

Creatures Development Ressources

<http://double.co.nz/creatures/>

Table des matières

1	Genetics	3
1.1	Genes	3
1.2	Reference Tables	3
2	Brain Lobes	3
2.1	Overview	3
2.2	Brain Lobe Genetics - Genes' Header	4
2.3	Cell Body Page	5
2.4	D0 Growth Page	6
2.5	D0 Dynamics Page	7
2.6	Dendrite Overview	8
2.7	Issues	9
2.8	Perception Lobe Tutorial	10
3	Chemical Emitter	13
3.1	Overview	13
3.2	Dialog Box	13
3.3	Breakdown – <i>Gene Header</i>	13
4	Chemical Receptor	14
4.1	Overview	14
4.2	Dialog Box	14
4.3	Breakdown	14
5	Initial Concentrations	15
5.1	Overview	15
5.2	Dialog Box	15
5.3	Breakdown	15
6	Creature Instinct	15
6.1	Overview	15
6.2	Dialog Box	16
6.3	Breakdown	16
6.4	Notes	17
7	Creature Stimulus	17
7.1	Overview	17
7.2	Dialog Box	17
7.3	Breakdown	18
8	Brain Lobes – <i>Creatures 1</i>	19
8.1	Overview	19
8.2	Lobe 0 – Perception Lobe	20
8.3	Lobe 1 – Drive Lobe	21
8.4	Lobe 2 – Stimulus Source Lobe	21
8.5	Lobe 3 – Verb Lobe	23
8.6	Lobe 4 – Noun Lobe	23
8.7	Lobe 5 – General Sensory Lobe	23



8.8	Lobe 6 – Decision Lobe	24
8.9	Lobe 7 – Attention Lobe	25
8.10	Lobe 8 – Concept Lobe	25
8.11	Lobe 9 – Regulator Lobe	26
8.12	What are Dendrites?	27
8.13	State Variable Rules – Overview	27

Liste des tableaux

1	Lobes and perceptible settings	16
2	Norn Lobes for <i>Creatures 1</i>	19
3	Norn Lobes for <i>Creatures 2</i>	20
4	Copied Lobes in Perception	20
5	Drive Lobe Data Entries <i>Drives</i>	21
6	Source Lobe Data Entries	22
7	Verb Lobe Data Entries	23
8	General Senses Lobe Data Entries	24
9	Regulator Lobe Loopbacks	26
10	List of SVRules for Creatures 1 and Creatures 2	29

Table des figures

1	General Interface for Brain Genetics Design	4
2	Cell Body Interface for Brain Genetics Design	5
3	D0 growth Interface for Brain Genetics Design	6
4	D0 dynamics Interface for Brain Genetics Design	7
5	Impact of original height and with on original connections.	10
6	Proximity connection of cells.	10
7	General Senses.	10
8	Brain Cell Monitor — Example 1.	11
9	Brain Cell Monitor — Example 2.	11
10	Interface for Chemical Emitter Design	13
11	Interface for Chemical Receptor Design	14
12	Interface for Initial Concentration Design	15
13	Interface for Intincts Design	16
14	Interface for Stimulus Design	17
15	Creatures 1 Brain Map.	20
16	Creatures 2 Brain Map.	20



1 Genetics

The behavior of a norn and its interaction with the environment within Creatures is controlled by its Digital DNA. This DNA code can be modified using the Creatures Genetics Kit. The Genetics Kit is the official means of editing a norns genome but there are various third party tools available as well. See the links page for other sites that may contain such utilities.

While this documentation concentrates mainly on the Norn brain lobes there are a number of other genes that are modifiable via the Genetics Kit. The online help in the Genetics Kit is not very helpful when it comes to descriptions of the individual fields within each gene dialog box. Follow the links below to get information on the various fields for each gene dialog in the Genetics Kit based upon information I've gathered on the net and experiments I've performed.

1.1 Genes

- Brain Lobe
- Chemical Receptor
- Chemical Emitter
- Chemical Reaction
- Chemical Half-Lives
- Chemical Initial Concentrations
- Stimulus
- Genus
- Appearance
- Pose
- Gait
- Instinct
- Pigment
- Pigment Bleed

1.2 Reference Tables

- State Variable Rules
- Cell List connection
- Brain Map and Models

The information presented here was written for Creatures 1 but all the information is still valid for Creatures 2 genetics. I have not yet updated the screenshots of the genetics kit for Creatures 2 but not much new was added that needs explanation.

2 Brain Lobes

The brain lobe is one of the most complicated portions of norn genetics. The Cyberlife Genetics Kit provides the ability to add, delete or modify brain lobes. The dialog box for modifying brain lobes contains a bewildering number of options. The purpose of this brain lobe discussion is to describe what each of these options does. Note 08/10/1999 : The information on brain lobes below was created for Creatures 1. All the information is still valid for Creatures 2 genomes, but the screen shots show the Creatures 1 genetics kit.

The descriptions of the brain lobe settings shown here have mostly been discovered through experimentation. The tutorials available in another section describes how most of the information was found. The purpose of this section is to provide a single place where all the information gathered through the tutorials and other places can be placed for easy use. It will be updated frequently as I discover new things.

2.1 Overview

Most standard norns have nine brains lobes. A number of the lobes are 'special' in that they have a particular function hard-coded within the Creatures executable. A limited amount of genetic modification can be done to these lobes as most of their functionality is provided through the Creatures system. The other lobes are completely genetically defined and this is where some of the most interesting modifications to a Norn can be made.

A brain lobe sits within a 64x48 grid of neurones (often referred to here as cells). A lobe has a location within this brain grid described as an (X, Y) coordinate and (width, height). Each lobe therefore has a set size measured in neurones. Each neurone in a lobe has a numeric value associated with it known as the 'state' represented as a number from between 0 and 255 inclusive. The current value of the 'state' of a neuron indicates the level of activity. The value of the neurone 'state' can be set via the Creatures executable directly, CAOS macro code, or via dendritic links from other cells in other brain lobes. In this manner a network of cells and lobes is created that ultimately forms the 'brain' or neural net of the norn.

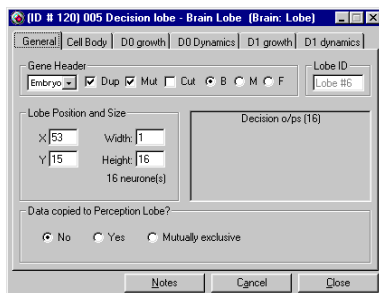
All neurones within a single lobe behave in a similar manner to perform some function. Each lobe has genetically defined information associated with it that affects the neurones in that lobe. So a brain lobe is really a group of neurones that perform in a similar manner.

A detailed description of the genetically defined information associated with each lobe follows. This is the information that was obtained through experimentation. After this a brief description of the functionality of the brain lobes in a standard norn is provided. This document will conclude with some ideas and thoughts on future brain modifications that could be made to improve or modify a norns behavior.

2.2 Brain Lobe Genetics - Genes' Header

The Cyberlife Genetics Kit has a 'tabbed' dialog box for modifying the genetic information associated with a brain lobe. Each page of the dialog box holds information about the lobe and the breakdown below will be done page by page as defined by the genetics kit.

General Page



The example dialog given here is from the Decision lobe of a Ron norn. The 'general' page provides the standard gene header information that describes when the gene switches on, how mutations affect it and the required sex of the norn for the gene to switch on. Also included is information on the placement of the lobe within the brain grid, a description of the lobe, a unique numeric identifier for the lobe and an indication as to whether lobe data is copied to the perception lobe.

FIGURE 1 – General Interface for Brain Genetics Design

Gene Header

Embryo : This field indicates the age at which the brain lobe gene will switch on. Most brain lobes turn on at the 'Embryo' stage to ensure they operate correctly as soon as the norn is born. I know of no lobes that are turned on at different ages and I don't know if this functionality even works. This has been added to the list of issues to explore.

Dup : If this flag is checked then the lobe can be duplicated as a genetic mutation. That is, multiple copies of this lobe will be able to exist due to mutation.

Mut : Sets whether genetic mutations can be applied to the values contained within this brain lobe.

Cut : If set the brain lobe can be removed during breeding or as the result of mutation. Most brain lobes have this unchecked as a norn without any of the standard lobes would not be able to function.

B/M/F : Defines the gender that this brain lobe will switch on. This could potentially be used to create norns that have different intelligence levels or behaviour depending on the gender of the norn. I haven't tested if this functionality actually works. This has been added to the list of issues to explore.



Lobe Id

Lobe # : Contains the numeric identifier for the brain lobe. This is assigned automatically by the genetics kit and cannot be changed. The first 8 numbers (0 through to 7) correspond to specific lobes that are hard-coded within the Creatures executable. If you number a lobe with one of these hard coded numbers (using a hex editor or such like) then Creatures may stomp over your neurone values at any time so be warned.

Lobe Position and Size

X : Contains the X coordinate of the lobes location within the brain grid. I don't know what the effect of overlapping brain lobes will be. Added to the issues list.

Y : Contains the Y coordinate of the lobes location within the brain grid.

Width : Contains the amount of neurones in the X direction that the lobe uses within the brain grid.

Height : Contains the amount of neurones in the Y direction that the lobe uses within the brain grid.

The total number of neurones contained within the lobe is calculated as (Width * Height).

Description

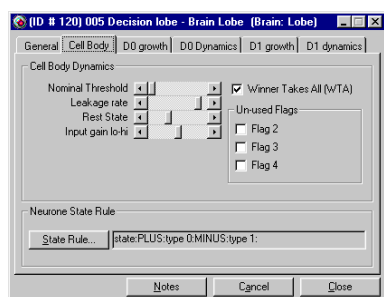
This contains a description of the functionality of the brain lobe and is not editable. I think only the standard brain lobes have descriptions associated with them. All additional lobes get a default like 'Interneurone lobe'.

Data Copied to Perception Lobe

No : None of the information in this brain lobe will be duplicated in the perception lobe (lobe 0).

Yes : Every neurone within this brain lobe will have its state value duplicated in the perception lobe. See the perception tutorial for details.

Mutually Exclusive : A lobe marked as mutually exclusive affects the way dendrites are formed between the perception lobe and the concept lobe. If a lobe is marked as mutually exclusive then only one cell from this lobe may contribute to the forming of a particular concept in the concept lobe. See the perception tutorial for details.

