Contents

1 Introduction

Ellipsis is a well-established particle-in-cell finite element modelling program, written by Dr. Louis Moresi, and extended into 3D by Dr. Richard Albert and Dr Frederick Dufour. This document describes the use of Ellipsis and the format of its input file. Sections within the Ellipsis input file are divided into Novice and Advanced user options. This manual is also divided in such a manner with Beginner user options outlined in chapter 3 and the Advanced user options discussed in chapter 4. General overviews of each major section within the Ellipsis input file are given within each sub-heading in the Beginner chapter.

1.1 Type Conventions

Throughout this document, the following typesetting conventions are used:

- Text set in `typewriter` font is as it occurs in the input files.
- Text set in *italics* are not literal, but are replaced with the relevant key-word.

1.2 Acknowledgments

2 Overview

2.1 Axes and Model Geometry

The **X** axis extends horizontally toward the right, the **Z** axis extends vertically down, and the **Y** axis extends out of the screen (Only applicable when Geometry=cart3d in the input file). See Figure 1. This is a right-handed coordinate system.

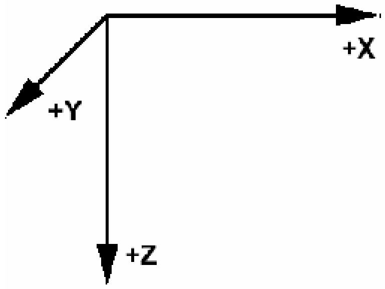


Figure 1: Ellipsis coordinate axes

2.2 Data Types

There are seven main data types for values.

- **Boolean:** A true/false value, either on or off. Numbers are also valid, with 0 being a synonym for off, and all other numbers synonymous with on.
- **Integer:** Decimal integer, optionally signed (e.g. 11, -7).
- **Real:** Decimal floating point, with optional sign and exponent (e.g. 100, -0.3, 1.6e-2 (= 1.6 x 10-2)).

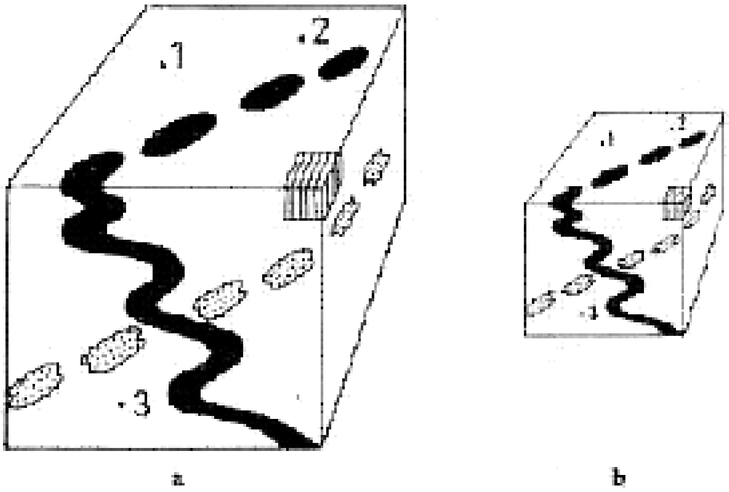


Figure 2: Illustration of two geometrically similar structures

- **String:** Any series of characters. If embedded spaces or tabs are desired, the value must be quoted (by using ”...” or ’...’).
- **List of Integers:** A comma-separated list of integers, each of which satisfy the above requirements for the Integer type (e.g. 4,7,-3,6).
- **List of Reals:** A comma-separated list of reals, each of which satisfy the above requirements for the Real type (e.g. 0.0,1.7,3.6e3).
- **List of Characters:** A comma-separated list of single characters (e.g. X,Y,W,B,F).

2.3 Units

In order to obtain computational efficiency when dealing with very large numbers, throughout Ellipsis almost all data is considered to be unit independent. That means that the units can be chosen arbitrarily, as long as consistency is maintained throughout the entire input. This is simply a process of **scaling** real world values into smaller numbers by using a **scaling factor** (see section 2.4).

2.4 Scaling and Non-Dimentionalisation

Models in Ellipsis can be scaled to be non-dimentional, that is as long as two systems are geometrically and dynamically similar, results from the one experiment can be applied to both systems no matter what the size of each system. Models are said to be geometrically similar when all lengths, areas and volumes are related by the same scaling factors and all angles between arbitrary lines are equal (see Figure 2). Models are said to be dynamically similar when the inertial, pressure, viscous and body (gravity) forces scale with the same scaling factors between the two systems.

The ellipsis model input parameters are scaled from real world values into dimensionless values in order to minimise computation time and increase the accuracy of the solution. We use the non-dimensional scaling approach given by:

$$E = N/S \tag{1}$$

where E is a dimensionless Ellipsis variable, N is the dimensional real world parameter and S is a dimensional scaling factor.

As an example, scaling from real world model length dimensions of 450km wide by 150km deep may be carried out using a length scaling factor (SL) of 1.5 x 10^5 m resulting in non-dimensional model geometry of 3 units wide by 1 unit deep.

