The GenGIS Manual

From The GenGIS wiki

Welcome to the GenGIS manual. The sections below provide enough information to configure and start using GenGIS. Stay tuned for updates and send us your feedback when you get an opportunity.

Contents

Introduction / Overview of GenGIS

Purpose

Geography has always been an important component of evolutionary and ecological theory. The advent of sequence typing approaches such as 16S ribotyping, DNA barcoding using the COX1 gene, and multi-locus sequence typing, gives us the opportunity to understand how communities of organisms interact, disperse and evolve. This sequencing revolution is tightly coupled to the development of new algorithms for assessing and comparing populations based on their genes.

Coupled with these developments is the availability of high quality, public domain digital map data. By integrating molecular data with cartography and habitat parameters, we can visualize the geographic and ecological factors that influence community composition and function.

GenGIS is designed to bring these components together into a single software package that satisfies the following criteria:

Free and Open Source

> GenGIS is released under a **Creative Commons Attribution - Share Alike 3.0** license (http://creativecommons.org/licenses/by-sa/3.0/) , and we have made extensive use of other free packages such as wxWidgets, R, and Python. Making GenGIS freely available allows it to be downloaded and used anywhere in the world, and allows users to inspect and modify the source code.

User-Friendly Interface

> Although GenGIS is built to deal with challenging scientific questions, our goal is to make the software easy to use. This is particularly important as many users will have little experience with digital map data, apart from applications such as Google Earth.

Adaptible and Extensible

> The principal strength of many open-source projects lies in the ability of a loosely organized community of users to develop and enhance the software: R and BioPerl are two examples of successful open-source projects with many contributors. Since the potential applications of GenGIS are much broader than those we have in mind, we aim to make it as easy as possible to extend its capabilities by exposing the internal data structures and offering a plugin architecture.

Citing GenGIS

The best citation for GenGIS is indicated in boldface on the Main GenGIS page (http://kiwi.cs.dal.ca/~beiko/mediawiki/mediawiki-1.13.4/index.php) .

Where to go for Help

- The latest version of the GenGIS manual (http://kiwi.cs.dal.ca/~beiko/mediawiki/mediawiki-1.13.4/index.php?title=The_GenGIS_Manual) .
- Text and video tutorials are available on the Tutorials (http://kiwi.cs.dal.ca/GenGIS/GenGIS_Tutorials) page.
- The FAQ (http://kiwi.cs.dal.ca/~beiko/mediawiki/mediawiki-1.13.4/index.php?title=The_GenGIS_Manual) page keeps track of GenGIS-related questions.
- Please email *beiko [at] cs.dal.ca* with any questions or feedback about the software.

Installation

Getting the Latest Version of GenGIS

Download the latest version (http://kiwi.cs.dal.ca/GenGIS/Download) for Windows or Mac to get started visualizing and analyzing data in GenGIS.

Developer Version – building from source code

The source code for GenGIS is available on the Download (http://kiwi.cs.dal.ca/GenGIS/Download#Source_Code) page.

Building on Windows	GenGIS can be compiled using Microsoft Visual C++ 2008 Express Edition (http://www.microsoft.com/visualstudio/en-us/products/2008-editions/express) . The Visual Studio solution file (*GenGIS.sln*) is located in the 'win32/build/msvc' directory. Please note that the built-in Python console is only available in Release builds.

Building on MacOSX	GenGIS can be compiled using the Makefile located in the 'mac/build/gcc' directory. To compile, simply run 'make' within a terminal. Development has been performed using the gcc 4.3.3 (http://gcc.gnu.org/) compiler.

System Requirements

GenGIS has been developed and tested on the following operating systems:

- **Windows** (XP, Vista & 7) 32-bit binaries compatible with 64-bit Windows releases.
- **Mac OS X** (v10.5 'Leopard' & v10.6 'Snow Leopard') Intel-based only.

Support for Linux is not a development priority at this time. Porting to Linux should be a fairly straightforward operation as GenGIS has been developed using cross-platform libraries. Efforts to port GenGIS to other operating systems are encouraged. As always, your feedback on this project is greatly appreciated.

Input data

Data File Types

GenGIS works with four different types of data files:

1. Digital Map File (sample)
2. Location File (sample)
3. Sequence File (sample)
4. Tree File (sample)

Data files should be loaded in the following orders:

```
Map → Location → Sequence → Tree
```

or

```
Map → Location → Tree
```

GenGIS currently supports a single map, location and sequence file during a session. Any number of tree files can be opened.

Note: The location of sequences (sequence file) and leaves (tree file) must each map to an existing location given by the location file. The leaves of a tree file can be either locations or sequences.

Maps

GenGIS relies on the Geospatial Data Abstraction Library (GDAL) to import several digital map file formats and projections. More information, including community support and downloads, can be found at the GDAL website (http://www.gdal.org/) .

Note: The 'gdal_merge.py' script, 'gdalwarp.exe' and 'gdal_translate.exe' executables have been very useful in preparing maps compatible with GenGIS.

Supported formats

GenGIS supports the formats listed on the GDAL Raster Formats (http://www.gdal.org/formats_list.html) page. Note that not all formats have been tested at this time. The following formats have been found to work reliably:

- GeoTIFF
- Arc/Info ASCIIGRID
- USGS DEM (and variations thereof)

Projections

If you wish to use a specific projection (http://en.wikipedia.org/wiki/Map_projection) , you must specify it before loading your map -

GenGIS is unable to render reprojections on the fly. This is particularly true if you are loading the default world map (from GTOPO30) that ships with GenGIS: the default Mercator projection stretches the polar regions, whereas Plate Carre or Robinson will provide a much less distorted world view.

To specify the projection before loading the map, right click "New Study : Study" in the *Layers* tab, and select *Properties*. Selecting the Projection tab will allow you to choose your projection.

GenGIS currently does not support projections in which a single point is displayed in multiple locations. The best example of this is the default world map, which is modified to stretch from 89.9 degrees North to 89.9 degrees South latitude. Since the poles stretch across the entire upper and lower edges of a map in a projection such as Plate Carre, GenGIS is unable to display these properly.

Typical limits on map size

Higher processor speeds and more system RAM are required to work effectively with larger maps. Typically one gigabyte of RAM permits working with maps that are 10 megabytes or slightly greater in size.

Reducing Map Size

If the map resolution of is too high for the system hardware being used, the GDAL Utilities (http://www.gdal.org/gdal_utilities.html) 'gdalwarp' or 'gdal_translate' can be used to reduce the density of points in the map. Decreasing the level of detail is an acceptable tradeoff in many cases.

Location File

The location file must be provided in comma-separated format (e.g., the .CSV files that can be exported from Microsoft Excel). The first line of the file must be a series of headers. Each subsequent line should contain a set of attributes for a single location.

- The first entry of the header must be a **unique location identifier** with the label *Site ID* or *Sample ID*.
- A **vertical coordinate** labelled as *Latitude* (decimal degrees) or *Northing* (Universal Transverse Mercator, UTM) must be included but can be present at any column position after the unique location identifier. Note that positive values = north and negative values = south.
- A **horizontal coordinate**, labelled as *Longitude* (decimal degrees) or *Easting* (Universal Transverse Mercator, UTM) must be included but can be present at any column position after the unique location identifier. Note that Positive values = east and negative values = west.

The **first line** of the file may look something like the following:

```
Site ID,Latitude,Longitude
```

or

```
Site ID,Northing,Easting
```

depending on the coordinate system.

GenGIS provides the ability to specify many different custom column headers within the Location file, including longer descriptive site names, environmental parameters, or a time stamp. For instance, a location file header might look like this:

```
Site ID,Latitude,Longitude,File Size,Environment Type,Geographic Location,Site Name,Country
```

Each of these values **must** then be specified for every entity (= row in the file), even if they are called NULL or some other placeholder value.

