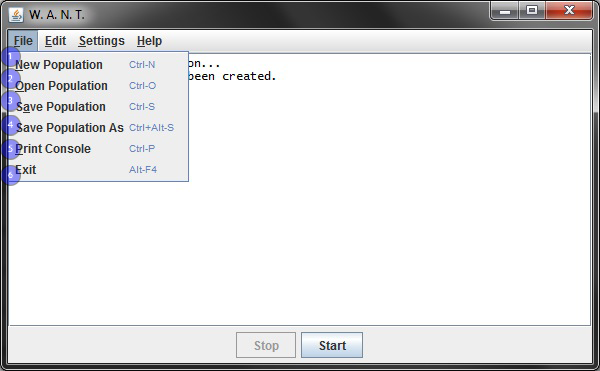
3 – Settings – From the Settings menu, the user can load a different parameters file or test the predictions.

4 – Help – From the Help menu, the user can access the User Manual and the System Manual, as well as information about the software and where to go for more help.

5 – Stop – If there is a population currently training, clicking Stop will pause the training, allowing the user to save the current population or safely exit the system.

6 – Start – Clicking start will cause the system to start training the currently loaded population. If there are no populations loaded, it will begin training a new population.

**File Menu**



1 – New Population – When the user selects New Population, any current populations are cleared (if any), and a completely new population is established to train.

2 – Open Population – This feature allows the user to load a saved population. Then, the user can continue training that population.

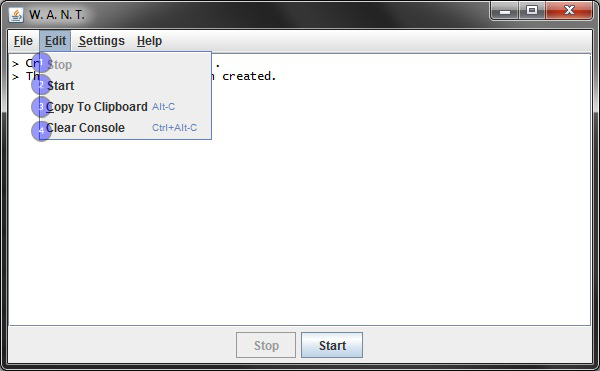
3 – Save Population – This feature allows the user to save a population at any point in its training, so that the saved state can be called at a later point in time. If the population has already been saved, this feature will overwrite the old population and replace it with the new.

4 – Save Population As – This feature performs similarly to Save Population, but will allow the user to save populations with previous save files to a new file.

5 – Print Console – This feature allows the user to print the current state of the console.

6 – Exit – Selecting Exit will terminate the system and shut it down.

**Edit Menu**



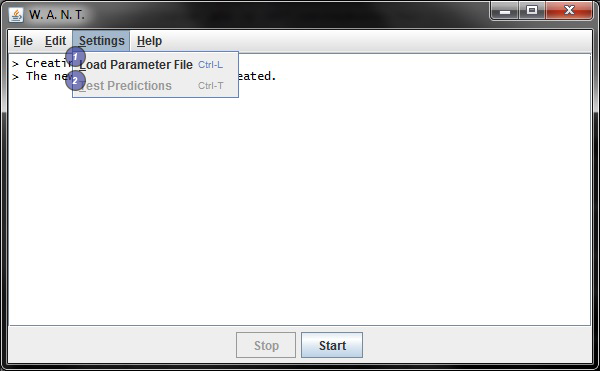
1 – Stop – This feature is exactly the same as the Stop button at the bottom of the window. If there is a population currently training, selecting Stop will pause the training, allowing the user to save the current population or safely exit the system.

2 – Start – This feature is exactly the same as the Start button at the bottom of the window. Selecting start will cause the system to start training the currently loaded population. If there are no populations loaded, it will begin training a new population.

3 – Copy to Clipboard – This feature allows the user to copy the console to the clipboard, which can then be pasted into other locations.

4 – Clear Console – This feature will clear the console of all training and data.

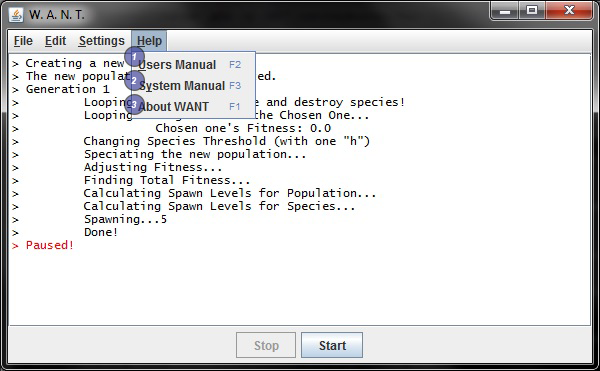
**Settings Menu**



1 – Load Parameter File – This feature allows the user to load a parameters file and utilize the altered values for the training. This cannot be changed after you start training.