Using a Thermal diffusivity scaling factor (S_κ) of 1 x 10^-6, a time scaling factor (S_t) can be found using:

$$S_t = S_L^2/S_\kappa \tag{2}$$

With a viscosity scaling factor (S_η) of 1 x 10^{21} and a gravity scaling factor (S_g) of 1 m/s^2, a density scaling factor (S_ρ) can be found:

$$S_\rho = \frac{S_g S_L S_t}{S_\eta} \tag{3}$$

A stress scaling factor (S_s) can now be defined as:

$$S_s = S_\rho S_g S_L \tag{4}$$

Using a velocity scale (S_u) defined by:

$$S_u = \frac{S_L}{S_t} \tag{5}$$

The strain rate scaling factor (S_ϵ) is subsequently given by:

$$S_\epsilon = S_t \tag{6}$$

Temperature is scaled between non-dimensional values of 0.17 and 1 corresponding to temperatures of 273K and 1603K respectively using a temperature scale (S_T) of 1603K.

Scaling factors for thermal properties heat generation (S_{Hgen}), thermal expansion coefficient (S_α) and heat flux (S_Q) can be defined as:

$$S_{Hgen} = S_{Cp} = \frac{S_s}{S_\rho S_T} \tag{7}$$

$$S_\alpha = \frac{1}{S_T} \tag{8}$$

$$S_Q = \frac{S_{Cp} S_\rho S_T S_L}{S_t} \tag{9}$$

3 Input Template

This section outlines *most* of the input variables that can used in the Ellipsis input template. The input file is plain text, consisting of a large list of token=value pairs. Blank lines are ignored, and everything following a `#` on a line is ignored as a comment. Values cannot be split over multiple lines. Where code from the input template is displayed below in `typewriter font` comments to the right describe what the variable is for, what type of value the variable expects (boolean, integer, etc.), possible options or data ranges for the variable and the default value. While this list is quite extensive there may still be variables not listed here in this manual (something that should hopefully change over time).

3.1 General

`DESCRIBE=off` (Boolean) Whether to describe the search for parameters. Default is `off`.

`VERBOSE=off` (Boolean) Whether to print out the input values as they are read in. Default is `off`.

`BEGINNER=off` (Boolean) Whether to be verbose when parameters are missing. Only applicable when `VERBOSE=on`. Default is `off`.

`verbose=off` (Boolean) Whether the code should be verbose about its behavior. Default is `off`.

`datafile=""` (String) The root name of the output file names, ie. the prefix to which all file extensions will be appended.

3.2 Advection-diffusion Parameters

`minstep=1` (Integer) Minimum number of steps in model. Default is `1`.

`maxstep=500` (Integer) Minimum number of steps in model. Default is 1000, though this will depend on how long you want the model to run

`Geometry=cart2d` (String) This defines the geometry of the model. The following options are available:
`cart2d` - 2D Cartesian
`cartpt5d` - 2.5D Cartesian
`cart3d` - 3D Cartesian
`axi` - Axisymmetric
`cylinder` - Cylidrical
`sphere` - Spherical

3.3 Solver Related Matters

`vel_relaxations=10` maximum number of velocity loops (default=10)

`piterations=100` maximum Uzawa iteration loops (default=100)

`viterations=20` number of velocity iterations before checking convergence (default=20)

`Solver=multigrid` multigrid or (later, conjugate gradient)

`mg_cycle=1` style of multigrid cycle
1 = V cycle, 2 = W cycle, ... (default=1)

`accuracy=` desired accuracy of Uzawa algorithm (default=1.e-4)

`delta_accuracy_factor=1.0` change in accuracy level->level (> 1 for nonN, < 1 for Newt)
0.001 < factor < 10.0 (default=0.2)

`gs_under_relax=1.0` Gauss-Seidel under-relaxation (default=1.0)

3.4 Model Design

In this section the model geometry is assigned and the resolution is then set.

3.4.1 Model Geometry

Here you define the geometry of your model. There is no default for this section, though values are usually scaled to be between 0 and 5.

`dimenx=3.0`	(Real) Length of model along the X axis
`dimenz=2.0`	(Real) Length of model along the Z axis
`dimeny=1.0`	(Real) Length of model along the Y axis (only relevant for when `Geometry=cart3d` is selected)

3.4.2 Resolution

Resolution of models in Ellipsis is handled using a **multigrid**. This is where a solution is converged upon using a coarsely meshed model (top image in figure below) and this solution is used as input for a more finely gridded mesh and so on. This is a very computationally efficient method to solve for high resolution problems. As illustrated in Figure 3 at each multigrid level, each cell on the coarse mesh is divided into 4 cells on the finer mesh. For example, a level 1 coarse mesh with 4 cells will be divided into a level 2 mesh with 16 cells, a level 3 mesh with 64 cells and so on. Each of these cells are then filled with a user specified number of tracers or particles (now you understand the term particle in cell modelling and can feel comfortable bringing it up at a dinner party!). If the user has specified a tracer density of 4, there will be 16 tracers per element/cell since the value for tracer density refers to how many tracers in each axis direction. For example, a tracer density of 4 means create a matrix of 4 tracers in the X direction and 4 tracers in the Z direction (for a 2D model) resulting in a matrix of 16 tracers within the element. Thus a level 1 mesh with 4 cells will contain a total of 64 tracers, the level 2 mesh with 16 cells will contain 256 tracers, the level 3 mesh with 64 cells will contain 1024 tracers, etc.

The resolution can then be changed in two ways, by either increasing the base mesh resolution (increasing the model resolution) or by increasing the number of multigrid levels (increasing model resolution).