Sequence File

The basic specification of the sequence file is even simpler, with only two required fields:

- A **unique location identifier** that is also found in the location file
- A **unique sequence identifier**

The **first line** of the file must begin with the following column headings:

```
Site ID,Sequence ID
```

Columns containing custom information can be added after the two mandatory headings. Each row of the sequence file must define a value for each of the columns identified in the header line. Note that the 'sequence file' need not contain any molecular sequence data, nor do the entities necessarily need to have a one-to-one correspondence to actual sampled sequences.

A simple sequence file might summarize the taxonomic classification of each sampled sequence:

```
Site ID,Sequence ID,Best_match,Species,Genus,Family,Order,Class,Phylum,Superkingdom
```

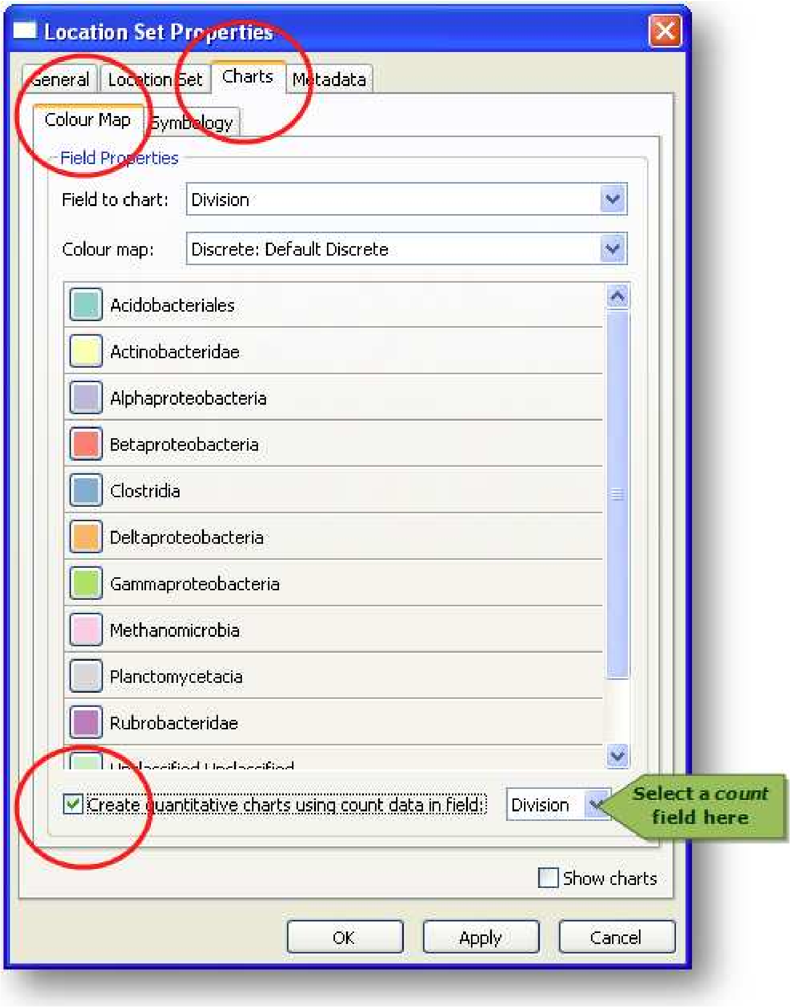
Multiple identical sequences at a location

A *count* field may be specified within a Sequence File to indicate the number of times a given sequence is present at a given site. This has the benefit of significantly reducing the size of a Sequence File.

```
 Site Id, Sequence Id, Count, Domain, Phylum, Class, Order, Family, Genus, Species
```

The *count* field can be assigned any name within the Sequence File. Multiple *count* fields may be specified indicating different quantitative aspects of a named sequence type such as number of sequences or total base pairs observed. The *count* field can then be specified in various locations within GenGIS. For example, quantitative pie charts reflecting the number of times each sequence type is observed at a location can be created through the Location Set Properties dialog:

1. Open the Location Set Properties dialog
2. Navigate to the **Charts > Colour Map** tab
3. Select "Create quantitative charts using count data in field"
4. Select the desired count field from the adjacent drop down menu



Field to chart: Division

Colour map: Discrete: Default Discrete

- Acidobacteriales
- Actinobacteridae
- Alphaproteobacteria
- Betaproteobacteria
- Clostridia
- Deltaproteobacteria
- Gammaproteobacteria
- Methanomicrobia
- Planctomycetacia
- Rubrobacteridae
- Unclassified Unclassified

☑ Create quantitative charts using count data in field: Division

□ Show charts

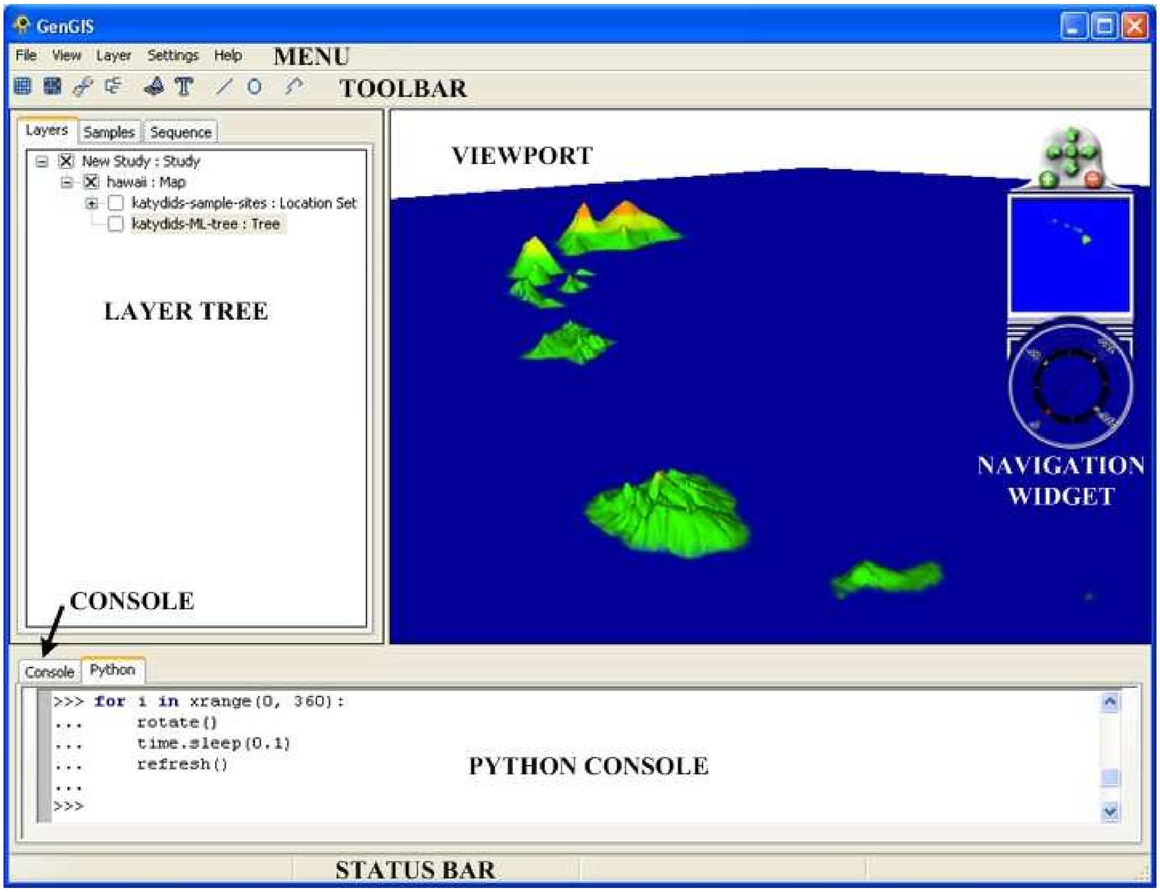
Trees

Input trees should adhere to the Newick file format (http://evolution.genetics.washington.edu/phylip/newicktree.html) , with the additional constraint that leaf labels must match up exactly with either a **Site ID** from the location file or a **Sequence ID** from the sequence file.