2 – Test Prediction – This feature allows the user to compare the results of the current population against collected data to test the quality of the training.

**Help Menu**



1 – User Manual – Selecting this feature will open the User Manual. This feature is unavailable while training is running.

2 – System Manual – Selecting this feature will open the System Manual. This feature is unavailable while training is running.

3 – About W.A.N.T– Selecting the About W.A.N.T. feature will open a pop up with information about the software and where to go for more help. This is the only Help feature available while training.

**Start-Up Procedures**

To start the system:

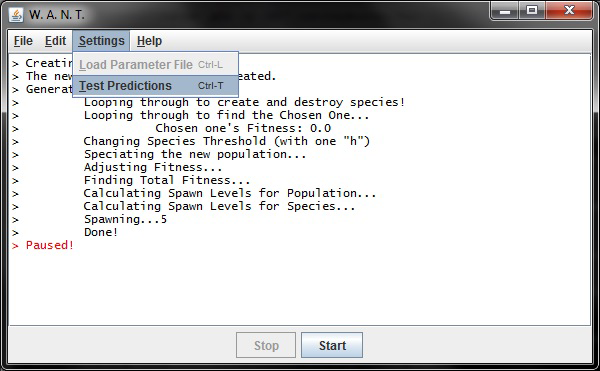
**WINDOWS**: Open the file named WANT.exe

**MAC or LINUX**: Open the file named WANT.sh

If you do not wish to use the default parameters, you must go to the Settings menu and select Load Parameter File and select the file with the parameters you wish to use. If you want to use the defaults, you can proceed to train by either pressing Start to create and train a new population or load a saved population by going to the File menu, selecting Open Population, and then selecting the desired population.

**Loading Predictions**

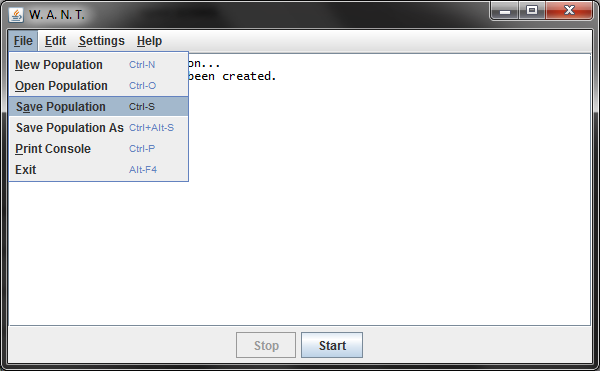
To load the predictions, first load a genome which is outlined below. Once you have the genome loaded, go to setting and click on test predictions and select your predictions file.



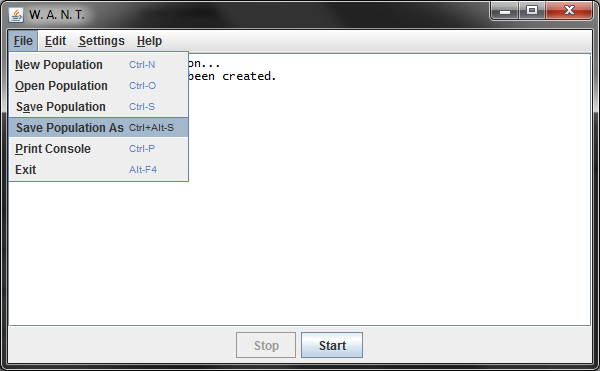
**Saving a Genome**

To save a population, go to the File menu and select Save Population or Save Population As. If it is a new population, you will be prompted to navigate to the desired directory and name the population as desired. Also, the system will automatically save the current population based on the value in the params.gapf file. The default value is 12 generations.

If the population was loaded from a saved population and you select Save Population, it will overwrite the previous population and save the new population under the same name.