`# Grid Mesh and Levels`	
`mgunitx=5`	(Integer) Coarsest (base) multigrid cell dimensions (elements) along the X axis
`mgunitz=3`	(Integer) Multigrid cells along the Z axis. The default/minimum is 2.
`levels=4`	(Integer) Number of multigrid levels. The default is `1`.

`# Tracer setup`	
`Tracers=on`	initialise tracers (default=`on`)
`Tracer_appetite=0.5`	(size(tracer1)+size(tracer2)) x Tracer_appetite (default=`0.5`)
`Tracer_voids=off`	allow tracers to disappear (default=`off`)
`Tracer_rect=1`	number of rectangular regions of different tracer densities (default=`0`)
`Tracer_rect_density=4`	tracer density (N x N per finest element) (<=12)
`Tracer_rect_x1=0`	the coordinate extent of region should be same size as the model box
`Tracer_rect_x2=dimenx`	
`Tracer_rect_z1=0`	
`Tracer_rect_z2=dimenz`	

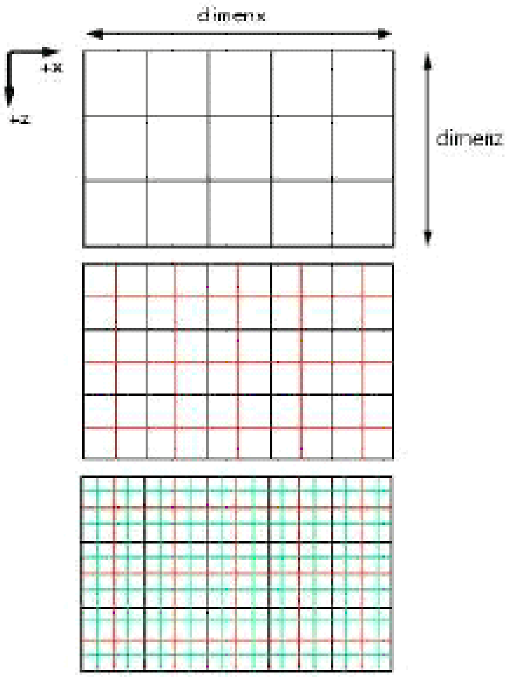


Figure 3: Illustration of the multigrid meshing process

3.5 Material Properties

For each material you wish to use in you Ellipsis model you must assign numerous properties. These include (but are not limited to) the Ellipsis variables in the following sections.

3.5.1 General Properties

`Material_1_density=`	density (default=1.0)
`Material_1_porosity=`	initial porosity (default=0.0) NB: initial porosity = 0 ensures that Bulk_visc is constant
`Material_1_Bulk_visc=`	bulk visc ratio at initial porosity (>1.0(2D), >2/3(3D)) bulk visc = ratio*visc (default=-1.0=infinite) div(v) + p/(bulk_visc-2/3*visc) = 0
`Material_1_Bulk_modulus=`	B in dp = B*div(v)*dt (slightly compressible formulation) (default=0.0) where dp is on tracers (isotropic stress)
`Material_1_reproduction=on`	allow tracer reproduction (default=on)
`Material_1_phases=1`	number of unique phases (first phase is phase 0) (default=1) then visc = [sum(1/visc_n)] ˆ(-1)
`Material_1_T_block=`	(blocking) T above which phase change can occur (default=-1.e32)
`Material_1_rheol_cpts=`	number of rheological components (at least one per phase) (default=1) then visc = [sum(1/visc_n)] ˆ(-1)
`Material_1_Trange_min=-1.e8`	temperature range to which rheology applies
`Material_1_Trange_max=1.e8`	(default=-1.e8,1.e8)
`Material_1_rheol_phase=0`	phase to which each rheology applies (start from 0) (default=0)

3.5.2 Colouring

`# Colouring`	
`Material_1_Red=`	RGB values for "cold" material (list one per PPM file)
`Material_1_Green=`	("hot" and "cold" are determined from T extremes)
`Material_1_Blue=`	
`Material_1_Opacity=`	opacity for "cold" material (negative=off)
`Material_1_Red_hot=`	values for "hot" material
`Material_1_Green_hot=`	
`Material_1_Blue_hot=`	
`Material_1_Opacity_hot=`	
`Material_1_Red_strained=`	values for strained material
`Material_1_Green_strained=`	
`Material_1_Blue_strained=`	
`Material_1_Opacity_strained=`	

3.5.3 Rheological Properties

`# Rheological model`	
`Material_2_rheol_T_type=1`	rheological temperature-dependence model (default=2) (1) visc=N0*exp(-T1*T) (Frank-Kamenetski) (2) visc=N0*exp (E+Z*z)/(T1*(T+T0)) (Arrhenius) where z=depth
`Material_2_viscN0=`	N0 in viscosity models (default=1.0)
`Material_2_viscT1=`	T1 in viscosity models (default=1.0)
`Material_2_viscT0=`	T0 in Arhennius viscosity model (default=0.0)
`Material_2_viscZ=`	Z in Arrhenius viscosity model (default=0.0)
`Material_2_viscE=`	E in Arrhenius viscosity model (default=0.0)
`Material_2_viscTmax=`	maximum and minimum T to use in calculating viscosity
`Material_2_viscTmin=`	(default=1.e32,0.0)
`Material_2_sdepv_expt=`	exponent ”s” in stress dependance of viscosity