2.3 Cell Body Page

The 'Cell Body' page provides information associated with each neurone (otherwise known as 'cell') within the brain lobe. It is here that the behaviour of the neurones are defined. Much of the information obtained about this page is from tutorial one and tutorial two.

FIGURE 2 – Cell Body Interface for Brain Genetics Design

Cell Body Dynamics

Nominal Threshold : This is a number with a value from 0 through to 255. When the state is set for a neurone it will not fire unless the value of the neurone state is greater than this threshold. The value of the neurone after firing will be state minus threshold.

Leakage Rate : This defines the speed at which the state will drop from its current value to its rest state. It has values like 'Instantly', '5 Seconds', right through to '52 Years' and is represented by a number from 0 to 255.

Rest State : This is the resting state of the neurone. That is, if the neurone has not been fired or activated it will sit on the value set here. The value is from 0 through to 255.

Winner Takes All : If this flag is checked then the lobe is a winner-takes-all lobe. This means that after all the neurone values for the lobe are calculated the highest firing neurone (highest state value) is the only one that fires and all the other neurones have their output value set to zero. An example is the decision lobe where only one decision can be made.

Neurone State Rule

An SVRule is like a miniature program written in a special programming language, which has a number of 'opcodes' or operations that it can perform on various pieces of data. Only eight individual opcodes are allowed in an SVRule making it very small and fast to execute - the SVRule for every cell in the brain must execute 10 times per second! – **State Rule** : The SVRule defined here is used to calculate the new state value of a neurone. The result of the SVRule becomes the state value of the neurone. In this way the state for neurones within lobes can be defined genetically and different calculations can be performed depending upon the task of the lobe. The SVRule possibilities are huge and can be of a great complexity. Remember that SVRules are used in a number of places in a brain lobe but the specific one here is only used to calculate the value of each neurone 'state'.

2.4 D0 Growth Page

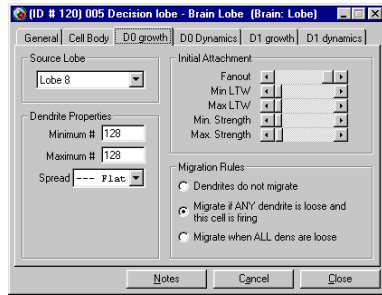


FIGURE 3 – D0 growth Interface for Brain Genetics Design

This is one of the first pages that describes the behaviour of dendrite connections in a lobe. Dendrites are to the brain lobes as wiring is to integrated circuits. They connect the neurones in different lobes and allow information to travel between them. Each brain lobe can have dendrite connections with up to two different source lobes. These are described as 'type0' dendrites and 'type1' dendrites. This page describes the settings for 'type0' dendrites.

Source Lobe : This defines the brain lobe these dendrites are connected to. Data flows from neurones in the source lobe defined here to neurones in this brain lobe.

Dendrite Properties

Minimum # : Defines the minimum number of dendrites that will be connected between each cell in the current lobe and a source lobe cell.

Maximum # : Defines the maximum number of dendrites that will be connected between each cell in the current lobe and a cell in the source.

- At birth a norn will have a random brain wiring pattern set up. Each cell will have a number of dendrites ranging between the minimum and maximum number defined here. As dendrites migrate the minimum and maximum numbers are important as they define whether there is room for a new dendrite connection to be made or removed. The decision on how many dendrites to use when creating a lobe depends upon the type of information to obtain from the source lobe and what to do with it.

Spread : This appears to define the pattern used to define the 'fanning' of the dendrite connection between the current lobe and the source lobe. The options are 'Flat', 'Normal', 'Saw' and 'waS'. If the spread is 'flat' then there will be two dendrites connecting neurone x from the source lobe to neurone x in the destination lobe, then (x+1) to (x+1), etc. With the other settings, a connection from neurone x in the current lobe to neurones x and (x+1) in the source lobe, from neurone (x+1) to neurones (x+1) and (x+2), etc. The exact layout of the dendrites will also depend on the 'fanout' value described later.

Initial Attachment

Fanout : When the dendrite wiring for a brain lobe is initially created on birth of a creature, this value defines how each dendrite connection from the current lobe 'fans out' to neurones in the source lobe. The value varies from 0 through to 8.

Min LTW : A number ranging from 0 through to 255 described as the minimum Long Term Weight. A dendrite has a value known as the long term weight (LTW) and it has a minimum and maximum possible value. It's actual value floats somewhere in between the minimum and maximum.

Max LTW : The maximum value of the Long Term Weight as a number from 0 through to 255. It must be equal to or greater than the min LTW.

Min Strength : Each dendrite has a strength value represented as a number from 0 through to 255. The strength represents how 'strong' the link between the source neurone and the current neurone. When the strength is '0' then the dendrite may migrate as the link is not strong. The minimum number defined here is used for the initial wiring of the brain lobe and for initial values after dendrite migration.

Max Strength : The maximum strength value that can be assigned to a dendrite on initial brain wiring.

Migration Rules

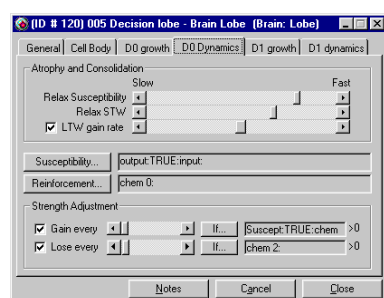
The radio button settings here define how dendrites will migrate between cells. An example of dendritic migration is the connections between the perception lobe and the concept lobe. The perception lobe is the source lobe and the concept lobe is the destination lobe. There may be several neurones from the source lobe connected to a single cell in the destination lobe. As a norm is punished then the connections on the source lobe neurones will migrate to other source lobe neurones to form more appropriate 'concepts' for the norm to learn and make decisions from.

Dendrites do not migrate : With this setting dendrites will not migrate at all and stay with the initial connections made at birth. This is most often used when the maximum and minimum number of dendrites is one with a flat spread. Nothing needs to migrate.

Migrate if ANY dendrite is loose and this cell is firing : If a neurone in the current lobe has a number of dendrites attached to a source lobe, this setting will cause the dendrites to migrate if any of these dendrites has a strength of zero and the neurone is currently firing. If a dendrite in the neurone is loose (ie. has strength of zero) but the neurone is not firing then no migration will occur. Only one of the dendrites has to be loose for the migration to occur.

Migrate when ALL dendrites are loose : In this case every dendrite from a given neurone in the current lobe must have a strength of zero for the dendrites to migrate. If only one is zero then no migration will occur.

2.5 D0 Dynamics Page



This is probably the most complicated portion of norm genetics. The definition of the dynamics of the dendrites. The 'D0' page describes the dynamics for the 'type0' dendrites. The dynamics defines how the strength value of a dendrite is adjusted, how the STW and LTV are adjusted, etc.

A number of values are associated with a dendrite. A brief description follows to make it easier to understand what the settings on this page mean. Most of these definitions were obtained from the paper by Steve Grand *et al* obtained from the Cyberlife web site.

FIGURE 4 – D0 dynamics Interface for Brain Genetics Design

STW : Short Term Weight - used to modulate input signals. STW is constantly relaxing towards LTV. It is the STW combined with the input value from the source lobe that is used to calculate the 'dendrite value'. The formula used appears to be : $dendritevalue = sourcecell * (stw/255)$

where 'source cell' is the value of the cell that this dendrite is attached to from the source lobe and 'stw' is the dendrites current short term weight value. The formula for calculation of STW appears to be : $stw = ltv + (susceptibility/255) * reinforcement$.

LTV : Long Term Weight - acts as a rest state for STW and provides statistical response to reinforcement. LTV is constantly rising towards STW. LTV and STW move towards each other at different rates (and these rates are defined genetically).

Susceptibility : Current susceptibility to reinforcement. As can be seen in the STW calculation above, the susceptibility modulates the reinforcement value. So the higher the 'susceptibility' value, the more 'reinforcement' affects the short term weight value.

Strength : Controls dendrite migration. Represents how 'strong' the dendrite link is. If strength hits zero then the link is not strong and may migrate.

Atrophy and Consolidation

Relax Susceptibility : The is the half-life of the 'susceptibility' of the dendrite. It is a value from 0 through to 255 and defines a time interval in a similar manner to that of 'Leakage Rate' in 'Cell Body Dynamics'. The value shown in the image above is 20 seconds. So when the susceptibility of a dendrite is set via the Susceptibility SVRule this setting describes how quickly it 'relaxes' back to the rest state. The relationship between this value and the susceptibility SVRule is similar to that of 'Leakage Rate' and 'State SVRule' in the 'Cell Body' page.

Relax STW : This is the half-life or relaxation rate of the Short Term Weight value for the dendrite. The value in the image above represents 5 minutes. Once again it is a number from 0 through to 255 representing a time span. The STW relaxes towards the Long Term Weight (LTW) value of the dendrite. This relax rate sets the speed of that relaxation.

LTW Gain Rate : If this option is checked then LTW constantly rises towards STW at the rate defined in the slider box. This number is a rate in the same manner as all the other leakage rates (even through the genetics kit does not display it as such). For example, if set to 10 seconds (a value of 40) then the LTW will increment by one every 10 seconds until it reaches its rest state (which is the value of STW).

Susceptibility : An SVRule that defines how the susceptibility to reinforcement for that dendrite. The higher the value of the result of this svrule, the more effect 'reinforcement' has on the value of STW. The SVRule defined in the image above basically says if there is an output value for the cell (ie. it is firing) then the susceptibility of the dendrite is equal to that of the input value.

Reinforcement : This is the SVRule used to compute changes in STW. Typically it is set to the value of some chemical. This chemical is adjusted using the various receptors/emitters available in a norn. The STW is increased based upon the value computed by this SVRule modulated by the value of the susceptibility SVRule. In the example above it means that STW is adjusted when the amount of brain lobe chemical 0 changes in this norn.

The relationship between STW, LTW, susceptibility and reinforcement is shown by the following formulae that is used to calculate the values : $dendritevalue = sourcecell * (stw/255)$ and $stw = ltw + (susceptibility/255) * reinforcement$.

Strength Adjustment

Gain Every : If this is checked then the strength value of the dendrite connection will be increased by the value indicated in the slider (from 0 through to 255) if the result of the given SVRule expression is greater than zero. The SVRule given above is 'Suscept :TRUE :chem0 :TRUE :STW'. I believe that this expression means that if the susceptibility of the connection is greater than zero and chemical 0 is greater than zero then the value of the expression is STW. So the dendrite will be strengthened if it is susceptible to reinforcement, has chemical 0 and the STW is greater than zero.