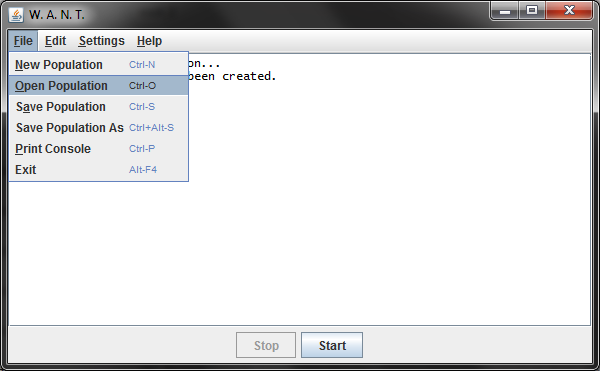


If the population was loaded from a saved population and you want to save the current population but still want to keep the old population, select Save Population As. You will then be prompted to enter the name for the new file.

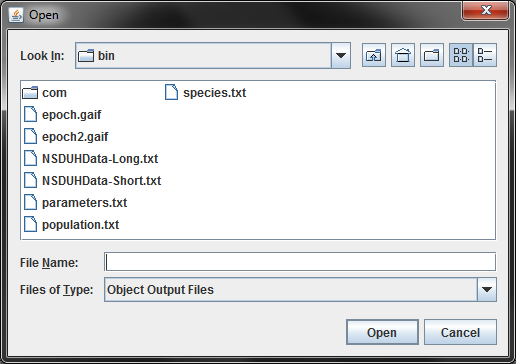


**Loading a Genome**

To load a saved Population, go to the File menu and select Open Population.



Then navigate to the directory containing the saved population, and open the file.

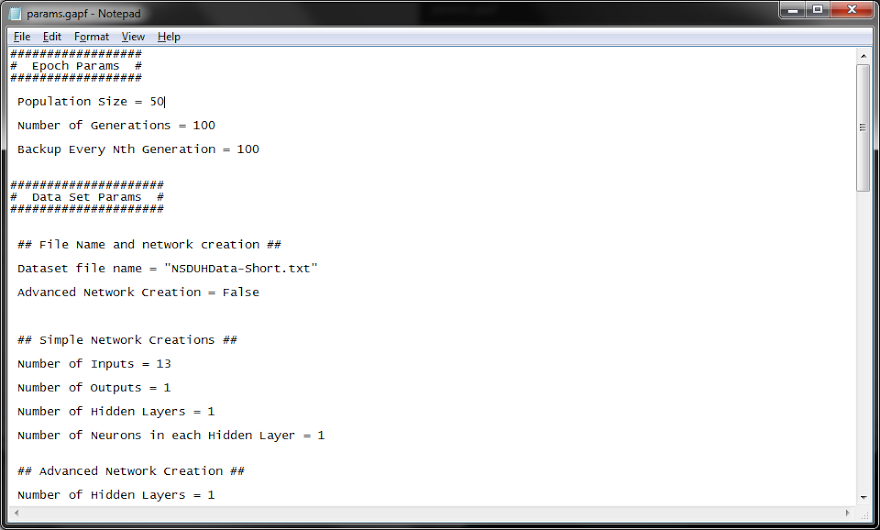


**The Params File**

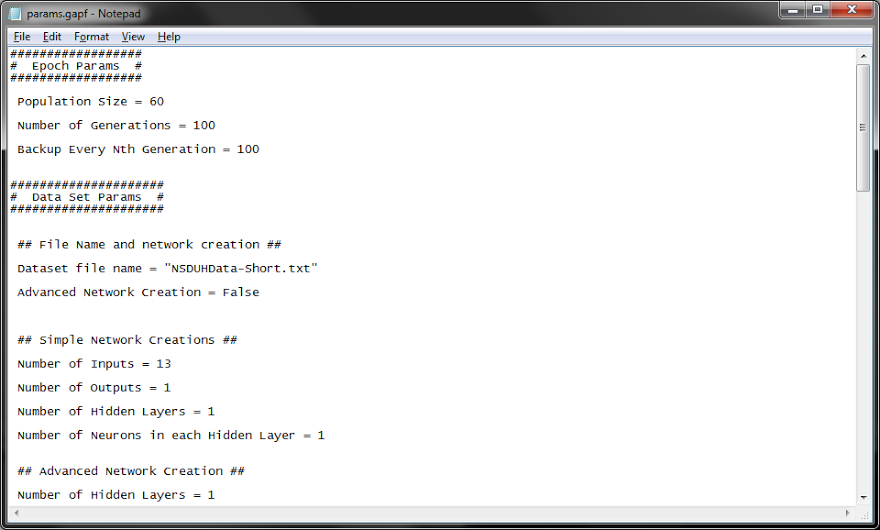
The file params.gapf contains the default settings to train the populations. If you want to alter some aspect of the training, you can do so through this file. It is recommended that you make a copy of the original file and keep it untouched so as to always have the option to return to the default settings. When changing parameters, be absolutely certain not to change the parameter names nor the order of the parameters. The only aspects that should ever be changed are the values themselves. Changing anything else will cause the system to be unable to find the value and crash. The values should be replaced with values of the same type- numbers with numbers, true with false, etc. Again, do not change anything but the values of the parameters. When saving the altered parameters, save them with a “.gapf” extension. For more information on the effects of each parameter, see “Modifying the Genome” in the System Manual.