`# Stress-strain relationship`	
`Material_2_yield_stress_minimum=`	minimum yield stress for plastic deformation (default=1.e-32)
`Material_2_yield_stress_maximum=`	maximum yield stress for semi-brittle effect (default=1.e32)
`Material_2_yield_stress_B0=`	”cohesion” B0 in above eqn (default=1.e32)
`Material_2_yield_stress_Bp=`	”friction coefficient” Bp in above eqn (default=0.0)
`Material_2_yield_stress_Ea=`	ratio Ea = f(0,0)/f(E0,0) (default=1.0,range=[0,1])0
`Material_2_yield_stress_E0=`	strain weakening E0 (default=1.e32)
`Material_2_yield_stress_En=`	exponent En in f(e), e¡E0 (default=0.0)
`Material_2_yield_stress_Bz=`	”friction coefficient” Bz in above eqn (default=0.0)
`Material_2_yield_stress_Bc=`	tension cutoff Bc in above law (default=1.e32)
`Material_2_yield_stress_Edota=`	ratio Edota = f(0,0)/f(0,Edot0) (default=1.0,range=[0,1])
`Material_2_yield_stress_Edot0=`	strain rate weakening Edot0 (default=0.0)
`Material_2_yield_stress_Edotn=`	exponent Edotn in f(e), edot¡Edot0 (default=0.0)
`Material_2_yield_stress_ET=`	T above which strain weakening is reset (default=1.e32)
`Material_2_yield_stress_E0dt=`	time rate of strain reduction (healing)

3.5.4 Thermal Properties

`# Thermal parameters`	
`Material_2_therm_exp=`	thermal expansion coefficient (default=0.0)
`Material_2_therm_diff=`	thermal diffusivity (default=0.0)
`Material_2_Cp=`	isobaric heat capacity (default=1.0)
`Material_2_Qt=`	internal heating rate by mass (default=0.0)

3.6 Assign Material Distributions

Once materials have been defined in your Ellipsis template you can assign rectangular, triangular or circular regions of a specified material type.

3.6.1 Rectangles

`Material_rect=`	number of rectangular regions with different tracer properties (default=0)
`Material_rect_property=`	tracer group name (propertiescolour)
`Material_rect_x1=`	coordinates of tracer regions
`Material_rect_x2=`	(successively overwritten if rectangles overlap)
`Material_rect_z1=`	
`Material_rect_z2=`	

3.6.2 Triangles

```
Material_trgl=
Material_trgl_property=
Material_trgl_x1=
Material_trgl_x2=
Material_trgl_x3=
Material_trgl_z1=
Material_trgl_z2=
Material_trgl_z3=
```

3.6.3 Circles

```
Material_circ=
Material_circ_property=
Material_circ_x=
Material_circ_z=
Material_circ_rad=
```

3.7 Assign Pre-Strained Region Distributions

In Ellipsis you can assign rectangular, triangular or circular regions with arbitrary strain.

3.7.1 Rectangles

3.7.2 Triangles

`Strain_trgl=`	Number of triangular regions of arbitrary strain
`Strain_trgl_x1=`	X coordinate of first triangle vertex
`Strain_trgl_z1=`	Z coordinate of first triangle vertex
`Strain_trgl_x2=`	
`Strain_trgl_z2=`	
`Strain_trgl_x3=`	
`Strain_trgl_z3=`	
`Strain_trgl_mag=`	Magnitude of strain (range=[0,1])

3.7.3 Circles

3.8 Initial Conditions

`gravacc=10`	gravitational acceleration (default=9.81)

3.8.1 Thermal conditions

`toptbcval=0 bottbcval=1`	initial temperature gradient (defaults=0.0,1.0)
`num_perturbations=0`	
`perturbmag=`	magnitude of T perturbation
`perturbk=`	wavenumber in x direction (will get multiplied by pi)
`perturbky=`	wavenumber in y direction

3.8.2 Temperature field distributions

`Temp_rect=1`	Number of rectangular temperature regions (default=0)
`Temp_rect_x1=`	coordinates of region
`Temp_rect_x2=`	
`Temp_rect_z1=`	
`Temp_rect_z2=`	
`Temp_rect_hw=`	half-width of smoothed edge
`Temp_rect_mag=`	magnitude of initial condition
`Temp_rect_ovl=`	A/M/R = add/multiply/replace overlaps (in increasing coord direction)

3.8.3 Previous Conditions

`previous_temperature_file=""`	initial temperature configuration to use
`particle_input=""`	initial particle configuration (XDR binary)
`previous_particle_data=""`	initial particle properties to use
	possible keywords for binary_data and particle_data:
	Temp = temperature
	Pres, PorP = solid pressure, pore pressure
	Visc = viscosity
	Poro = porosity
	Edot = strain rate
	StrP, StrT = integrated plastic/total strain
	Grsz = grain size

3.9 Boundary Conditions

3.9.1 Moving Boundaries

`BCmoveX0v=1.0`	Left hand boundary. Positive = extensional, negative = compressional
`BCmoveX1v=1.0`	Right hand boundary