The Environment

Overview

- The graphical user interface for GenGIS consists of a collection of different interface elements. Many features of GenGIS can be accessed through the **Menu**.
- Data loaded into GenGIS is organized into a **Layer Tree**, which is made explicit to the user in the panel on the left. This hierarchical structure provides a natural organization of data and allows the properties of related data items to be set easily.
- The most commonly used features are exposed on the **Toolbar**.
- Data visualizations are displayed in the 3D **Viewport**. Mouse navigation within this 3D environment follows a standard world-in-hand navigation model.
- Alternatively, the camera position and angle can be modified using the **Navigation Widget**, which also provides an overview map.



```
>>> for i in xrange(0, 360):
...     rotate()
...     time.sleep(0.1)
...     refresh()
...
>>>
```

- The **Console** provides feedback to the user such as the results of statistical tests and warnings about potential problems with loaded data.
- Almost all features available through the graphical user interface can be accessed within the **Python Console** window. This includes loading data layers, modifying camera parameters, and accessing location or sequence data.
- Information about interface elements or graphical features within the **Viewport** is displayed on the **Statusbar**.

Menu Items

File Menu

- **Save image as:** Save *Viewport* contents to a PNG image file.
- **Exit:** Exits the program.

View Menu

- **Panes → Side Panel:** Hide/unhide the left side panel containing the *Layer tree*.
- **Panes → Console:** Hide/unhide the bottom panel containing the *Console* and *Python Console*.

- **Widgets → Navigation Control:**

Layer Menu

- **Add map:** Add a map to the currently selected study.
- **Add location set:** Add a location set to the currently selected map.
- **Add sequence data:** Add sequence data to the currently selected location set.
- **Add tree:** Add a geographic tree model to the currently selected map.

Note: *Adding multiple location sets, sequence sets or trees is experimental at this time.*

- **Remove layer:** Remove the currently selected layer.
- **Hide all layers:** Hide all layers so they are no longer displayed in the *Viewport*.
- **Show all layers:** Show all layers in the *Viewport*.

Show/Hide the zoom and navigation buttons on the Navigation widget.

- **Widgets → Overview Map:** Show/Hide mini-map display on the navigation widget.
- **Widgets → Compass:** Show/Hide the compass on the navigation widget.

- **Widgets → Show all:** Show all controls of the navigation widget.
- **Widgets → Hide all:** Show/Hide the compass on the navigation widget.

- **Camera Position → Default:** Move camera to its default perspective position.
- **Camera Position → Top:** Move camera to give a top-down view of the map.

- **Detail → Fine:** Increase the level-of-detail of the map.
- **Detail → Coarse:** Decrease the level-of-detail of the map.

Toolbar Buttons

The main toolbar provides easy access to frequently used features.

▦	**Add map**	Add a map to the currently selected study.
▦	**Add location set**	Add a location set to the currently selected map.
	Add sequence data	Add sequence data to the currently selected location set.
	Add tree	Add a geographic tree model to the currently selected map.
	Default perspective view	Move camera to its default perspective position.
T	**Top view**	Move camera to give a top-down view of the map.
/	**Draw layout line**	Draw a straight line which can be used to layout graphic elements such as a 2D tree or pie charts.
○	**Draw layout ellipse**	Draw an ellipse which can be used to layout graphic elements such as a 2D tree or pie charts.
	Draw geographic axis	Draw a polyline which can be used to test the goodness-of-fit between a tree and a non-linear geographic axes (see 2D Pylogenetic Trees).

Note: *Adding multiple location sets, sequence sets or trees is experimental at this time. GenGIS does not support loading multiple maps.*

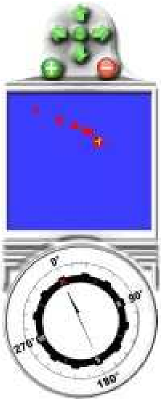
Navigating the Viewport

GenGIS offers three different options to navigate maps: direct mouse gestures, a navigation widget, and two predefined views.

Mouse

- The map can be **moved** by dragging with the left mouse button.
- Camera **pitch** is controlled by dragging with the right mouse button in vertical directions.
- Map **rotation** is performed by dragging with the left mouse button in horizontal directions.
- Camera **zoom level** is controlled using the mouse scroll wheel.

Navigation Widget

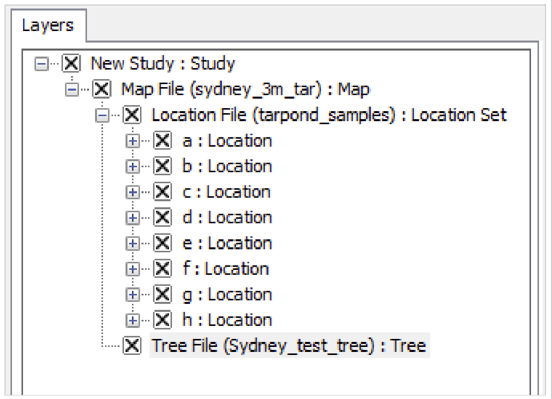


- The navigation widget is located within the top right region of the viewport and provides additional controls to navigate the map.
- The arrows at the top of the widget move the map transversely while the 'plus' and 'minus' buttons control camera zoom.
- Clicking within the overview map (mini map) repositions the main map such that the area clicked is centred on the viewport.
- Clicking or dragging the compass face rotates the map.

Predefined Views

- GenGIS also provides predefined *Perspective* and *Top* views.
- To quickly switch to a perspective view or a top down view use either the corresponding toolbar buttons (left) or the menu options located in the **View → Camera Position**.

Layer Tree Controller



Layers

- ☒ New Study : Study
  - ☒ Map File (sydney_3m_tar) : Map
    - ☒ Location File (tarpond_samples) : Location Set
      - ☒ a : Location
      - ☒ b : Location
      - ☒ c : Location
      - ☒ d : Location
      - ☒ e : Location
      - ☒ f : Location
      - ☒ g : Location
      - ☒ h : Location
    - ☒ Tree File (Sydney_test_tree) : Tree

The Layer Tree Controller (Windows)

- The Layer Tree Controller is located in a left panel within GenGIS and provides a hierarchical view of the Study, Map, Location Set, Location, Sequence and Tree layers.

- Double clicking on the title of any layer within the controller (or right-clicking the same region and selecting *Properties* from a pop-up menu) opens a corresponding properties dialog for that layer.

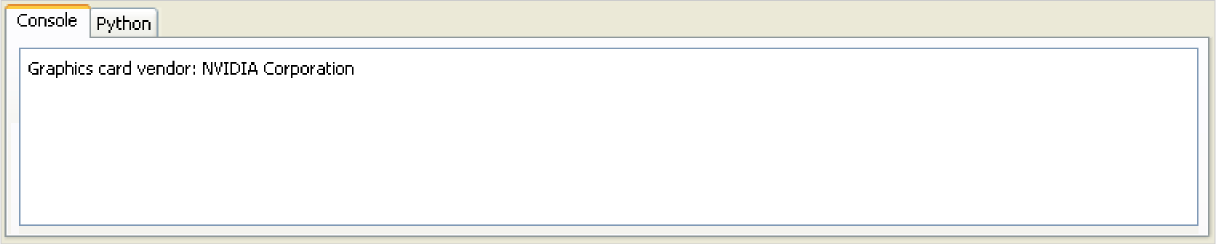
- Layers can be enabled (shown) and disabled (hidden) by clicking on corresponding 'x' control boxes within the controller.

Console Panels

GenGIS features an output console and a Python console both located at the bottom of the main interface.

Output Console

The Output Console displays a log of successful program operations (e.g., loading data files such as map files) as well as possible errors.