Lose Every : If this is checked then the strength value of the dendrite connection will be decreased by the value indicated in the slider (from 0 through to 255) if the result of the given SVRule expression is greater than zero. The SVRule given above is 'chem 2'. So if the value of chemical 2 is greater than zero then the strength of the dendrite will be reduced.

2.6 Dendrite Overview

This is a brief outline of how dendrite values are calculated mostly obtained from the technical papers available at the cyberlife web site. I'll add to this and hopefully provide some programs that demonstrate how things work to make it easier to understand. The following is quoted from one of the technical papers by Steve Grand *et al* :

After disturbance, both the STW and the LTW relax exponentially towards other, with the LTW being the slower. The STW therefore reacts strongly to individual reinforcement episodes, while the LTW effectively computes a moving average of many STW disturbances : if a creature meets with situation X and finds that its chosen course of action was undesirable, then it should immediately be strongly disinclined to repeat the action, especially as many of the incentives to do so may still be present. However, situation X may not always be as bad as first experience suggests, and so the creatures long-term interpretation should be less sweeping

Dendritic Migration. The initial wiring is defined at birth according to a small number of genetic rules. Generally, neurones attempt to connect from one lobe to another in a direct spatial mapping, with multiple dendrites fanning out in a specified distribution to either side of the optimum source cell. After birth, however, individual dendrites may migrate and form new connections (always within the same source lobe). Periodically, a Strength value change is computed for each synapse using SVRules, often in response to chemical changes. If the Strength falls to zero, the dendrite disconnects and follows the appropriate rule about how to find a new connection. These migration rules were chosen in order to fulfill the requirements for the initial brain model.

2.7 Issues

The following is a list of issues to be examined as I explore the functionality of each field in the genetic structure of the brain lobe gene. If you know the answer to any of these issues please email me. Meanwhile I'll be working my way through them in no particular order.

1. Can brain lobes be switched on or off based upon the gene header details. That is, can different lobes be created for different genders and can lobes be turned on when a norm hits certain ages. Is there any useful benefit that can be explored if this functionality exists.
2. What is the effect of overlapping brain lobes? If the (X, Y, W, H) values of a lobe cause a portion to overlap another lobe what affect does this have on the values of the neurones?
3. Does the location of a brain lobe within the brain grid or the different height and width combinations have any affect on the functionality of a brain lobe? Does it affect dendrite fanout and migration for example.
4. What does 'Mutually Exclusive' in the 'Data Copied to Perception Lobe' do?
5. What is the effect of changing 'input gain hi-lo' on the 'Cell Body' page?
6. Work out the exact affect of 'spread' and fanout in the dendrite pages. Draw a diagram showing what it does instead of that horrible textual description!!
7. Determine if the 'shape' (W, H) of a lobe affects the fanout of the dendrites.
8. Describe the functionality of STW and LTW of dendrites.
9. Are the Min LTW, Max LTW, Strength, etc only used for randomly setting values of the dendrites at birth? Or do they define the minimum and maximums that can apply throughout the lifetime of the dendrite?
10. Work out exactly what each setting for the migration rules of the dendrites means.
11. Does migration of dendrites only occur when the cell is firing in all cases?
12. Investigate and describe what all the values for a dendrite are (susceptibility, etc).
13. Investigate how susceptibility works.
14. Document what time span each number in the half-life sliders relates to.
15. Write a program that demonstrates how STW, LTW, etc interact.
16. Expand and refine the dendrite overview.

Progress on Issues

Issue 3

Yes the height and width of a lobe affects the original dendrite connections. The following image shows a source lobe of height 1, width 3 connected to a destination lobe with height 1, width 3 : *see figure 5.*

Notice how the dendrite links are basically cell by cell. Cell 0 is connected to cell 0 and 1. Cell 1 is connected to cell 1 and 2, etc. If the lobes are done as a 3x3 square we get more of a spread in the connections : *see figure 6.*

I've only filled in a few lines here. I really need a 3D diagram to do it justice! Image the two lobes overlay each other. Now each cell can connect to the cells to either side of it. So cell 5 in the output lobe is next to cell 5 and cell 7 in the input lobe so they can be connected. Cell 8 is next to Cell 4 so it can be connected, etc.

By changing the width and height of the input and output lobe you can adjust the initial spread of the dendrites - perhaps giving a wider spread than would otherwise have been done if it was one straight lobe. I believe it would also give more cell choices for dendrite migration as well.

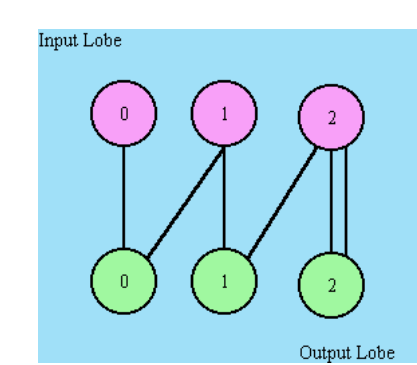


FIGURE 5 – Impact of original height and with on original connections.

Issue 4

Solved thanks to some pointers from a couple of anonymous tipsters. Information has been updated on the Brain Lobe page and the Perception Lobe Tutorial.

2.8 Perception Lobe Tutorial

The perception lobe works slightly differently to the other brain lobes so I’ve created this tutorial to demonstrate what makes it special. — The purpose of the perception lobe appears to be to summarise everything that a norn can ‘perceive’ into one brain lobe so the norn can eventually store concepts and memories based upon these perceptions (done in the concept lobe). This tutorial attempts to answer the questions about how the perceptible lobes are summarised into the perception lobe and how to work out what each cell of the perception lobe means. — The usual method of passing cell information from one lobe to another is through wiring up dendrites. Using dendrites you can only summarise information from two other brain lobes (the D0 and D1 dendrite connections). With the perception lobe there is more than two lobes that need to be summarised so it appears a mechanism specifically for dealing with the perception lobe was built. – In the Genetics Kit each brain lobe has an option that can be set indicating whether or not the lobes data is copied to the perception lobe.

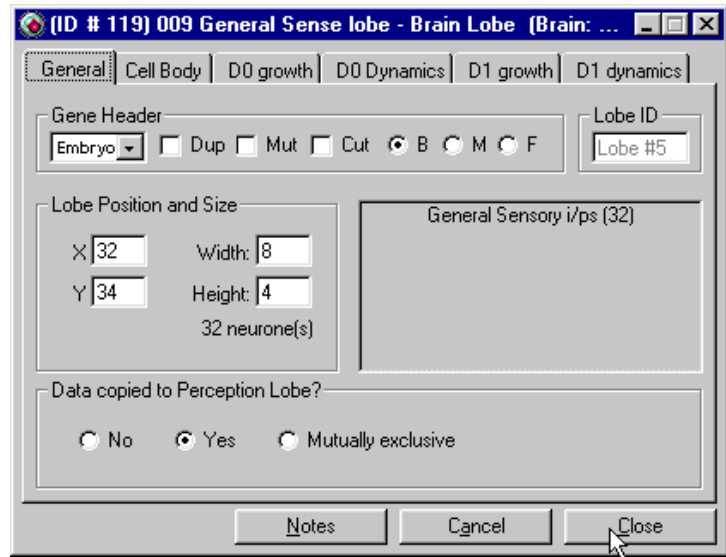


FIGURE 7 – General Senses.

The perception lobe must have a number of cells equal to or greater than the total number of cells in all lobes marked as ‘Yes’ or ‘Mutually exclusive’. The total number of cells in the above lobes equals 104. The size of the perception lobe is 112 so there is a little room to spare there.

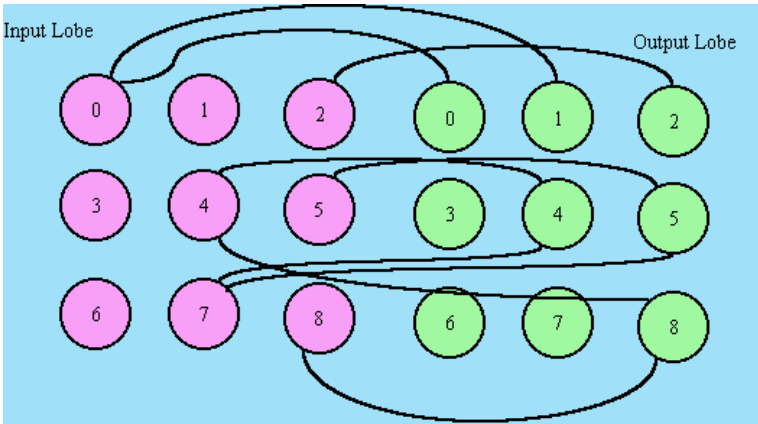


FIGURE 6 – Proximity connection of cells.

As can be seen from the above dialog box (figure 7) from the general sense lobe the options for perception are :

- No
- Yes
- Mutually exclusive

The brain lobes that are marked as ‘Yes’ or ‘Mutually exclusive’ in an original norn are :

Lobe	Size	Data copied to perception lobe ?
Drive lobe	16	Mutually exclusive
Verb lobe	16	Mutually exclusive
General sense lobe	32	Yes
Attention lobe	40	Yes

My theory from observation is that starting from the lowest numbered perceptible lobe, the output of each cell of that lobe is copied to the lowest available cell in the perception lobe. So with the perceptible lobes listed above I think the mapping to the perception lobe cells will be :

Perception cell number	Other lobe cell number
0-15	Drive lobe 0-15
16-31	Verb lobe 0-15
32-63	General sense lobe 0-31
64-103	Attention lobe 0-39

I do not know what the setting 'Mutually exclusive' means. From my tests it appears to do the same as 'Yes' but further experimentation may show otherwise. The following examples will attempt to demonstrate whether my theory about how the perception lobe works and is mapped is correct or not.

For this example we start with a norn with the normal nine lobes selected in Creatures and we will test the lobes with Perceptible marked as 'Yes'. Run the BrainCellMonitor program, connect to Creatures, and use the 'Add' button to view the following Lobe/Cell/Dendrites :

- Lobe 0 Cell 32 Dendrite 0
- Lobe 0 Cell 63 Dendrite 0
- Lobe 5 Cell 0 Dendrite 0
- Lobe 5 Cell 31 Dendrite 0

This will allow us to view the first and last cell in the general sense lobe and compare it against the perceptible cell we think they will be copied to.