3.9.2 Temperature

`Temp_bc_rect=`	number of rectangular bc ranges (surfaces) (default=0)
`Temp_bc_rect_norm=`	normal to plane of surface (X, Z, Y)
`Temp_bc_rect_icpt=`	normal-axis intercept of bc plane
`Temp_bc_rect_aa1=`	lateral coordinate extent in 1st dimension
`Temp_bc_rect_aa2=`	
`Temp_bc_rect_hw=`	half-width of bc smoothing edge
`Temp_bc_rect_mag=`	magnitude of bc

3.9.3 Velocity

`Velocity_z_bc_rect=`	number of rectangular bc ranges (surfaces) (default=0)
`Velocity_z_bc_rect_norm=`	normal to plane of surface (X, Z, Y)
`Velocity_z_bc_rect_icpt=`	normal-axis intercept of bc plane
`Velocity_z_bc_rect_aa1=`	lateral coordinate extent in 1st dimension
`Velocity_z_bc_rect_aa2=`	
`Velocity_z_bc_rect_hw=`	half-width of bc smoothing edge
`Velocity_z_bc_rect_mag=`	magnitude of bc

3.9.4 Other

`periodicx=off`	wrap-around bc (2D only) (default=off)
`periodic_rm_vx=off`	removes periodic bc up to a constant (default=off)
`free_upper=off`	free upper surface (pseudo) (default=off)
`free_lower=off`	free lower surface (pseudo) (default=off)
`initial_isostasy=off`	set initial topography from stress-balance (default=off)

3.10 Output Files

Output from Ellipsis can either be textual or graphical. The use may define the number of timesteps between output of data and toggle compression of the output data files with the following lines:

`storage_timesteps=1`	data writing interval (based on average timestep) (default=50)
`checkpt_timesteps=1`	PPM (graphics) file writing interval (default=10)
`COMPRESS=off`	output files compressed upon creation (default=on)

Various data types can be output from Ellipsis runs, including data realated to temperture, stress, strain rate and velocity.

`datatypes="Temp,Pres,Pstn"`	desired nodal output variables (ascii file) possible keywords for datatypes: Velx, Vely, Velz = x, y, z velocity PoVx, PoVz = x, z pore liquid velocity Pres, PorP = (nodal) solid pressure, pore pressure Temp = temperature, Comp = compression Strf = stream function (2D cartesian coords only) Pbdy = phase boundary

`particle_data=""`	desired particle output variables (binary file)
	possible keywords for particle_data:
	Temp = temperature
	Pres, PorP = solid pressure, pore pressure
	Visc = viscosity
	Poro = porosity
	Edot = strain rate
	StrP, StrT = integrated plastic/total strain
	Grsz = grain size

`averages=""`	horizontally averaged values for output (ascii)
	possible keywords for averages:
	Temp = temperature
	Visc = viscosity
	Poro = porosity
	Velo = magnitude of velocity
	Urms, Vrms, Wrms = rms of x, z, y velocities

`timelog=""`	time record of large-scale averages (ascii) possible keywords for timelog: Nuss = Nusselt number Vrms, Vxrm, Vyrm, Vzrm = total and directional rms velocities Shfl, Bhfl = average surface and basal heat fluxes Svav, Bvav = surface and basal vrms

`observables=""`	surface observables (slice at z=0/zmax) (ascii) possible keywords for observables: Shfl, Bhfl = surface and basal heat fluxes Vxsf, Vysf, Vzsf = surface velocities Tpgx, Tpbx = surface and basal topography Tpgk, Tpbk = surface and basal topography wavenumbers Grvx, Grbx, Grtx = surface, basal, and topographical gravity signals Grvk, Grbk, Grtk = wavenumbers of above Geox, Gebx, Getx = surface, basal, and topographical geoid Geok, Gebk, Getk = wavenumbers of above

3.11 Graphical Specifications of Output Files

`PPM_files=`	number of PPM files at each output step (default=1) & first PPM file is *.ppm0, etc.
`PPM_height=`	vertical size of output PPM file (default=256)
`PPM_coloring=`	variable upon which to base colouring possible choices for PPM_coloring: (default=1) 1=temperature, 2=viscosity, 3=stress=visc*edot, 4=solid pressure, 5=grainsize, 6=compression, 7=strain rate, 8=accumulated strain 9=pore pressure, 10=permeability, 13=melt production
`PPM_coloring_autorange=`	automatically scale colour (default=1)
`PPM_coloring_min=`	min value for color scale (default=0.0)
`PPM_coloring_max=`	max value for color scale (default=1.0)
`PPM_show_strain=`	colour according to actual strength change (default=0.0)

3.12 Profile/History Extraction

During an Ellipsis run you may specify to have certain data recorded at localised points in the model, called *Sampling Tracers*. To create a Sampling Tracer the user must define the number of *Sampling Tracers* to be included in the model, whether the sampling Tracer will be Eulerian (stay fixed to the initial grid position to which you assigned it) or Lagrangian (moves within the fixed mesh), the output file for the information to be displayed in and the initial *XYZ* location of the *Sampling Tracer*. Numerous different types of data can be recorded at individual *Sampling Tracers*, such as temperature, velocities and stress/strain. The type of data you which to record is designated using a numbered class system assigned to `the Sampling_field` variable, for example temperature corresponds to number 1 (see the code and comments below for full list of available variables).