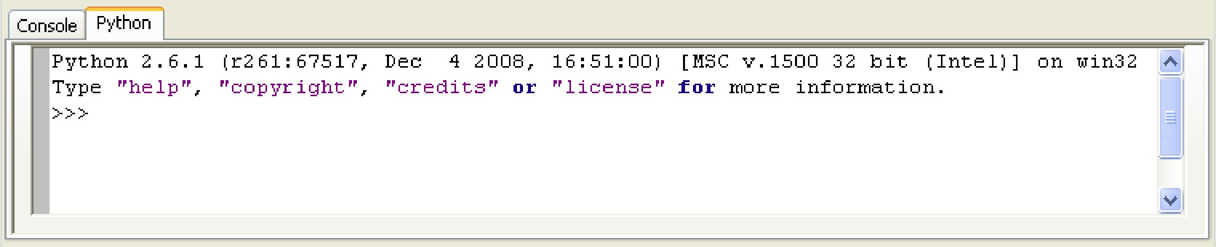


```
Graphics card vendor: NVIDIA Corporation
```

The GenGIS console.

Python Console

The Python console contains a fully-functional Python interpreter that provides access to data structures within GenGIS (e.g., location layer data) and API functions (e.g., the ability to automate camera and lighting controls via Python scripting). The Python console is described in greater detail later in the manual.



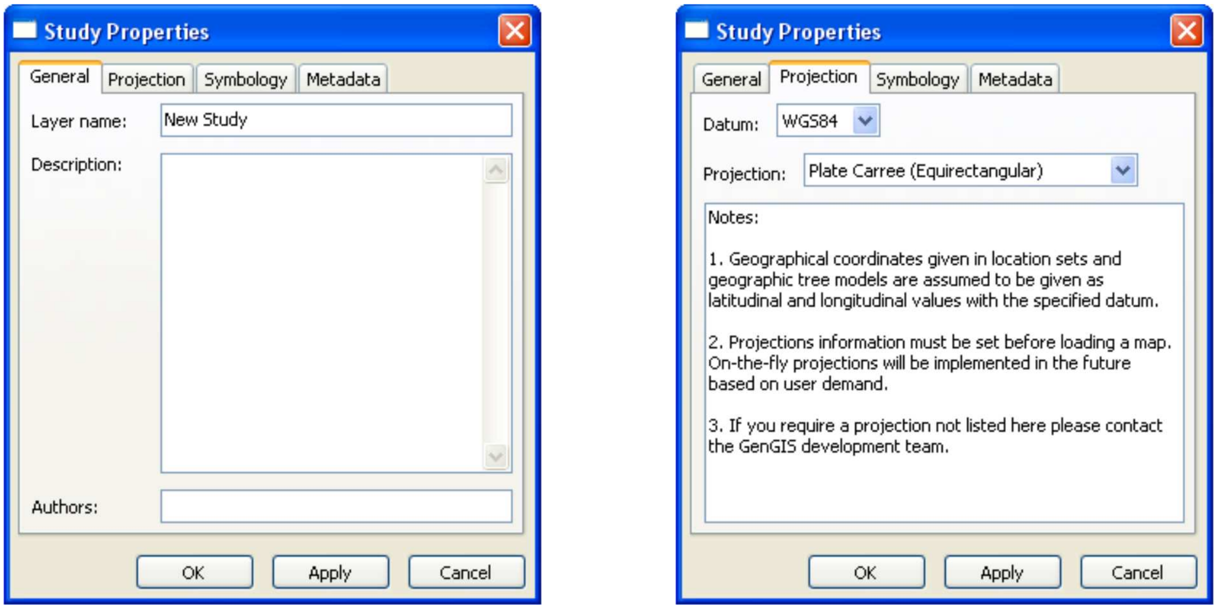
```
Python 2.6.1 (r261:67517, Dec  4 2008, 16:51:00) [MSC v.1500 32 bit (Intel)] on win32
Type "help", "copyright", "credits" or "license" for more information.
>>>
```

The GenGIS Python console.

Layer Property Dialogs

Study Layer Properties

The Study Layer is automatically created during a new session. The Study Properties dialog provides controls to change settings such as study layer metadata (name, description and authors), background colour and terrain resolution. Components of the Study Layer properties dialog are explained in greater detail below.



General Tab

Layer name: New Study

Description:

Authors:

Projection Tab

Datum: WGS84

Projection: Plate Carree (Equirectangular)

Notes:

1. Geographical coordinates given in location sets and geographic tree models are assumed to be given as latitudinal and longitudinal values with the specified datum.

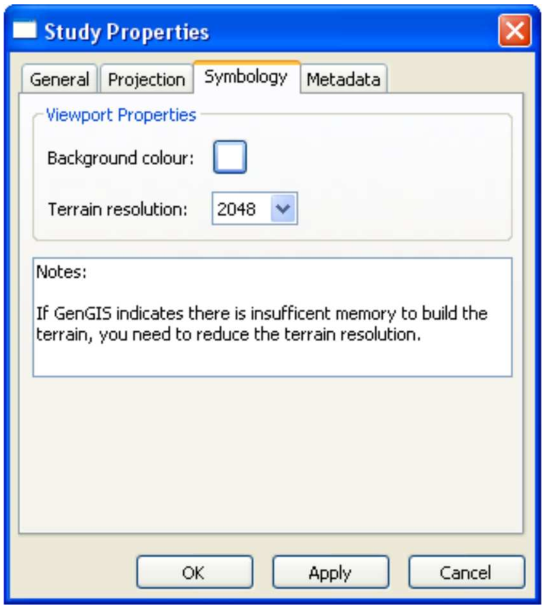
2. Projections information must be set before loading a map. On-the-fly projections will be implemented in the future based on user demand.

3. If you require a projection not listed here please contact the GenGIS development team.

- The General tab contains three editable layer metadata fields: *Layer name*, *Description*, and *Authors*.
- Editing the study *Layer name* property also updates its corresponding representation within the Layer Tree Controller.

- The *Datum* and *Projection* fields can be set from within the Projection tab.
  Note: The projection must be set before loading a map.

Symbology Tab



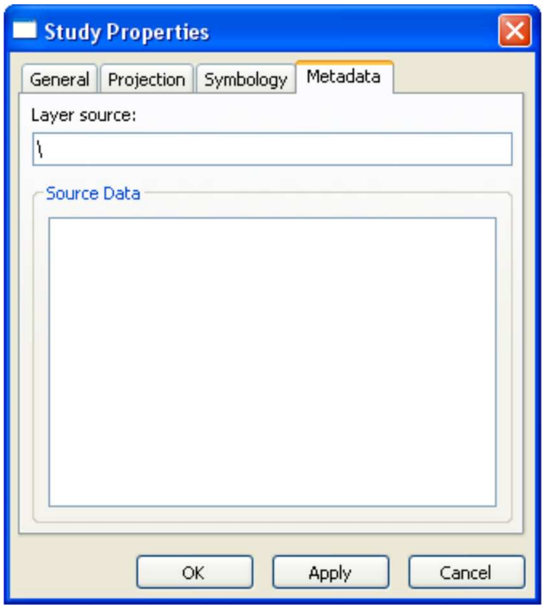
Background colour: □

Terrain resolution: 2048

Notes:

If GenGIS indicates there is insufficent memory to build the terrain, you need to reduce the terrain resolution.