We will now use CAOS commands to fire particular cells in the general sense lobe (lobe 5) and see if the corresponding cell in the perception lobe (lobe 0) fires with equivalent values.

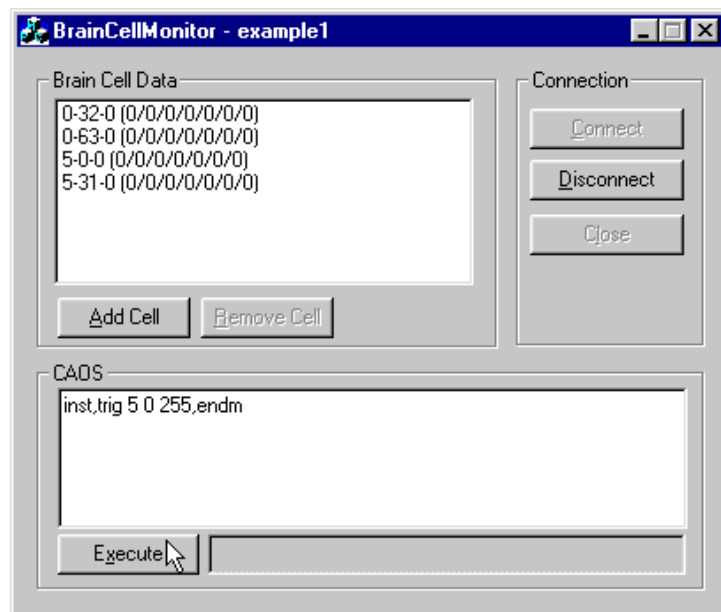


FIGURE 8 – Brain Cell Monitor — Example 1.

Try executing the CAOS command : `inst,trig 5 0 255,endm` — Notice how the perception lobe cell number 32 output number increases and starts to decrease in a similar way to the general sense lobe cell we just fired :

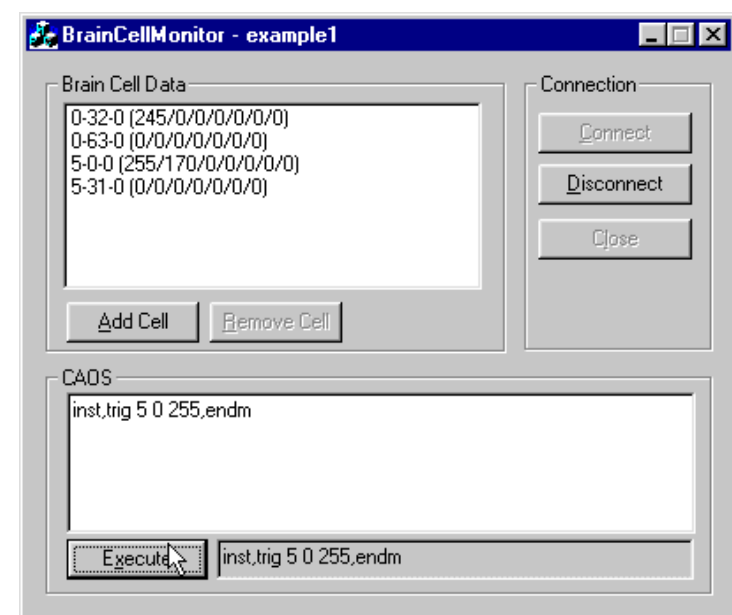


FIGURE 9 – Brain Cell Monitor — Example 2.

This seems to imply that our assumed mapping may be correct.

Try executing the CAOS command : `inst,trig 5 31 255,endm`

Notice how the perception lobe cell number 63 output number increases and starts to decrease in a similar way to the general sense lobe cell we just fired. This also seems to confirm our mapping. Try using different cell numbers with lobe 5 and you will see that the mapping we came up with is correct. Now we'll try it with the Attention lobe. Use BrainCellMonitor to view the following cells :

- Lobe 0 Cell 64 Dendrite 0
- Lobe 0 Cell 103 Dendrite 0
- Lobe 7 Cell 0 Dendrite 0
- Lobe 7 Cell 39 Dendrite 0

Try executing the CAOS command : `inst,trig 7 0 255,endm` and executing the CAOS command : `inst,trig 7 39 255,endm` — Once again you should see the equivalent perception lobes firing.

You may notice that when doing the General sense example that if you fire both general sense lobes then both perception lobes also fire. But in the Attention example, firing both will only cause one perception lobe cell to fire - why is this? It is because the Attention lobe is marked 'Winner takes all'. This means that only the cell with the highest output actually fires - all the others get automatically set to zero and this data is copied to the perception lobe.

Well what about the 'Mutually Exclusive' option? Lets try some examples with the Verb lobe which is marked "mutually exclusive". I don't use the Drive lobe just yet as the drives for a norn are constantly changing and can get in the way of our observations. Try viewing the following :

- Lobe 0 Cell 16 Dendrite 0
- Lobe 0 Cell 31 Dendrite 0
- Lobe 3 Cell 0 Dendrite 0
- Lobe 3 Cell 15 Dendrite 0

Try executing the CAOS command : `inst,trig 3 0 255,endm` and executing the CAOS command : `inst,trig 3 15 255,endm` — Once again you should see the equivalent perception lobes firing. The verb lobe is also marked 'Winner takes all' so you'll see similar effects to that shown in the Attention lobe. — You can try examples with the drive lobe as well but as mentioned before the norn drives can get in the way of observation.

The Mutually Exclusive option works in the same manner as lobes marked as 'Yes'. The difference between the two options appears when dendrites are linked between the perception lobe and the concept lobe. Usually from 1 to 3 perception cells are linked to a particular concept cell. The perception cells that are linked can change during the lifetime of a norn based on experiences it has encountered. But only one cell from a given Mutually Exclusive lobe can contribute to the forming of a particular concept. Drive is mutually exclusive so we can have a concept of 'hungry' but not 'hungry' and 'tired'. Verb lobe is also mutually exclusive so we can have a concept of being told to 'push' but not being told to 'push' and 'pull'.



3 Chemical Emitter

3.1 Overview

A chemical emitter gene controls the injection of a particular chemical into the norn based upon data retrieved from some Organ, Tissue or Locus within the norn. The Organ, Tissue or Locus is a data value controlled by the Creatures executable ranging in value from 0 through to 255. It can represent a brain cell, external stimuli or other things selected from the dialog box. Every given time period the Organ, Tissue or Locus is examined and compared against a number of settings defined in this gene. Depending upon the results of this comparison an amount of chemical can be injected into the norn.

3.2 Dialog Box

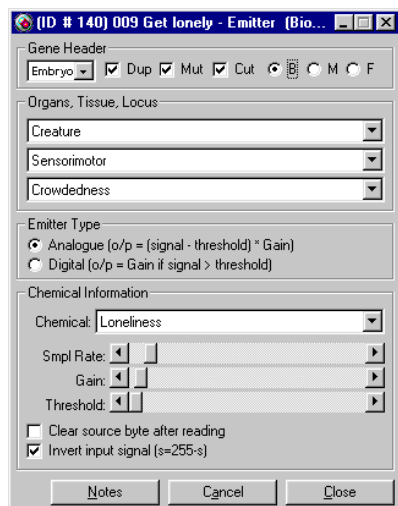


FIGURE 10 – Interface for Chemical Emitter Design

3.3 Breakdown – *Gene Header*

Emitter Type

Analogue : The amount of chemical injected into the norn will be proportional to the signal level received from the locus. The following formula will be applied to know how much chemical to inject : $(signal - threshold) * (gain/255)$.

Digital : A constant amount ('gain') of chemical will be injected into the norn when the 'signal' reaches a specified 'threshold'.

Chemical Information

Chemical : Defines the chemical that will be added to the norn.

Smpl Rate : Sample Rate (inverted) indicates how often the emitter is processed and therefore how often the chemical will be injected.

Gain : Defines how much chemical will be injected. Note that this will increase the amount of chemical into by the given amount, not replace it. See the description of 'Emitter Type' above for the calculation used

Threshold : The locus must be firing at this level for a chemical injection to occur. In the case of Digital emitters the signal from the locus must be greater than this threshold. For Analogue emitters the signal is reduced by the amount of this threshold. See 'Emitter Type' above for the calculation.

Clear Source Byte after Reading : After the emitter is processed the locus value will be reset to zero if this option is selected.

Invert Input Signal : The signal value used in all calculations in the emitter will be $255 - signal$ if this option is selected. This new signal value will then be used for all emitter calculations (including against the threshold).

Gene Header

Embryo : This field indicates the age at which this gene will switch on. Most genes turn on at the 'Embryo' stage to ensure they operate correctly as soon as the norn is born.

Dup : If this flag is checked then the gene can be duplicated as a genetic mutation. That is, multiple copies of this gene will be able to exist due to mutation.

Mut : Sets whether genetic mutations can be applied to the values contained within this gene.

Cut : If set the gene can be removed during breeding or as the result of mutation.

B/M/F : Defines the gender that this gene will switch on. This is often used to select between genes that involve male or female functionality (eg. hormones).

4 Chemical Receptor

4.2 Dialog Box

4.1 Overview

A chemical receptor gene allows an Organ, Tissue or Locus within the norn to be changed based upon the level of a chemical within that norn. The chemical associated with the receptor is constantly monitored to see if it surpasses a given threshold. When that threshold is reached a formula is calculated from the chemical amount and the result is applied to the Organ, Tissue or Locus selected.

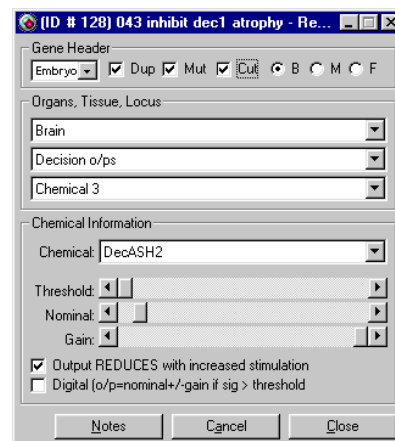


FIGURE 11 – Interface for Chemical Receptor Design

4.3 Breakdown

Gene Header : The gene header is the same for all genes.