`Sampling_tracers=`	number of sampling tracers (default=0)
`Sampling_lagrangian=`	(fixed) Eulerian=0 (Stay fixed with mesh) Lagrangian=1 (Move within mesh) (default=0)
`Sampling_plot_num=`	PPM file in which profile is stored (default=0)
`Sampling_x=`	initial x location (default=0.0)
`Sampling_z=`	initial z location (default=0.0)
`Sampling_y=`	initial z location (default=0.0)
`Sampling_field=`	field to sample (default=0) 1=temperature, 2=x velocity, 3=z velocity 4=nodal pressure, 5=strain rate, 6=stress=visc*edot 14=depl
`Sampling_dirn=`	profile direction (1=x, 2=z, 3=y) (default=0)
`Sampling_normalize=`	0=unnormalized, 1=normalized (default=0)
`Sampling_plot_min=`	If not autoranging, then need a max/min for the scale
`Sampling_plot_max=`	(default=1e4)
`Sampling_R=`	RGB colour mixes for profiles scaled between 0 and 1 (255 real RGB terms) (default=0.0)
`Sampling_G=`	
`Sampling_B=`	

3.13 Scaling

If you which to edit your input template in the Ellipsis GUI and you want the GUI to be aware of the scaling parameters you have used to scale the parameters in the input file, you must include the following comments in your input template:

`# Ellipsis GUI parameter:ellipsisScalingModel=""`
Scaling model to use (default=USIMS EluScaling 1.0)
`# Ellipsis GUI parameter:ellipsisScaling_Length=`
Length scaling factor
`# Ellipsis GUI parameter:ellipsisScaling_Gravity=`
Gravity scaling factor (default=1.0)
`# Ellipsis GUI parameter:ellipsisScaling_ThermalDiffusity=`
Thermal Diffusivity scaling factor
`# Ellipsis GUI parameter:ellipsisScaling_Temperature=`
Temperature scaling factor
`# Ellipsis GUI parameter:ellipsisScaling_Viscosity=`
Viscosity scaling factor

3.14 God Switches

The *God Switches* are simply switches which let you turn on or off certain aspects of the code which would otherwise be tedious to do so. For example to turn off the temperature dependent viscosity behaviour of the code you can simply set `TDEPV=off`, a task that would otherwise involve removing or commenting out all variables relating to temperature dependent viscosity for all materials.

`TDEPV=on`	use temperature-dependent rheological parameters (default=on). Note: off is faster than turning all viscosity values to 1
`VMAX=off`	use maximum viscosity (default=off)
`VMIN=on`	use minimum viscosity (default=off)
`visc_max=`	maximum, minimum viscosity cut-offs (no defaults)
`visc_min=`	
`SDEPV=off`	use stress dependence of viscosity (default=off)
`GRDEPV=off`	use grain size dependence of viscosity (default=off)
`YIELD=on`	yield stress parameters on/off (default=off)

4 Graphical User Interface

While the previous sections of this documentation outlined the specifics of the Ellipsis text input file, in order to allow easier use of the Ellipsis modelling software a Graphical User Interface (GUI) was designed at the University of Sydney School of Earth Sciences. The GUI is open-source, written in Java and therefore platform independent and sufficiently generic such that it could easily be adapted to other modeling codes with similar objectives. The GUI window is composed of several tabs, with each tab referencing a particular aspect of creation of an Ellipsis model.

4.1 Hardware and Software Requirements

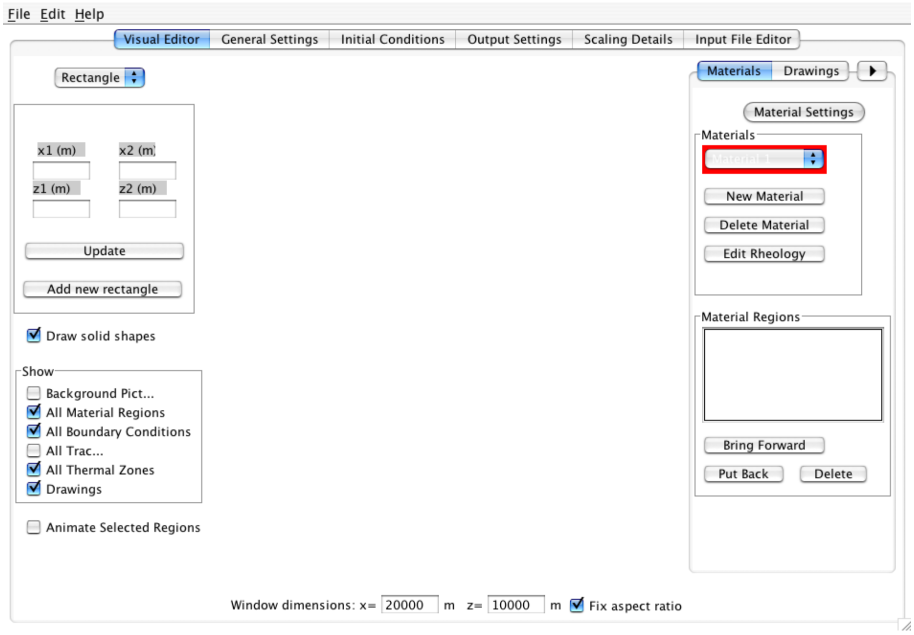
Ellipsis GUI was implemented in Java, and so can be run on a range of different platforms without modification or recompilation, using the Java virtual machine. The system has been tested on computers running the Windows, Solaris, Linux and Macintosh operating systems. The only requirements are that the computer has version 1.4.2 or higher of Java installed, has enough memory, and has a display with a screen size large enough to allow the entire main window to be visible at one time. However, all modern computers should satisfy the necessary hardware requirements. The program has been tested on a PC which had only had 32 megabytes of memory, and which was running Windows 98, and it worked without any problems.