Example:

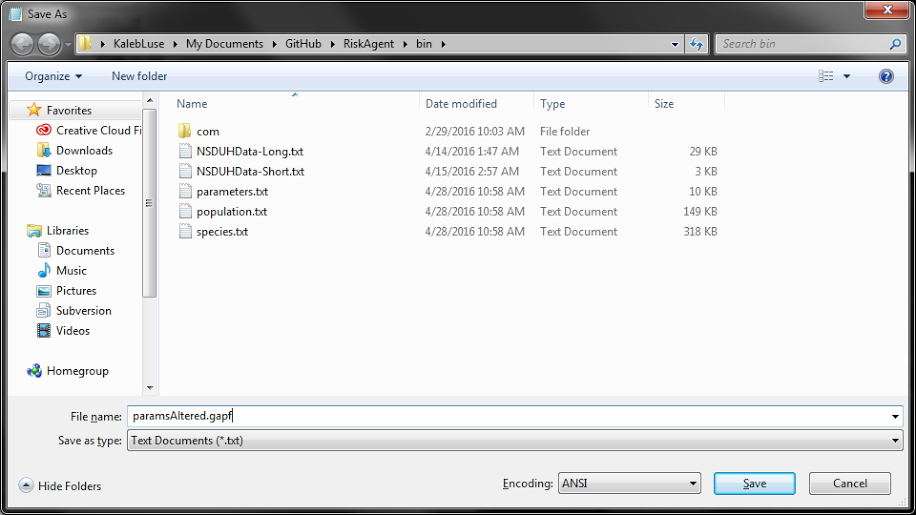
Here is the default Epoch section of params.gapf



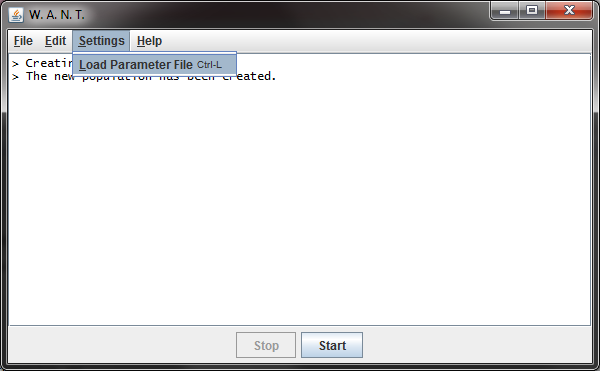
Notice that the population size is set to 50. Let’s change that to 60.



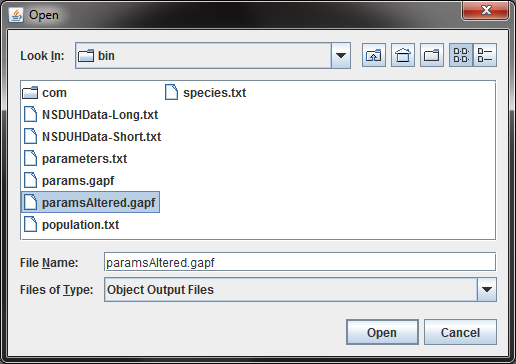
Now that it is changed, we will need to save it. Let’s call it paramsAltered.gapf.