Chemical Information

Chemical : Defines the particular chemical that will be monitored.

Threshold : Defines the level of the chemical that must exist before the given locus is activated. In the case of Digital receptors the amount of chemical must be greater than this threshold before the locus is activated. For Analogue receptors the signal (ie. amount of chemical) is reduced by the amount of this threshold before calculating the amount to stimulate the locus by. See Formula below for the calculation used.

Nominal : Defines the default or base value used to stimulate the locus. This what the locus will be set to if the amount of chemical is not greater than the threshold.

Gain : Defines what the value of the locus will be set to in the case of a digital receptor or as a scaling factor for calculating the amount in an analog receptor. See Formula below for details.

Output REDUCES with increased stimulation : If this is checked then all adjustments to the base value of the receptor (ie. the Nominal amount) will reduce that base value. If cleared all adjustments will increase the base value.

Digital : The locus will be set to a constant amount if the chemical amount is greater than a certain threshold if this option is checked. If it is unchecked then the receptor is analogue. This means that the locus will be set to a value in proportion to the amount of chemical. See below for details.

Formula

Analogue receptors formula for calculating the value for the locus :

$$Nominal + (((ChemicalAmount - Threshold) * Gain / 255) * R)$$

Where R is 1 if 'Output Reduces with increased stimulation' is not checked and -1 if it is checked. So the nominal will be reduced or increased based upon this flag.

Digital receptors formula for calculating the value for the locus :

$$Nominal + ((ChemicalAmount > Threshold ? Gain : 0) * R)$$

So if the chemical amount is greater than the threshold then the locus setting will be the Nominal amount increased or decreased by the Gain depending on the setting of 'R'. If it is not greater than the threshold then the locus setting will be equivalent to Nominal.



5 Initial Concentrations

5.1 Overview

This gene controls the amount of chemical that will exist within the norn when it is born (or when the gene is activated?). One gene of this kind exists for each chemical within the norn that requires an initial starting value. I don't know the effects of changing the 'switch on' age. If the age is set to something other than embryo does this mean that an amount of chemical will be injected when the creature reaches that age?

5.2 Dialog Box

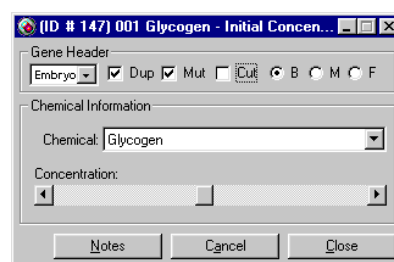


FIGURE 12 – Interface for Initial Concentration Design

5.3 Breakdown

Gene Header : The gene header is the same for all genes.

Chemical Information

Chemical : This will be the chemical that will have the initial concentration defined for this gene.

Concentration : The amount of chemical to be initially injected into the norn. It is a value from 0 through to 255.

6 Creature Instinct

6.1 Overview

An instinct gene provides a way of training the neural net of the norn brain so it will react in certain ways in certain situations. Instinct genes are often used to provide default behavior for things that are very important to the norn. Examples are responding to the verbs typed by the user. These genes do not provide behavior that will always occur. The experiences of the norn during its lifetime can override behavior defined by instincts.

Instincts are processed while a norn is being hatched and while a norn sleeps. It is very important for a norn to get regular sleep so the instincts are constantly reinforced. During this instinct processing time the lobes and neurones define in the gene are set inside the norn brain and the chemical (usually punishment or reward) is processed as if the norn had performed this action.

For example the instinct gene responding to the verb 'come' has the same effect when processed as if the norn had chosen the decision to 'come' when the user typed 'come' and then got a pat from the hand to reward it. This would encourage the norn to do this when awake.

6.2 Dialog Box

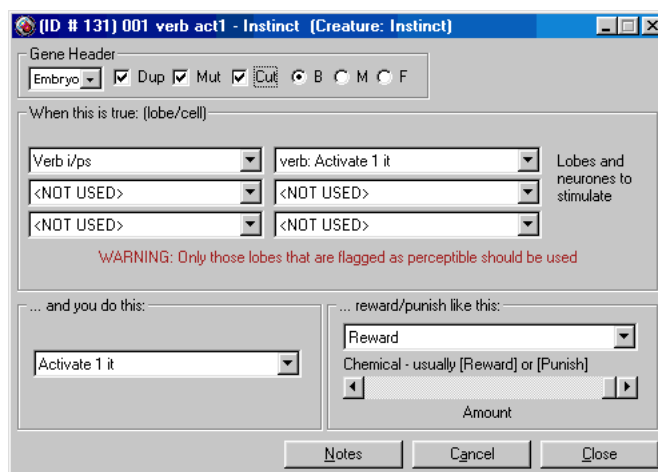


FIGURE 13 – Interface for Intincts Design

6.3 Breakdown

Gene Header : The gene header is the same for all genes.

When this is true : (lobe/cell)

Allows selection of the lobes and cells that will be fired within those lobes. This indicates what the norn will need to perceive before it will decide to take the action defined in the 'and you do this' area. A combination of up to three different lobe/cells may be selected. The Genetics Kit gives a warning that only those lobes marked as perceptible should be used. This makes sense as only the perceptible lobes contribute to forming concepts. Any other lobe will cause this instinct to have no real effect.

The lobes and their perceptible settings are :

Lobes and perceptible settings	
Lobe	Perceptible setting
Drive i/ps	Mutually exclusive
Stim source i/ps	No
Verb i/ps	Mutually exclusive
Noun i/ps	No
Gen/sense i/ps	Yes
Decision	No
Attention	Yes

TABLE 1: Lobes and perceptible settings

All lobes with a perceptible setting of 'Yes' or 'Mutually Exclusive' qualify as perceptible lobes so they should be able to be used in instincts.

and you do this...

This is the decision that the instinct will reinforce when the above lobe/cell combinations are active. Think of it as 'if the above lobe combinations occur then the norn will be more or less likely to make this decision depending on the reward/punish settings defined below. *Valid settings are any of the 16 possible decisions or verbs that the norn can perform* :

0. Default (**Stay**)
1. Activate 1 it (**Push**)
2. Activate 2 it (**Pull**)
3. Deactivate it (**Stop**)
4. Approach it (**Come**)
5. Retreat from it (**Run**)
6. Get it (**Get**)
7. Drop all (**Drop**)

8. Say what you need (**Think**)
9. Rest (**Sleep**)
10. Travel west (**Left**)
11. Travel east (**Right**)
12. Decision12
13. Decision13
14. Decision14
15. Decision15

reward/punish like this

This is where you select the chemical and amount of that chemical that will be injected into the norn when the above settings are made and the instinct processed. Although any of the 256 chemicals can be selected the only two that really make sense are 'Reward' or 'Punish'.

If 'Reward' is selected then the norn will be rewarded and encouraged to perform the action. If 'Punish' is selected then the norn will be discouraged and be less likely to perform the action defined.

6.4 Notes

Remember that instincts are only processed when a norn is born and while it sleeps. Too many instinct genes could cause a lot more sleep or longer sleep to be required for all the instincts to be processed. It may pay to have the more important instincts to be the lower numbered genes (as it appears to cycle through in gene number order).

Some instinct genes appear to use the Stim source lobe as an input even though it is not a perceptible lobe. I'm in the process of testing whether these instincts actually work as I suspect they may not. It depends whether the instinct processing mechanism still considers it as perceptible as the Stim source lobe feeds into the Attention lobe which is perceptible. Some research is needed here.

Although only 'Reward' and 'Punish' currently make sense for chemicals, others could be useful in genetically modified norns with different brain/chemical structures. There is scope for some interesting experimentation there. Only three lobe/cell combinations can be defined. This appears to match the number of dendrite links between the perception lobe and the concept lobe in a standard hatchery norn. These norns allow up to three perceptions to form a concept.

7 Creature Stimulus

7.1 Overview

A stimulus is an event that happens to the norn that it can react to in some manner. The stimulus gene allows the reaction to these stimuli to be defined. This can range from chemicals being injected to cell values in brain lobes being adjusted. There are a number of stimulus responses that are hard coded within the Creatures program. The stimulus genes that are created override these hard coded responses giving a lot of flexibility in determining how the creature will react to the world.

7.2 Dialog Box

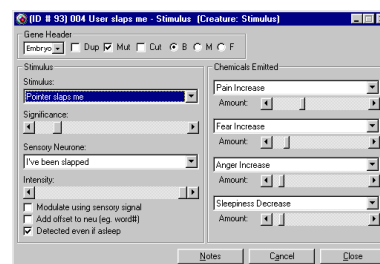


FIGURE 14 – Interface for Stimulus Design

7.3 Breakdown

Gene Header : The gene header is the same for all genes.

Stimulus

Stimulus : The stimulus that this gene is attached to. When this stimulus occurs then the effects defined by this gene will be applied. The following stimuli are available :

<ul style="list-style-type: none"> - Disappointment - Pointer pats me - Creature pats me - Pointer slaps me - Creature slaps me - It is approaching - It is retreating - I bump into wall - Object comes into view 	<ul style="list-style-type: none"> - Unrecognised word - Heard user speak - Heard creature speak - I am quiescent (periodic) - I've activated1 - I've activated2 - I've deactivated - I am approaching (periodic) - I have retreated 	<ul style="list-style-type: none"> - I have got - I have dropped - I've stated need - I am resting (periodic) - I am sleeping (periodic) - I am traveling (periodic) - spare action 1-4 - Involuntary action 0-7
---	---	--

Significance : The amount will be increased the neuron relating to the object that caused the stimulus in the Stimulus/Source lobe. This has the effect of making the object appear more interesting and so it is more likely to be looked at. The neuron that gets nudged by this amount appears to differ for each Stimuli (a neuron for each object type in view).

Sensory Neurone : Identifies a cell in the General Sense lobe which will get adjusted by the value given in the 'Intensity' field described below. This allows setting up what the norn can actually sense from the stimuli going on around it. The following cells are available and map directly to cells in the general sense lobe :

<ul style="list-style-type: none"> - <none> - I've been patted - I've been slapped - I've bumped a wall - I am near a wall - I am in a vehicle - User has spoken - Creature has spoken 	<ul style="list-style-type: none"> - Own kind has spoken - Audible event - Visible event - It is approaching - It is retreating - It is near to me - It is active 	<ul style="list-style-type: none"> - It is an object - It is a creature - It is my sibling - It is my parent - It is my child - It is opposite sex - spare 2-13
--	--	--

Intensity : This is the amount by which the particular sensory neurone will be increased by. So if the value here is '50' and the current value of the particular neurone is '25' then when this stimuli occurs the sensory neurone will fire at a value of about '75'.