4.2 Visual Editor

In the *Visual Editor* tab of the Ellipsis GUI the model geometry and material settings are assigned. The following is an outline of each section and the model parameters that can be assigned in each.

4.2.1 Model geometry

In the main *Visual Editor* window the user can set the size of the box used in the Ellipsis run by assigning values (in meters) to the window dimension at the bottom of the window show in Figure 4. Using the mouse or the boxes labeled *x1*, *x2*, *z1*, and *z2* (Figure 4), material regions can be defined by either rectangles, triangles or circles. Material regions can be edited by selecting the appropriate region from the *Material Regions* list in the lower right corner of the *Visual Editor* window.



☑ Draw solid shapes

Show

□ Background Pict...
☑ All Material Regions
☑ All Boundary Conditions
□ All Trac...
☑ All Thermal Zones
☑ Drawings

□ Animate Selected Regions

Window dimensions: x= 20000 m z= 10000 m ☑ Fix aspect ratio

Figure 4: Ellipsis GUI Visual Editor tab

4.2.2 Material properties

While in the *Materials* tab of the menu in the top left of the *Visual Editor* window you can set both the parameters general to all materials (using the *Material Settings* button) and also the rheological parameters of each material (using the *Edit Rheology* button).

4.2.3 Boundary Conditions

In the *Boundary Conditions* tab the user may assign various boundary conditions to the model, such as a fixed velocity along the left or right (or both) margins of the model, regions of thermal anomaly, heat flux conditions, localised stress/strain/veocity conditions etc.

4.2.4 Tracer setup

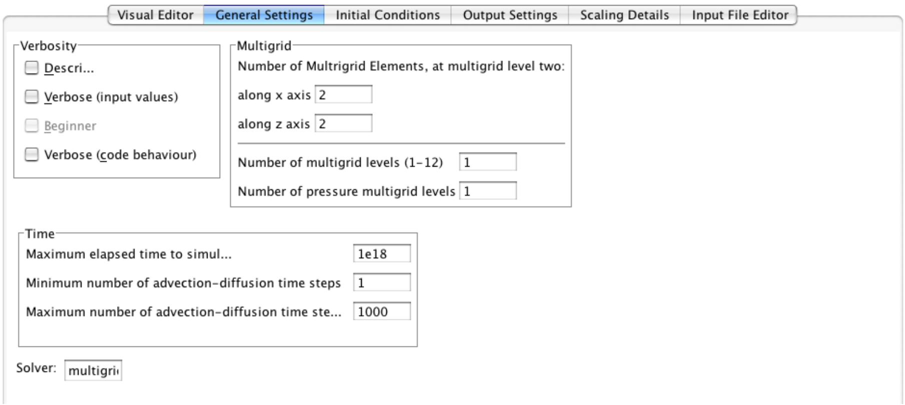
In the *Tracers* tab the user may define the density of tracers within each grid cell of the model simulation (see section 3.4.2).

4.2.5 Thermal zones

In the *Thermal Zones* tab the user may define regions of thermal anomaly using the GUI.

4.3 General Settings

In the *General Settings* tab the user can define what level of verbosity the model will run with (see section). You can also define the resolution of the base grid used for the multigrid analysis along with the total time the model may run for.



Verbosity

☐ Descri...

☐ Verbose (input values)

☐ Beginner

☐ Verbose (code behaviour)

Multigrid

Number of Multrigrid Elements, at multigrid level two:

along x axis 2

along z axis 2

Number of multigrid levels (1-12) 1

Number of pressure multigrid levels 1

Time

Maximum elapsed time to simul... 1e18

Minimum number of advection-diffusion time steps 1

Maximum number of advection-diffusion time ste... 1000

Solver: multigri

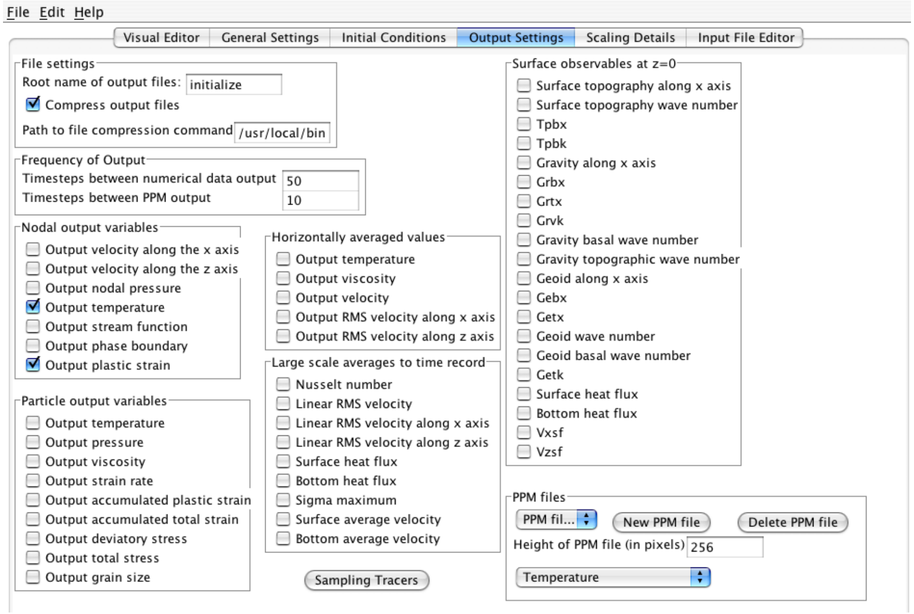
Figure 5: Ellipsis GUI General Settings tab