Metadata Tab



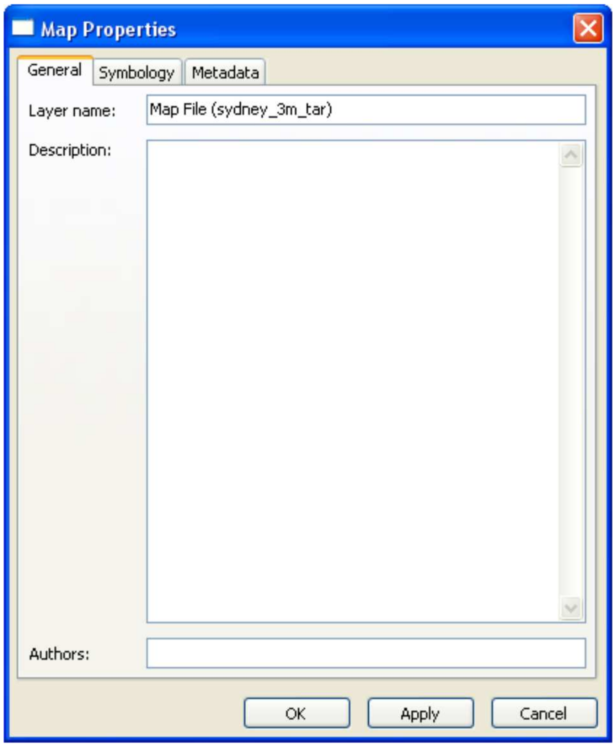
Layer source: \

- The Viewport background colour and the terrain resolution can be set from within the Symbology tab.

Map Layer Properties

The Map Layer displays a map file and directly interacts with other layers (e.g., location, sequence, tree). The Map Layer properties dialog controls settings such as map layer metadata (e.g., name, description), colour scheme and rendering detail. Metadata from the map file (e.g., dimensions, origin) is also displayed. At this time, GenGIS supports a single map layer per session. Components of the Map Layer properties dialog are explained in greater detail below.

General Tab



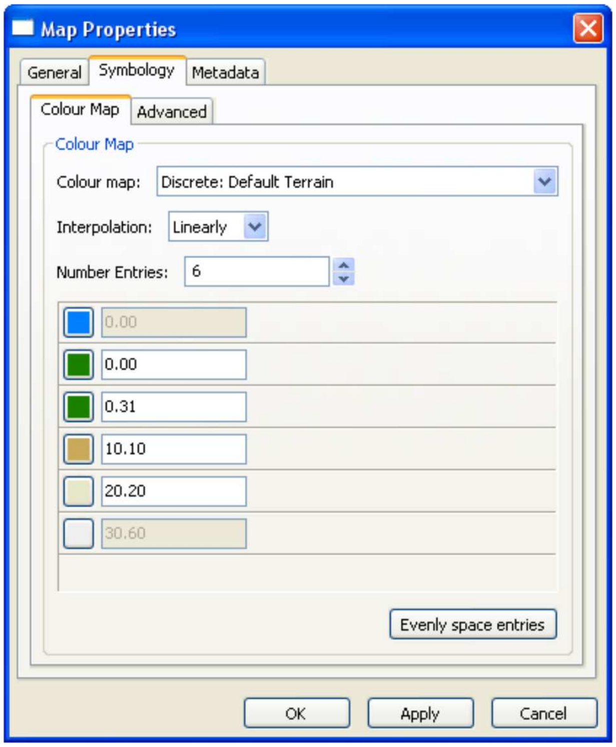
Layer name: Map File (sydney_3m_tar)

Description:

Authors:

The General tab contains three editable layer metadata fields: *Layer name*, *Description*, and *Authors*. Changing the layer name also changes the name of the layer within the Layer Tree Controller.

Symbology > **Colour Map** Tab



Colour map: Discrete: Default Terrain

Interpolation: Linearly

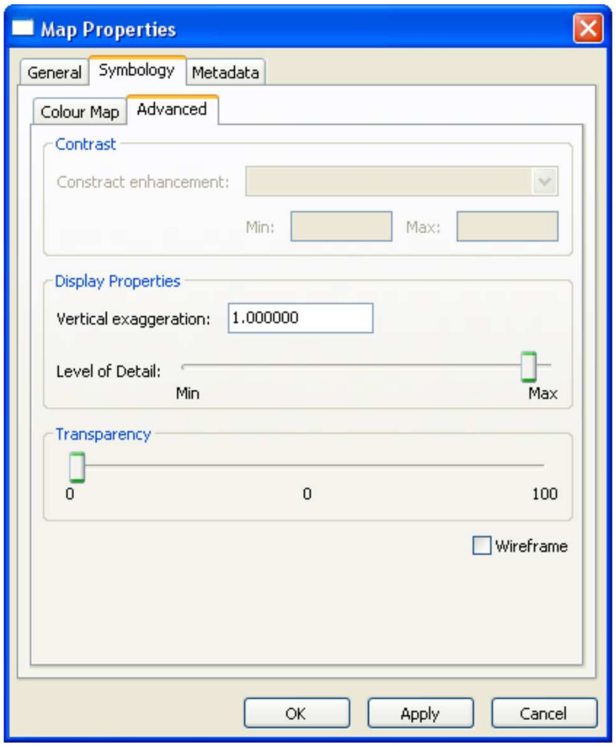
Number Entries: 6

🟦	0.00
🟩	0.00
🟩	0.31
🟫	10.10
□	20.20
□	30.60

The Colour Map tab provides controls to change the map colour scheme, where distinct colours represent differences in terrain elevation.

- The *Colour map* drop down menu provides pre-selected colour themes.
- The *Interpolation* drop down menu controls whether colours are "blended" (linear mode) or applied in distinct bands (discrete mode).
- The *Number Entries* controller specifies the number of distinct colour layers.
- Individual colours and their corresponding elevation can be adjusted in the nested window containing colour buttons.
- Colour-to-elevation representation can be evenly distributed by clicking the *Evenly space entries* button.

Symbology > **Advanced** Tab



Constract enhancement:

Min: Max:

Display Properties

Vertical exaggeration: 1.000000

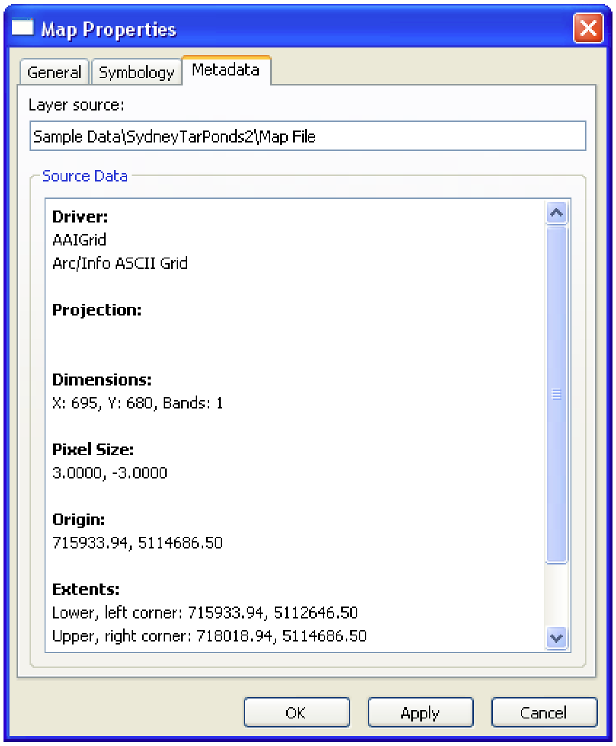
Level of Detail: Min Max

Transparency

0 0 100

□ Wireframe

Metadata Tab



Layer source:

Sample Data\SydneyTarPonds2\Map File

Source Data

Driver:
AAIGrid
Arc/Info ASCII Grid

Projection:

Dimensions:
X: 695, Y: 680, Bands: 1

Pixel Size:
3.0000, -3.0000

Origin:
715933.94, 5114686.50

Extents:
Lower, left corner: 715933.94, 5112646.50
Upper, right corner: 718018.94, 5114686.50