Modulate using sensory signal : From what I can gather this means that the 'significance' value applied to the stimulus source lobe is first modulated by the value of the neuron associated with the stimulus. I'm not exactly sure what particular value it is modulated by and perhaps it is best explained by describing some of the results I've gotten through experimentation. See the notes section for details. If anyone can provide a clearer explanation I'd appreciate an email.

Detected even if asleep : If the norn is asleep and this option is unchecked then the stimulus will be ignored and none of the effects will be applied. If it is checked then the gene will be processed even if the norn is sleeping.

Chemicals Emitted

Up to four chemicals can be selected along with a particular amount of that chemical to be injected. When the stimulus occurs the defined amount of these chemicals will be injected into the norn.



8 Brain Lobes – *Creatures 1*

8.1 Overview

The norn brain is divided into a number of lobes. In the standard *Creatures 1* genome there are nine and in the *Creatures 2* genome there are ten.

The brain lobes numbered from 0 through to 8 are hard coded by the *Creatures* executable to perform a particular function. For example, firing a cell in lobe 6 (decision) causes the creature to perform a particular action. The genetic definition of that lobe then defines what happens as a result of this firing. In the case of the decision lobe it is to manage the learning of whether the decision was a good one or a bad one.

The higher numbered brain lobes are not controlled by the *Creatures* executable at all. They are controlled completely through genetics. So for these lobes to do anything a system of receptors, emitters and dendrites must be set up to fire cells or act upon cells firing.

The purpose and function of each lobe is outlined below with descriptions.

Lobe 0 – Perception Copies data from other lobes to form a composite lobe containing all cells that can be used to form concepts.

Lobe 1 – Drive Holds the current values of the creatures drives.

Lobe 2 – Stimulus source Stimulating objects near to the creature cause cells in this lobe to fire so the creature can react to them.

Lobe 3 – Verb Cells in this lobe fire when the user types a verb in the speech box.

Lobe 4 – Noun Cells in this lobe fire when the user types a noun in the speech box or types 'look' with the hand hovering over an object.

Lobe 5 – General Sensory Indicates events that the norn can sense. Usually caused by the effects of processing stimulus genes.

Lobe 6 – Decision An output lobe. By firing a cell in this lobe you cause the creature to perform a particular action.

Lobe 7 – Attention An output lobe. By firing a cell in this lobe you cause the creature to look at a particular object.

Lobe 8 – Concept A storehouse of memories and concepts.

Lobe 9 – Regulator This lobe provides feedback loops for biochemical regulations.

While the lobes have individual functionality it is how they work together that results in the norn brain working. There lobes combine to form a couple of main learning systems that are described briefly in the individual lobe descriptions but will be covered in more detail in future updates :

- Attention seeking system
- Concept learning system
- Decision learning system

Number	Name	X	Y	Width	Height	Neurones
0	Perception	4	13	7	16	112
1	Drive	34	30	8	2	16
2	Source	15	24	8	5	40
3	Verb	37	24	8	2	16
4	Noun	21	3	20	2	40
5	Gen. Sense	32	34	8	4	32
6	Decision	53	15	1	16	16
7	Attention	44	30	5	8	40
8	Concept	12	6	40	16	640

TABLE 2 – Norn Lobes for *Creatures 1*

*Description of lobes positions and size, brain is 64 * 48. The Regulator lobe, SandraBellum, is not present.*

Number	Name	X	Y	Width	Height	Neurones
0	Perception	1	13	7	16	112
1	Drive	34	36	6	3	18
2	Source	15	32	8	5	40
3	Verb	37	32	8	2	16
4	Noun	21	1	20	2	40
5	Gen. Sense	32	40	8	4	32
6	Decision	62	15	1	16	16
7	Attention	44	36	5	8	40
8	Concept	11	5	40	16	640
9	Regulator	4	40	16	1	16

TABLE 3 – Norn Lobes for *Creatures 2* Description of lobes positions and size, brain is 64 * 48.

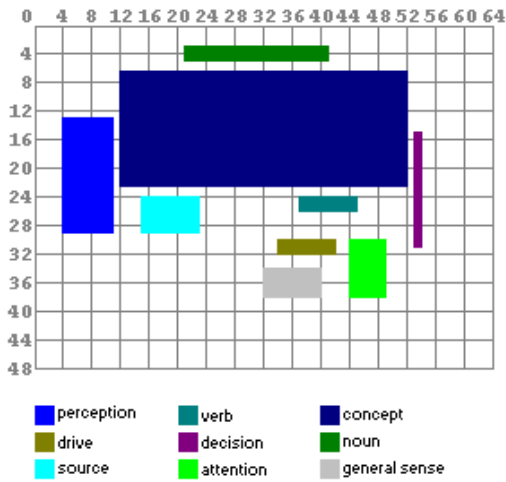


FIGURE 15 – Creatures 1 Brain Map.

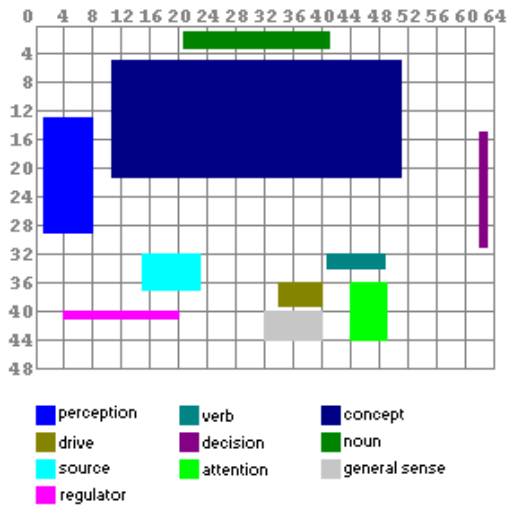


FIGURE 16 – Creatures 2 Brain Map.

8.2 Lobe 0 – Perception Lobe

The purpose of the perception lobe appears to be to summarise everything that a norn can 'perceive' into one brain lobe so the norn can eventually store concepts and memories based upon these perceptions (done in the concept lobe). This tutorial attempts to answer the questions about how the perceptible lobes are summarised into the perception lobe and how to work out what each cell of the perception lobe means.

The usual method of passing cell information from one lobe to another is through wiring up dendrites. Using dendrites can only summarise information from two other brain lobes (the D0 and D1 dendrite connections). With the perception lobe there is more than two lobes that need to be summarised so it appears a mechanism specifically for dealing with the perception lobe was built.

Brain lobes are marked as 'Yes', 'No' or 'Mutually Exclusive' to be copied in the Perception lobe. The brain lobes that are marked as 'Yes' or 'Mutually exclusive' in an original norn are :

Copied Lobes in Perception		
Lobe	Size	Data copied to perception lobe ?
Drive lobe	16	Mutually exclusive
Verb lobe	16	Mutually exclusive
General sense lobe	32	Yes
Attention lobe	40	Yes

TABLE 4: Copied Lobes in Perception

The perception lobe must have a number of cells equal to or greater than the total number of cells in all lobes marked as 'Yes' or 'Mutually exclusive'. The total number of cells in the above lobes equals 104. The size of the perception lobe is 112 so there is a little room to spare there.

8.3 Lobe 1 – Drive Lobe

This lobe holds the current state of the creatures drives (such as hunger, pain, etc). It directly maps to the particular drive chemicals that can be viewed using the science kit. Changing the values of the cells directly in this lobe using CAOS is unlikely to have much of an effect on the norn as the cells are directly set by receptors in the genome. This means that whatever values you manipulate with CAOS commands will be immediately overwritten with the current value of the drive chemical. To change drive values it would seem to be best to change the values of the chemicals either directly or through the drive increaser/reducer chemicals. *The size of the lobe was increased in Creatures 2 to cater for the addition of more drives. The cell values are outlined in the following table :*

Drive Lobe Data Entries <i>Drives</i>		
Cell	Creatures 1	Creatures 2
0	Pain	Pain
1	Need for Pleasure	Need for Pleasure
2	Hunger	Hunger
3	Coldness	Coldness
4	Hotness	Hotness
5	Tiredness	Tiredness
6	Sleepiness	Sleepiness
7	Loneliness	Loneliness
8	Overcrowdedness	Overcrowdedness
9	Fear	Fear
10	Boredom	Boredom
11	Anger	Anger
12	Sexdrive	Sexdrive
13	<i>Not Allocated</i>	Injury
14	<i>Not Allocated</i>	Suffocation
15	<i>Not Allocated</i>	Thirst
16	<i>No such cell</i>	Stress
17	<i>No such cell</i>	<i>Not Allocated</i>

TABLE 5: Drive Lobe Data Entries *Drives*

The drive lobe feeds into the Perception lobe and is marked as mutually exclusive. This means that, at most, one cell from the drive lobe will exist in any given concept that the creature forms. So the creature cannot form the concept of 'hot and hungry'. Only 'hot' or 'hungry' or a combination of these drives with some other perception lobe cell.

8.4 Lobe 2 – Stimulus Source Lobe

Cells in this lobe are activated directly by the Creatures executable. The lobe has 40 cells - one for each object classification in Creatures. – Each cell relates to a particular object classification. The cell fires for a given object if that object is within the creatures line of sight and is stimulating in some manner. Stimulus genes can be used to modify this lobe indirectly to indicate whether the object is more or less stimulating. The higher the cell value in a particular cell, the more stimulating that object currently is.

It is a 'winner takes all' lobe, meaning that the highest firing cell will be the only firing cell. This means that the most stimulating object in the area of the norn will be the only object that will appear to be stimulating to the norn. The State value for all the cells is calculated and the cell with the highest state will be the only one with an output value - indicating that it is firing.