4.4 Initial Conditions

The user may define the gravitational acceleration and angle of acceleration in the *Initial Conditions* tab.

4.5 Output Settings

In the *Output Settings* tab the user may define numerous different types of output (both textual and graphic) they may require.



File settings

Root name of output files: initialize

☑ Compress output files

Path to file compression command /usr/local/bin

Frequency of Output

Timesteps between numerical data output 50

Timesteps between PPM output 10

Nodal output variables

☐ Output velocity along the x axis

☐ Output velocity along the z axis

☐ Output nodal pressure

☑ Output temperature

☐ Output stream function

☐ Output phase boundary

☑ Output plastic strain

Particle output variables

☐ Output temperature

☐ Output pressure

☐ Output viscosity

☐ Output strain rate

☐ Output accumulated plastic strain

☐ Output accumulated total strain

☐ Output deviatory stress

☐ Output total stress

☐ Output grain size

Horizontally averaged values

☐ Output temperature

☐ Output viscosity

☐ Output velocity

☐ Output RMS velocity along x axis

☐ Output RMS velocity along z axis

Large scale averages to time record

☐ Nusselt number

☐ Linear RMS velocity

☐ Linear RMS velocity along x axis

☐ Linear RMS velocity along z axis

☐ Surface heat flux

☐ Bottom heat flux

☐ Sigma maximum

☐ Surface average velocity

☐ Bottom average velocity

Surface observables at z=0

☐ Surface topography along x axis

☐ Surface topography wave number

☐ Tpbx

☐ Tpbk

☐ Gravity along x axis

☐ Grbx

☐ Grtx

☐ Grvk

☐ Gravity basal wave number

☐ Gravity topographic wave number

☐ Geoid along x axis

☐ Gebx

☐ Getx

☐ Geoid wave number

☐ Geoid basal wave number

☐ Getk

☐ Surface heat flux

☐ Bottom heat flux

☐ Vxsf

☐ Vzsf

PPM files

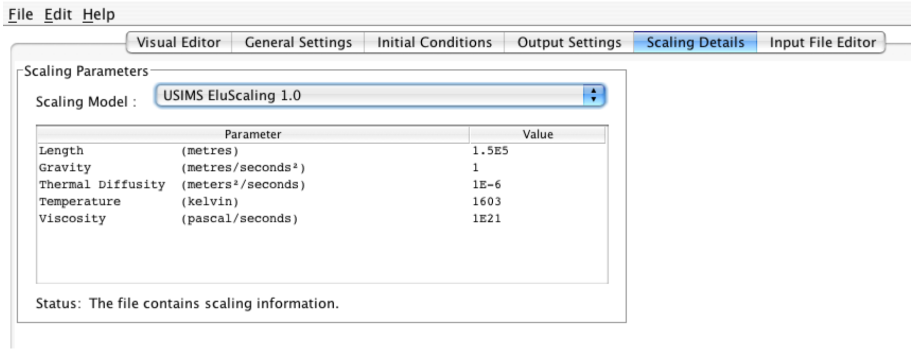
Height of PPM file (in pixels) 256

Temperature

Figure 6: Ellipsis GUI Output Settings tab

4.6 Scaling Details

All parameters relating to scaling (see section 2.4) can be defined in the *Scaling Details* tab. The user may select between scaling model parameters or using no scaling (input template will contain values entered into the GUI with no scaling applied) via a drop down menu. Since the GUI is open source there may be numerous different scaling models available and users are encouraged to add more, however, the default model is the *USIMS EluScaling* model. parameters shown in the windows available for editing by the user are the five independent variables which are used to define all subsequent scaling parameters.



Scaling Parameters

Scaling Model : USIMS EluScaling 1.0

Parameter		Value
Length	(metres)	1.5E5
Gravity	(metres/seconds²)	1
Thermal Diffusity	(meters²/seconds)	1E-6
Temperature	(kelvin)	1603
Viscosity	(pascal/seconds)	1E21

Status: The file contains scaling information.

Figure 7: Ellipsis GUI Scaling Details tab

When an input template *not* created by the GUI is loaded into the GUI, the scaling will automatically be turned off since the GUI will have no knowledge of the scaling used to create the input file.

4.7 Input File Editor

The *Input File Editor* shows the Ellipsis input file created by the GUI. In this section the user may for instance add parameters to the Ellipsis input file not available to them in the GUI. It is also possible to alter parameters in the Ellipsis input file in the *Input File Editor* and have those changes reflected in the GUI (and vice versa).