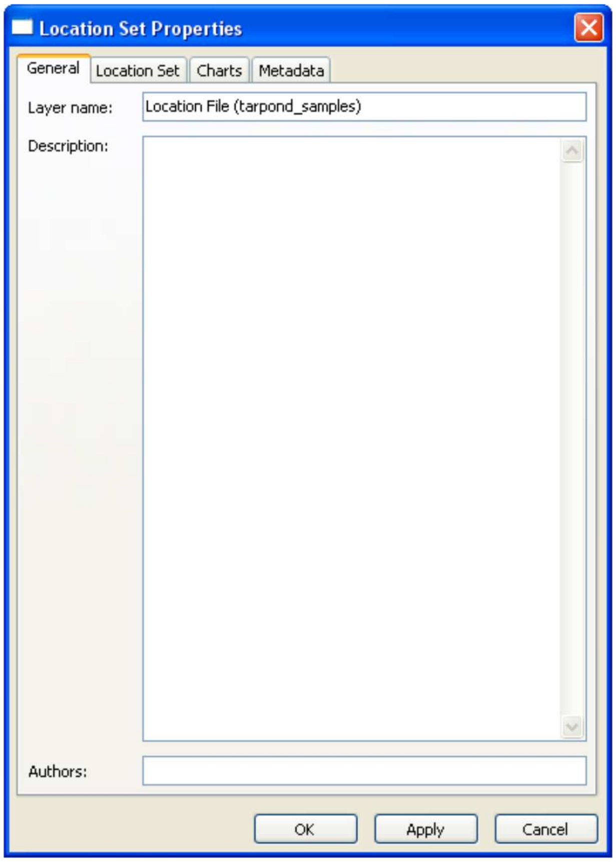
- Display properties such as vertical exaggeration, level of map detail and transparency can be adjusted within the Advanced tab.
- A wireframe mode checkbox displays the map as a transparent wireframe.

- Metadata from the map file (e.g., dimensions, origin) are visible within the Metadata tab.

Location Set Layer Properties

A location file is treated as a *location set* layer containing several distinct *location* layers. The Location Set Layer properties dialog controls settings such as location set layer metadata (e.g., name, description) and visual properties for both locations and charts (e.g., colour, size). Metadata from the location file (e.g., number of sites) is also displayed. At this time, GenGIS supports a single location file per session. Components of the Location Set Layer properties dialog are explained in greater detail below.

General Tab



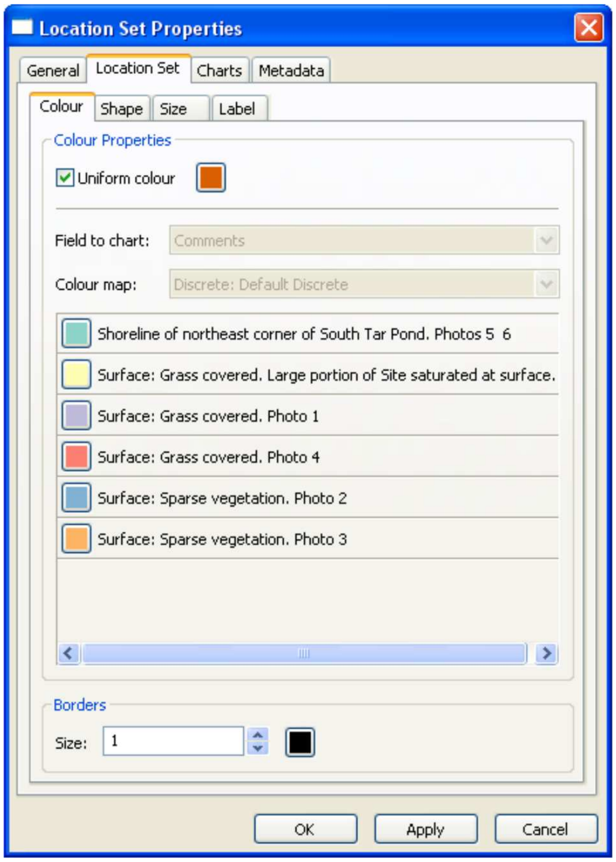
Layer name: Location File (tarpond_samples)

Description:

Authors:

The General tab contains three editable layer metadata fields: *Layer name*, *Description*, and *Authors*. Changing the layer name also changes the name of the layer within the Layer Tree Controller.

Location Set > **Colour** Tab



☑ Uniform colour

Field to chart: Comments

Colour map: Discrete: Default Discrete

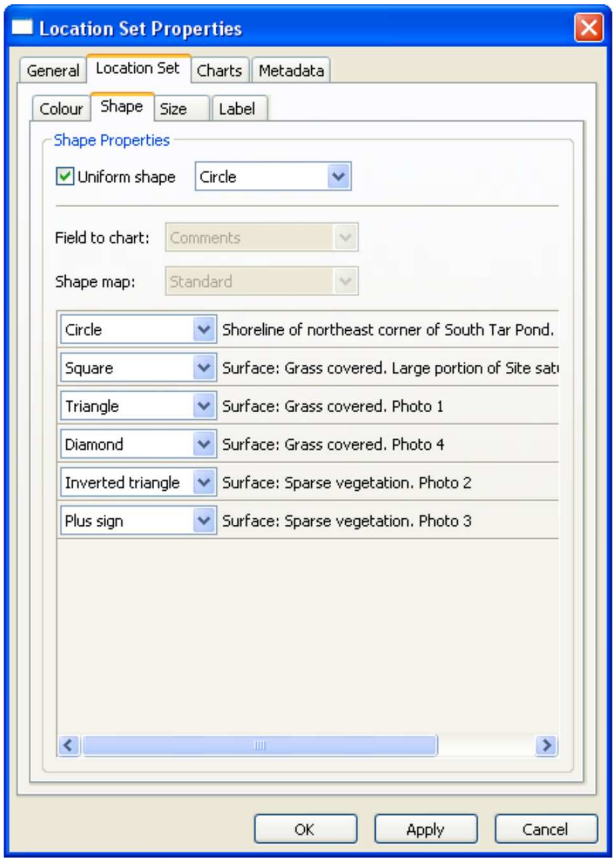
- Shoreline of northeast corner of South Tar Pond. Photos 5 6
- Surface: Grass covered. Large portion of Site saturated at surface.
- Surface: Grass covered. Photo 1
- Surface: Grass covered. Photo 4
- Surface: Sparse vegetation. Photo 2
- Surface: Sparse vegetation. Photo 3

Size: 1

The colour tab provides controls to modify location colour.

- All locations can be displayed in a uniform colour by selecting the *Uniform colour* check-box.
- Alternatively, each location can assigned a unique colour based on a property (e.g., environment type). This is performed by un-selecting the *Uniform colour* check-box and selecting appropriate *Field to char* and *Colour map* fields.
- Individual colours for locations can be adjusted in the nested window containing rows of colour buttons.
- Controls to modify location border colour and width are also provided.

Location Set > **Shape** Tab



☑ Uniform shape: Circle

Field to chart: Comments

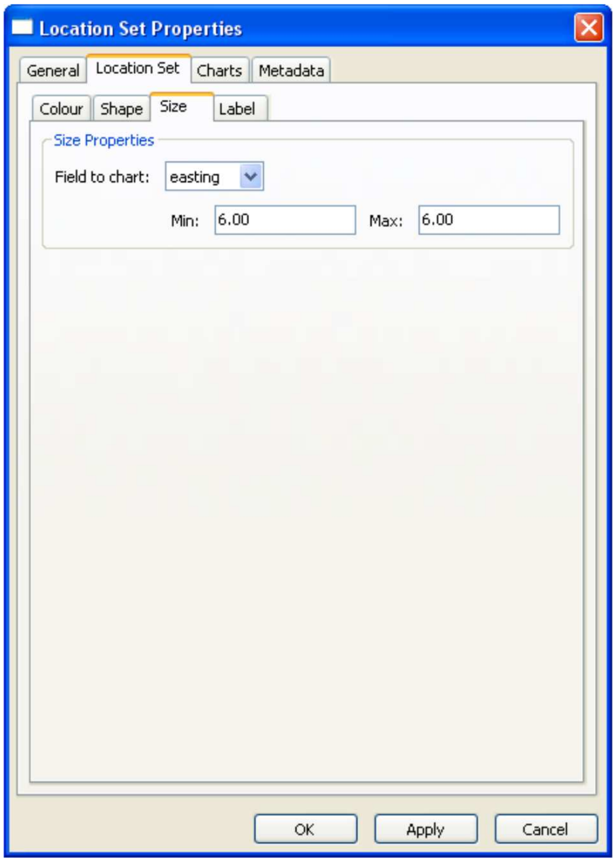
Shape map: Standard

Circle	Shoreline of northeast corner of South Tar Pond.
Square	Surface: Grass covered. Large portion of Site satı
Triangle	Surface: Grass covered. Photo 1
Diamond	Surface: Grass covered. Photo 4
Inverted triangle	Surface: Sparse vegetation. Photo 2
Plus sign	Surface: Sparse vegetation. Photo 3

The shape tab provides controls to modify location shape.