The following table outlines the cell values for this lobe :

Source Lobe Data Entries		
Cell	Creatures 1	Creatures 2
0	Current norn	Current norn
1	Hand	Hand
2	Call button	Call Button
3	Water	Nature
4	Plant	Plant
5	Egg	Egg
6	Food	Food
7	Drink	Drink
8	Vendor	Dispenser
9	Music	Implement
10	Animal	Cliff Edge
11	Fire	Detritus
12	Shower	Medicine
13	Toy	Toy
14	Bigtoy	Weather
15	Weed	Badplant
16	Incubator	Nest
17	Blackboard word 33	Badbug
18	Blackboard word 34	Bug
19	Blackboard word 35	Badcritter
20	Blackboard word 36	Critter
21	Blackboard word 37	Seed
22	Blackboard word 38	Leaf
23	Blackboard word 39	Root
24	Blackboard word 40	Flower
25	Blackboard word 41	Fruit
26	Mover	Mover
27	Lift	Lift
28	Computer	Computer
29	Fun	Mediabox
30	Bang	Message
31	Blackboard word 47	Leftright
32	Blackboard word 48	Incubator
33	Blackboard word 49	Teleporter
34	Blackboard word 50	Blackboard word 50
35	Blackboard word 51	Machine
36	Norn	Norn
37	Grendel	Grendel
38	Ettin	Ettin
39	Shee	Shee

TABLE 6: Source Lobe Data Entries

The cell numbers appear to relate to the genus numbers for Objects (COB). So an Object with a genus value of 21 sitting near a norn in Creatures 2 will stimulate cell 21 in the norn, making it think a Seed is nearby.



8.5 Lobe 3 – Verb Lobe

The Verb lobe is another lobe that is controlled directly by the Creatures executable. Whenever a verb is entered by the user in the speech box of Creatures then the cell associated with this verb will be fired in this lobe. It's also possible that verbs mentioned by other creatures nearby could fire cells in this lobe but I haven't checked into this yet. – The following table outlines the cell values for this lobe :

Verb Lobe Data Entries		
Cell	Creatures 1	Creatures 2
0	Quiescent	Quiescent
1	Push	Push
2	Pull	Pull
3	Stop	Stop
4	Come	Come
5	Run	Run
6	Get	Get
7	Drop	Drop
8	Think	What
9	Sleep	Sleep
10	Left	Left
11	Right	Right
12	<i>Not Allocated</i>	Eat
13	<i>Not Allocated</i>	Hit
14	<i>Not Allocated</i>	<i>Not Allocated</i>
15	<i>Not Allocated</i>	<i>Not Allocated</i>

TABLE 7: Verb Lobe Data Entries

The word that was learnt for the particular verb slot is what must be typed in for that cell in this lobe to fire. So if the Norn knows 'Push' as 'foo' then typing 'foo' will fire the particular cell. If a complete command is entered like 'alice eat food' then cell 12 in this lobe (for eat) will fire along with the 'food' cell in the noun lobe.

8.6 Lobe 4 – Noun Lobe

Cells in the noun lobe are fired when a noun is entered by the user using the speech box of the Creatures executable. It works the same as the verb lobe described above but it is for the objects rather than the action to be performed on the object. It is another lobe controlled directly by the Creatures executable.

The noun lobe will also fire if the user moves the hand over an object and types the command 'look'. This causes the cell for that particular object to fire in the noun lobe. The result of this is the norn usually ends up looking at the requested object. See the description of the Attention lobe for details on how this works.

8.7 Lobe 5 – General Sensory Lobe

The cells in this lobe define what the norn can currently sense in the environment. Where the stimulus source lobe is the objects within the environment this lobe is various environmental factors relating to those objects. For example, cells for detecting that the creatures has just been patted, slapped, fallen, whether nearby creatures are the same sex, same species, the parents of the current creature, etc.

The cells in this lobe can be manipulated using the stimulus genes. When a particular stimulus occurs (either by the Creatures executable or using CAOS in a COB) the stimulus gene is activated. That gene can then cause a general sensory lobe cell to fire at a particular intensity so the norn can react or learn from that stimulus.

The lobe is copied to the Perception lobe allowing the cells to be used for forming concepts. This means that a norn can learn whether 'approaching an edge' is good or bad for example.

The following table outlines the cell values for the General Sensory Lobe :

General Senses Lobe Data Entries		
Cell	Creatures 1	Creatures 2
0	I've been patted	I've been patted
1	I've been slapped	I've been slapped
2	I've bumped into a wall	I've bumped into a wall
3	I am near a wall	I am near a wall
4	I am in a vehicle	I am in a vehicle
5	User has spoken	User has spoken
6	Creature has spoken	Creature has spoken
7	Own kind has spoken	Own kind has spoken
8	Audible event	Audible event
9	Visible event	Visible event
10	It is approaching	It is approaching
11	It is retreating	It is retreating
12	It is near me	It is near me
13	It is active	It is active
14	It is an object	It is an object
15	It is a creature	It is a creature
16	It is my sibling	It is my sibling
17	It is my parent	It is my parent
18	It is my child	It is my child
19	It is opposite sex	It is opposite sex
20	<i>Not Allocated</i>	It has pushed me
21	<i>Not Allocated</i>	It has hit me
22	<i>Not Allocated</i>	<i>Not Allocated</i>
23	<i>Not Allocated</i>	<i>Not Allocated</i>
24	<i>Not Allocated</i>	<i>Not Allocated</i>
25	<i>Not Allocated</i>	<i>Not Allocated</i>
26	<i>Not Allocated</i>	<i>Not Allocated</i>
27	<i>Not Allocated</i>	<i>Not Allocated</i>
28	<i>Not Allocated</i>	Approaching an edge
29	<i>Not Allocated</i>	Retreating from an edge
30	<i>Not Allocated</i>	Falling through the air
31	<i>Not Allocated</i>	Hitting the ground post fall

TABLE 8: General Senses Lobe Data Entries

8.8 Lobe 6 – Decision Lobe

Each cell in the decision lobe relates to a particular action that the norn can perform. These actions are the same as the verbs in the verb lobe. See the table in that lobe for descriptions of the individual cells.

The Verb lobe is an input lobe - it receives input typed in from the user. The decision lobe is an output lobe. By firing a cell in the decision lobe the Creatures executable will be directed to make the norn perform the particular decision related to the cell being fired. By manually firing cells in this lobe you can force a norn to perform certain decisions. This is how the 'shove' mode of the hand works. When you have the hand in 'shove' mode, pressing a norn results in either the 'left' (cell 10) or 'right' (cell 11) cell to be activated. This will usually cause the norn to move left or right.

The decision lobe is connected to the concept lobe using type 0 and type 1 dendrites. 128 dendrites of each type connect to the concept lobe. Each dendrite link is therefore linked to a particular concept. A concept is a combination of perceptable things (eg. hungry and looking at food, tired, sleepy and just been patted by the hand, etc).



The type 0 dendrites link from decision lobe cells to the 128 concept lobe cells indicate that that particular decision is good if those particular concepts are active. This means that if the concepts become active then the norn will be more likely to choose that decision over other decisions.

The type 1 dendrites link from decision lobe cells to the 128 concept lobe cells indicate that that particular decision is bad if those particular concepts are active. This means that if the concepts become active then the norn will be less likely to choose that decision over other decisions.

8.9 Lobe 7 – Attention Lobe

Each cell in the attention lobe relates to a particular object classification in the same manner as the Stimulus Source lobe and the Noun lobe. See the table in those lobe descriptions for the meanings of the individual cells.

The Stimulus Source and Noun lobes are input lobes - they receive input from the environment. The attention lobe is an output lobe. When a cell fires in the attention lobe then the Creatures executable is told to make the norn look at a particular object. If you manually fire a cell in this lobe you will see the 'Creatures View' in the game change to whatever object corresponds to the cell you fired.

Like the stimulus source lobe, it is a 'winner takes all' lobe. This means that only one cell will actually fire - the cell with the highest state value. This makes sense as the norn can only look at one object at a time. The attention lobe copies its information to the perception lobe enabling the current object being looked at to be used in forming concepts. — A mechanism that I call the attention seeking system exists that makes the norn look at the most stimulating object nearby or an object that the user recently asked the norn to look at (by typing 'look' or entering the name of the object). This system works by using dendrite connections from the attention lobe to the stimulus source and noun lobes.

There is one type 0 dendrite linking each cell in the attention lobe to the equivalent cell in the stimulus source lobe. This means that if a cell fires in the stimulus source lobe the value of the firing can be used in state variable calculations in the attention lobe.

There is one type 1 dendrite linking each cell in the attention lobe to the equivalent cell in the noun lobe. Effectively there is a link passing information from the stimulus source and the noun lobe through to the attention lobe.

8.10 Lobe 8 – Concept Lobe

By far the largest region of a norn's brain is used to store memories of the events that occur over its lifetime. This region is called Concept Space, and seems to be involved in the collation and organization of a norn's perceptions and experiences. We know that the neurons in this region are very mobile, and are constantly reconnecting themselves to the main sensory lobe in response to new experiences. Artificial stimulation of these neurons can trigger outward behavior, but it is not certain what the relationship is between a given neuron and the behavior it produces. The Concept lobe is the biggest lobe in a Norn's brain. In Creatures 1 it has a size of 640 neurons.

This lobe receives input signals from the different sense lobes (via the perception lobe in C1 and C2). One neuron of this lobe may have between one and three input dendrites. All inputs of a single neuron are calculated together with a logical AND operation. This is why this lobe is also called "Combination lobe" in Creatures 3 – it simply combines up to three different inputs to one output. Possible meanings may be : "I am hungry AND I see food" or "There is a creature AND it is a Grendel AND it is approaching".

The dendrite's strength is increased by the "reward" chemical and decreased by the "punish" chemical. Dendrites with a strength near zero may migrate to connect to different active input signals, making new random connections possible. The output of the neurons in this lobe is passed to the decision lobe. These dendrites may get positive or negative strength values, so they represent what the creature should or should not do in a certain situation.

8.11 Lobe 9 – Regulator Lobe

The regulator lobe (or Sandrabellum named after Sandra Linkletter / slink who created it) is not part of the norm attention, concept, and decision mechanism. It supplies functionality similar to that of the floor receptor/emitters in the receptor and emitter genes but provides a whole lot more flexibility. The following description is almost verbatim from Lis Morris, co-creator of the Canny norms :

It works like the hunger/glycogen neurones used in the life kit, except the whole lobe is dedicated to these kind of positive and negative feedback loops. Therefore, you always see neurones firing in the sandrabellum, since it simply records the amount of some chemicals in the blood stream (norm stream? they don't really have blood...).