With the changes saved, we now need to load it into the WANT system



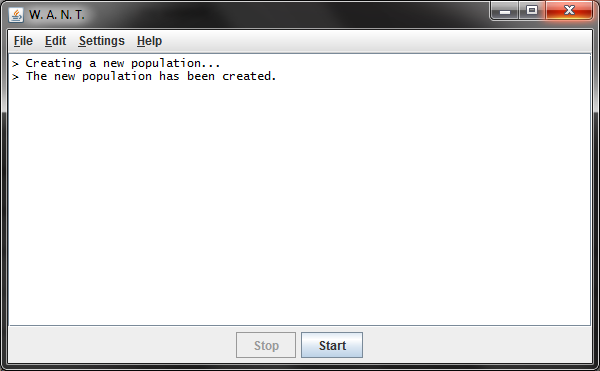
Find the file and open it

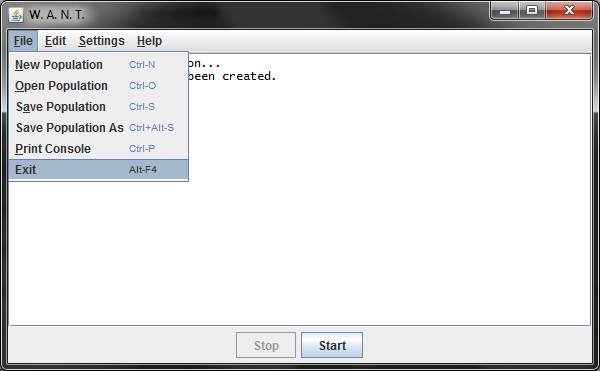


The system will now implement the newly changed parameter!

**Shut-Down Procedures**

To shut the system down, you can simply press the exit button in the top right corner or select Exit in the File menu.





**System Error Messages**

***Genetic Algorithm***

**RuntimeException**

* If the link sent in is disabled, or contains corrupt/invalid metadata.
* If the population size is invalid.
* If the generation number is invalid.
* If the back-up number is invalid.
* If the Advance Creation parameter is not set.
* If the Input Neuron number is invalid.
* If the Output Neuron number is invalid.
* If the Hidden Neuron number is invalid.
* If the Add Neuron parameter is not set properly.
* If the Add Link parameter is not set properly.
* If the Add Looped Link parameter is not set properly.
* If the Change Neuron Type parameter is not set properly.
* If the Mutate Bias Weight parameter is not set properly.
* If the Mutate Input Neuron parameter is not set properly.
* If the Mutate Neuron parameter is not set properly.
* If the Mutate Link parameter is not set properly.
* If the file does not exist
* If there is an error with the file.

**FileNotFoundException**

* When the file cannot be found

***Neural Network***

**ExceededNeuronInputException**

* When the amount of inputs has already been met.

**InvalidNeuronInputException**

* When the amount of inputs given is greater than that of the amount of Neurons in the NeuralNet, or when the amount of inputs given is greater than that of the input layer.
* When the amount of inputs given is not equal to the amount of weight minus one (for the bias weight), this Exception is thrown.
* When the index has already been populated.
* When the index is greater than the size of the array holding the inputs or less than zero.
* If the weight length is not the same as the original weight length.
* If the index is greater than or equal to the length of the weight array.
* If the amount of adjustments to be made is not equal to the amount of weights found in this Neuron, including the bias weight.
* If the index is greater than the length of weights given.

**InvalidNeuronOutputException**

* If the given size of outputs is not equal to that of the neural network's output layer.

**NeuronCannotFireException**

* When the Neuron cannot fire, as in it doesn't have enough inputs to fire yet.

***Util***

**IOException**

* When the file cannot be found.

When File IO is interrupted by an external program

**System Recovery Guide**

In the event of some kind of disaster it is nice to have procedures in place to recover from these disasters. The WANTool makes recovery easy as long as the user has not disabled the recovery feature which is explained in depth in the above section about how to modify the params.gapf. Located inside the params.gapf file is a parameter called Backup Every Nth generation. To disable this feature set the parameter to 0, but to enable it set the parameter between 50 and any value. What this feature does is allow the user to specify how often the genome is saved so that should something happen with the system while it is running, not all data will have been lost.

Power failure: In the event of a power failure, not all is lost. If the user has specified in the params.gapf a value for the backup every Nth generation, then to recover from a power failure all the user has to do is restart the console, and load the last saved genome which should be located in the epoch.gaif. There exists the potential that some data could still have been lost between generations but reloading the last saved genome will prevent the user from losing too much data and should provide a good backup point to recover from. The user should keep this in mind when setting the backup parameter and determine how many generations they can afford to lose. If they are running regression data that take hours to get through 50 generations then it may be useful to set this parameter at a lower number but if the WANTool is creating a new generation every second then the user may decide to set the backup to a larger number. It is recommended to set the parameter to save every 5-10 minutes.

Software failure: There are many things that can constitute a software failure and the WANTool tries its best to mitigate any unnecessary loss of data. The worst case scenario is similar to the situation encountered in a power failure. Simply restart the console and load the last saved genome. The amount of data lost is dependent on the backup parameter explained in the opening paragraph of the recovery section.

File Corruption: The only way to properly recover from a corrupt data file is to follow the proper backup procedures listed above. It is important to maintain copies of the genome object files in case something catastrophic were to occur with the system. To load from backups, simply navigate to the folder containing the back .gaif files and load the backup you wish to start from.