- All locations can be displayed in a uniform shape by selecting the *Uniform shape* check-box.
- Alternatively, each location can assigned a unique shape based on a property (e.g., environment type). This is performed by un-selecting the *Uniform shape* check-box and selecting appropriate *Field to char* and *Shape map* fields.
- Individual shapes for locations can be adjusted in the nested window containing rows of shape selection menus.

Location Set > **Size** Tab



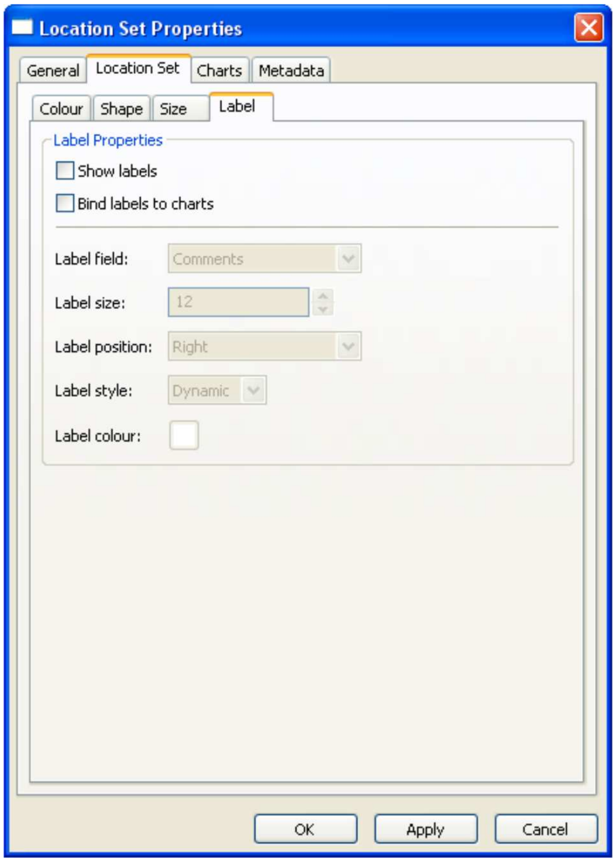
Field to chart: easting

Min: 6.00 Max: 6.00

The size tab provides controls to modify location size.

- Each location can assigned a unique size based on a numeric property (e.g., easting). This is performed by selecting a value from *Field to char* and assigning distinct minimum and maximum values.

Location Set > **Label** Tab



Label Properties

□ Show labels

□ Bind labels to charts

Label field: Comments

Label size: 12

Label position: Right

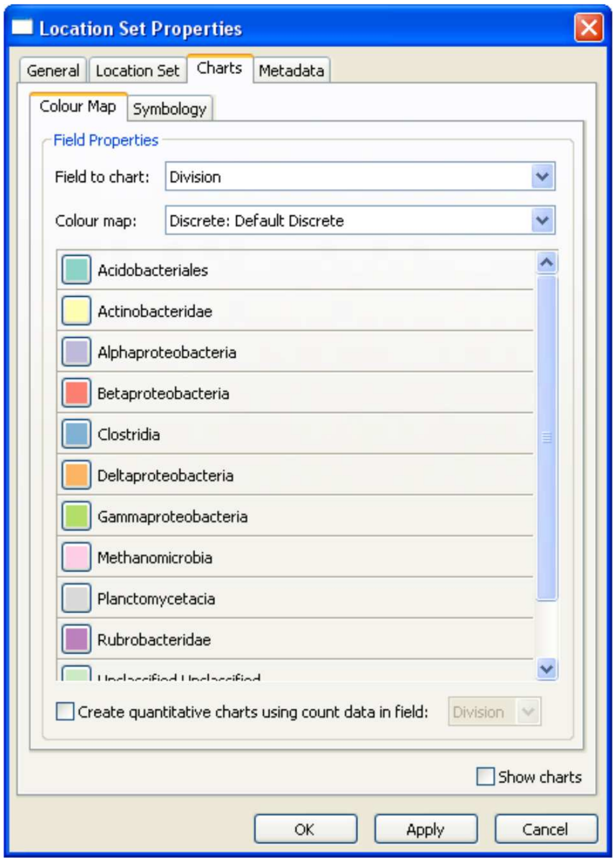
Label style: Dynamic

Label colour: □

Visibility and properties of labels stored in location files can be configured from within the Label tab.

- If location charts are enabled, labels can be bound to the charts by selecting the *Bind labels to charts* check-box.

Charts > **Colour Map** Tab



Field Properties

Field to chart: Division

Colour map: Discrete: Default Discrete

- Acidobacteriales
- Actinobacteridae
- Alphaproteobacteria
- Betaproteobacteria
- Clostridia
- Deltaproteobacteria
- Gammaproteobacteria
- Methanomicrobia
- Planctomycetacia
- Rubrobacteridae
- Unclassified Unclassified

□ Create quantitative charts using count data in field: Division

□ Show charts

- GenGIS supports **bar charts** and **pie charts** to represent sequence data within locations.
- Chart colour themes can be controlled from within the Colour Map tab.
- Different sequence metadata fields can be selected for graphing using the *Field to chart* menu.
- Charts can be enabled from either the Colour Map or Symbology tabs by selecting the *Show charts* check-box.

Note: Sequence data must be present in order to use the Colour Map tab controls.

Charts > **Symbology** (1) Tab (Chart type: *Bar Chart*)

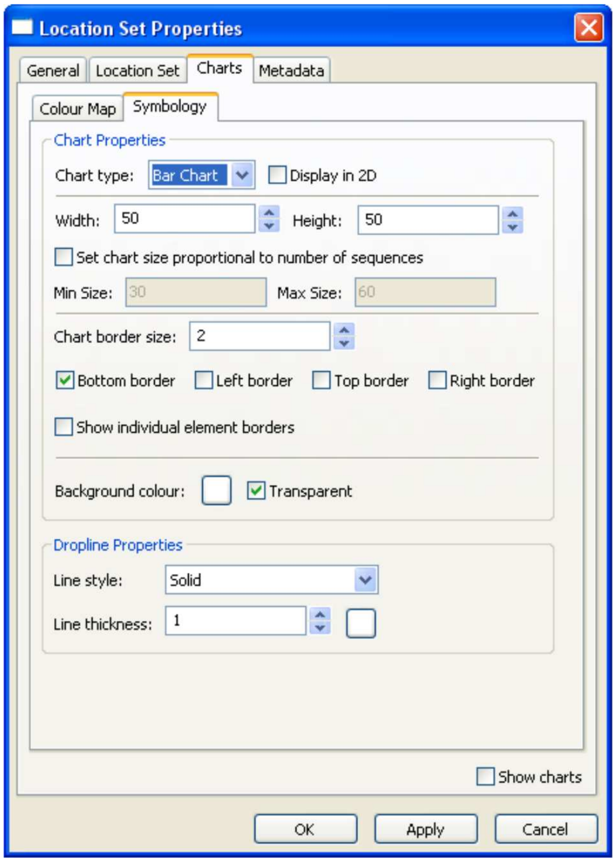


Chart type: Bar Chart ☐ Display in 2D

Width: 50 Height: 50

☐ Set chart size proportional to number of sequences

Min Size: 30 Max Size: 60

Chart border size: 2

☑ Bottom border ☐ Left border ☐ Top border ☐ Right border

☐ Show individual element borders

Background colour: □ ☑ Transparent

Line style: Solid

Line thickness: 1 □

☐ Show charts

- Chart size, position and background colour are all accessible through controls within the Symbology tab.
- Droplines visually connect locations with their corresponding charts. Dropline style, thickness and colour controls are located within the Symbology tab.
- Charts can be enabled from either the Colour Map or Symbology tabs by selecting the *Show charts* check-box.