These are then coupled to an emitter that either emits something useful given the level of chemical, or controls future emission of the precursor of that chemical. Here's a list of all the receptor emitter couplings :

Regulator Lobe Loopbacks		
Receptor	Neurone	Emitter
lactate	0	suffocation
This detects drowning, and causes a choke reflex in the norm. Since my alterations to the gene, it should in fact detect myoglobin, IMO.		
Water	1	Thirst
Inverted receptor. nuff said :-)		
Glucose	2	Insulin, hunger
Oops, [name removed - you'll have to guess], two emitters on one locus! I think this is why sometimes norms suddenly collapse when their levels of glycogen, muscle tissue and adipose tissue are low...the emitter suddenly switches to producing hunger, not insulin, essentially knacker the glycogen/glucose equilibrium. I'm going to make a new locus for one of these reactions in my next released genome.		
Adrenaline	3	Stress
Again, pretty self explanatory		
Cholesterol	4	Steroidine
Steroidine is the precursor for cholesterol, and this is a negative feedback loop.		
Testosterone	5	Inhibin
Another negative feedback loop.		
Glycogen	6	Glycogen Synthetase
Negative feedback loop, stops very high levels of glycogen forming		
Bile acid	7	Bilin
Negative feedback, bilin is the precursor of bile acid.		
Adipose tissue	8	Hunger
Thin norms get hungry.		
Starch	9	Fullness
Fat	10	Fullness
Protein	11	Fullness
These cause that slow reduction in hunger often seen after the initial decrease after eating a fatty food, such as zander fish. I was worried it was production of hunger decrease doing this, and confusing the norms, but obviously not...		
Adipose Tissue	12	Adrenaline
You're too fat, go and exercise! <G>		
ADP	13	Phosphoglycerokinase
Inverted receptor, causes formation of more glucose for ATP production when ADP is high.		
Muscle Tissue	14	180
<i>snipped note about the history of this one – it's all available on dejanews for those who are curious.</i> Senses high levels of muscle tissue and metabolises them, and stops catabolism of muscle tissue when levels are low.		

TABLE 9: Regulator Lobe Loopbacks



8.12 What are Dendrites ?

Dendrites are the links between cells in different brain lobes. They are the means for transferring information from one lobe to another. To help describe what dendrites do I'll use the example of the Attention lobe in a Ron norn.

The Attention lobe has forty cell locations. One for each category of object available in Creatures. The cell with the highest output value in this lobe will indicate the object that the norn is currently looking at (ie. paying attention to). The values for each cell are obtained from other lobes. The two lobes used as inputs to this lobe are the Stim Source lobe and the Noun lobe. Each of these two lobes also have forty cell locations. When a cell fires in one of these input lobes the result is carried down dendrites to the equivalent cell in the Attention lobe. The value of each cells from the two input lobes are summed and the result stored in the corresponding cell of the attention lobe. For example, if Cell 2 of Stim Source fires at value 50, Cell 3 of Stim Source fires at value 20, Cell 3 of Noun fires at value 40 then the result in the Attention lobe will be Cell 2 firing at value 50 and Cell 3 firing at value 50. How do the values of the cells get transferred to the Attention lobe? By traveling along dendrites. Dendrites are the electrical wiring between lobes.

If you look at the genetics kit dialog box for the Attention lobe you will see 4 tabs related to dendrites. Choose the first one 'D0 growth' and examine the information there. It tells us that the source lobe for these 'D0' (class 0) dendrites is the Stim Source lobe. That is, there is wiring that transfers information from each cell in the Stim Source lobe to cells in the Attention lobe.

The dendrite properties section tells us how many dendrites there are. In this case there is a minimum of '1' and a maximum of '1' spread out using a 'flat' pattern. This means that for each cell in the Stim Source lobe there will be one dendrite connecting it to the equivalent cell in the Attention lobe.

For now we won't worry about the other information. We'll create a norn with two new brain lobes, wire up the two lobes with different dendrite combinations and view the results.

8.13 State Variable Rules – Overview

The meanings of all the state variable rules in Creatures 1 and 2 and summarised in the table below. The functionality of these rules have been obtained through experimentation. As a result some of the information may be incorrect or incomplete but to the best of my knowledge it is accurate.

Opcode	Description	Creatures Version
end	Marks the end of an SVRule. Any opcodes appearing after this marker are ignored.	1.x
0	The integer number zero. Can be used for calculations.	1.x
1	The integer number one. Can be used for calculations. For example, adding or subtracting one from the current state.	1.x
32	The integer number thirty-two. Can be used for calculations. For example, adding or subtracting one from the current state.	2.x
64	The integer number sixty-four. Can be used for calculations. For example, adding or subtracting sixty-four from the current state.	1.x
128	The integer number 128. Can be used for calculations. For example, adding or subtracting one from the current state.	2.x
255	The integer number 255. Can be used for calculations. For example, adding or subtracting 255 from the current state.	1.x

Opcode	Description	Creatures Version
chem 0	Represents the current amount of chemical 0 in the brain lobe. This chemical can be added to a lobe through a genetic receptor.	1.x
chem 1	Represents the current amount of chemical 1 in the brain lobe. This chemical can be added to a lobe through a genetic receptor.	1.x
chem 2	Represents the current amount of chemical 2 in the brain lobe. This chemical can be added to a lobe through a genetic receptor.	1.x
chem 3	Represents the current amount of chemical 3 in the brain lobe. This chemical can be added to a lobe through a genetic receptor.	1.x
chem 4	Represents the current amount of chemical 4 in the brain lobe. This chemical can be added to a lobe through a genetic receptor.	2.x
chem 5	Represents the current amount of chemical 5 in the brain lobe. This chemical can be added to a lobe through a genetic receptor.	2.x
state	Represents the current value of the state of the cell.	1.x
output	Represents the current output value of the cell.	1.x
thres	The value of 'Nominal Threshold' defined in 'Cell Body Dynamics'.	1.x
type 0	The sum of type 0 dendrites. The individual value of each dendrite for this type is summed and returned. The individual value of a dendrite appears to be calculated as : $value = sourcecell * (stw/255)$ where 'source cell' is the value of the cell that this dendrite is attached to from the source lobe and 'stw' is the dendrites current short term weight value.	1.x
type 1	The sum of type 1 dendrites.	1.x
anded 0	If all type 0 dendrites are firing then this will be the value of the sum of these dendrites. If any of the type 0 dendrites are not firing then this value will be 0.	1.x
anded 1	If all type 1 dendrites are firing then this will be the value of the sum of these dendrites. If any of the type 1 dendrites are not firing then this value will be 0.	1.x
input	To be defined.	1.x
conduct	To be defined.	1.x
suscept	Current susceptibility to reinforcement. This is the value calculated by the 'susceptibility' state variable rule on the dendrite dynamics pages of the genetics kit. This opcode should only appear in the svrules associated with dendrites (otherwise there is no way to determine what dendrite it is pulling the data from).	1.x
STW	The value of the short term weight setting for the given dendrite. This opcode should only appear in the svrules associated with dendrites (otherwise there is no way to determine what dendrite it is pulling the data from). The short term weight for a dendrite appears to be calculated using the following formula with a relaxation decay applied to it : $stw = ltw + (susceptibility/255) * reinforcement$	1.x
LTW	The value of the long term weight setting for the given dendrite. The long term weight acts as a rest state for short term weight. STW and LTW reduce towards each other with LTW moving slower than STW. This opcode should only appear in the svrules associated with dendrites (otherwise there is no way to determine what dendrite it is pulling the data from).	1.x
Strength	Current value of dendrite strength. This opcode should only appear in the svrules associated with dendrites (otherwise there is no way to determine what dendrite it is pulling the data from).	1.x
TRUE	If the previous opcode evaluates to TRUE (ie. not zero) then execute the remaining opcodes otherwise the value of state is zero and the SVRule completes.	1.x
FALSE	If the previous opcode evaluates to FALSE (ie. zero) then execute the remaining opcodes otherwise the value of state is zero and the SVRule completes.	2.x
PLUS	Adds the value of the following opcode to the value of the previous opcode.	1.x



Opcode	Description	Creatures Version
MINUS	Subtracts the value of the following opcode from the value of the previous opcode. For example 'state :MINUS :1' will subtract '1' from the current value of 'state'.	1.x
TIMES	Takes the left hand opcode multiplied by the right hand opcode and divides this by 256. The result of this calculation is the value of this opcode. For example : '64 :TIMES :thres' where 'thres' is 32 will be $32 \times 64 / 256 = 8$.	1.x
INCR	Returns the value of the previous opcode incremented by one. For example 'state :INCR' will add one to the current value of state.	1.x
DECR	Returns the value of the previous opcode decremented by one. For example 'state :DECR' will subtract one from the current value of state.	1.x
rnd const	Not yet known.	2.x
leak in	This appears to be related to the value of the 'back propogation' state variable rule. If any other lobe has this current lobe as a 'source lobe' in its dendrite settings then the value of 'leak in' for the current lobe is the value of the 'back propogation' svrule from the destination dendrite. This can be used to feed data back to a source lobe from destination lobes. I'm unsure as to whether this information (from multiple destination lobes) is summed as I've only tested it with single destination lobes. More information will come as I find out more about it.	2.x
leak out	This appears to be related to the value of the 'forward propogation' state variable rule. If the current dendrite has a 'forward propagation' svrule then the 'leak out' opcode is the value of that 'forward propagation' calculation.	2.x
curr src leak in	Not yet known.	2.x
multiply	Not yet known.	2.x
average	Not yet known.	2.x
move twrds	This opcode takes two arguments and operates on the value of the previous opcode. It will take the value of the previous opcode and move it towards the first argument by the value of (argument 1 - previous opcode) / (256 / argument 2). For example : 'Suscept :move twrds :255 :64' will return a value which is susceptibility moved towards 255 by a quarter of the difference. If Suscept was 128 then the calculation would return $128 + (255 - 128) / (256 / 64) = 160$. This opcode allows you to perform a fairly complicated calculation while only taking up a few of the SVRule opcode places.	2.x
random	This opcode takes two arguments. It returns a random number between the first argument and the second argument. For example : 'random :0 :chem 5 :PLUS :state' adds a random number between 0 and the level of chemical 5 to the current state.	2.x
unused	To be defined.	1.x
ERROR	To be defined.	1.x

TABLE 10: List of SVRules for Creatures 1 and Creatures 2