Note: Sequence data must be present in order to use the Symbology tab controls.

Charts > **Symbology** (2) Tab (Chart type: *Pie Chart*)

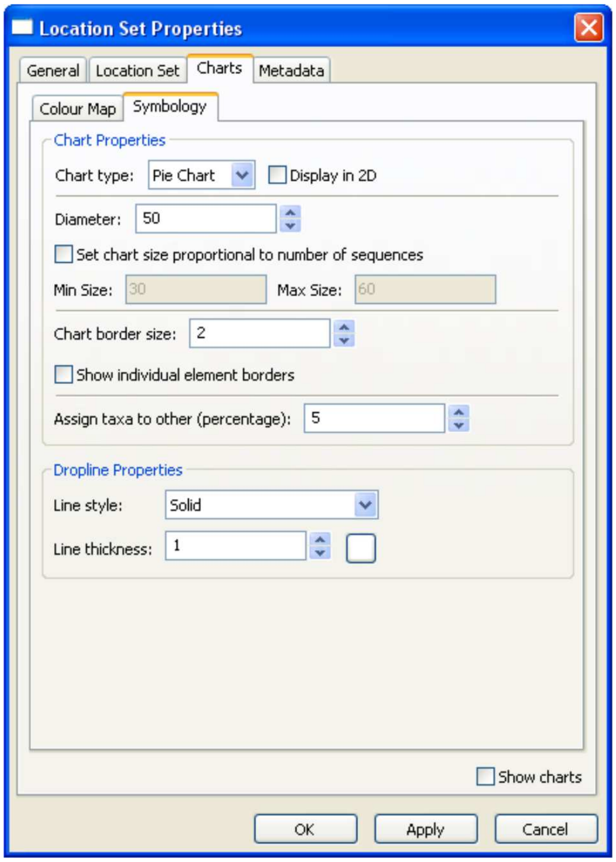


Chart type: Pie Chart ☐ Display in 2D

Diameter: 50

☐ Set chart size proportional to number of sequences

Min Size: 30 Max Size: 60

Chart border size: 2

☐ Show individual element borders

Assign taxa to other (percentage): 5

Line style: Solid

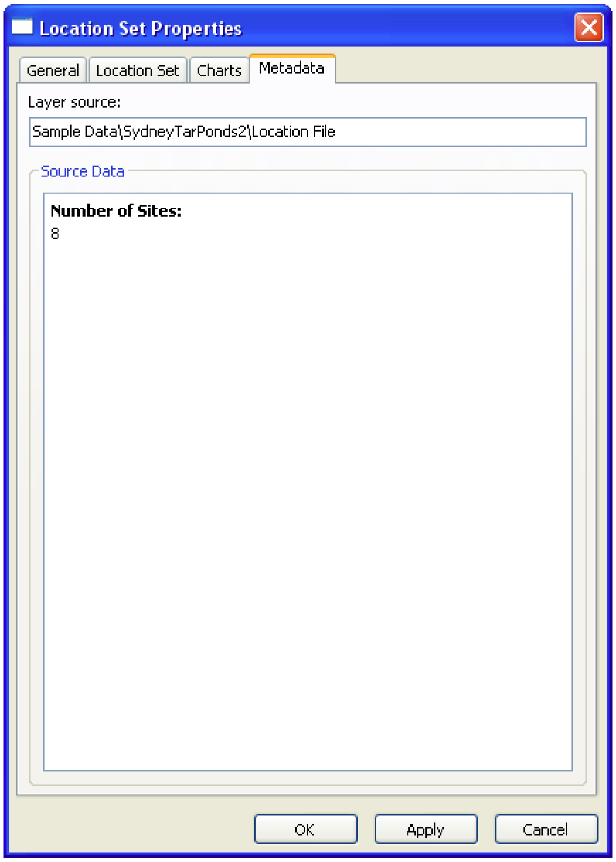
Line thickness: 1 □

☐ Show charts

- Different controls are available depending on the type of chart selected.

Note: Sequence data must be present in order to use the Symbology tab controls.

Metadata Tab



Layer source:
Sample Data\SydneyTarPonds2\Location File

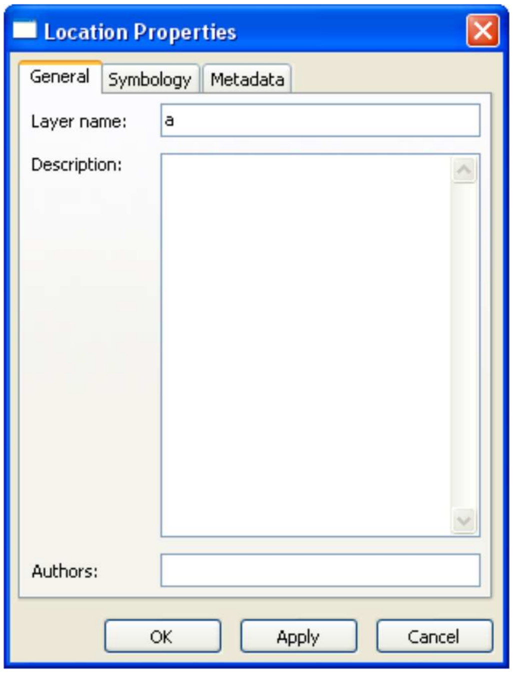
Number of Sites:
8

- Metadata from the location file (e.g., number of sites) is visible within the Metadata tab.

Location Layer Properties

Properties of individual locations are accessible through the Location Layer Properties dialog. Components of the properties dialog are explained in greater detail below.

General Tab



Location Properties

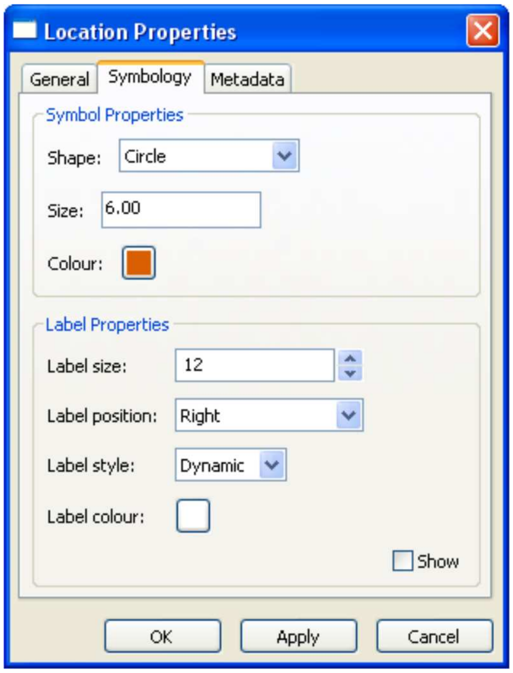
Layer name: a

Description:

Authors:

The General tab contains three editable layer metadata fields: *Layer name*, *Description*, and *Authors*. Changing the layer name also changes the name of the layer within the Layer Tree Controller.

Symbology Tab



Location Properties

Symbol Properties

Shape: Circle

Size: 6.00

Colour: 🟧

Label Properties

Label size: 12

Label position: Right

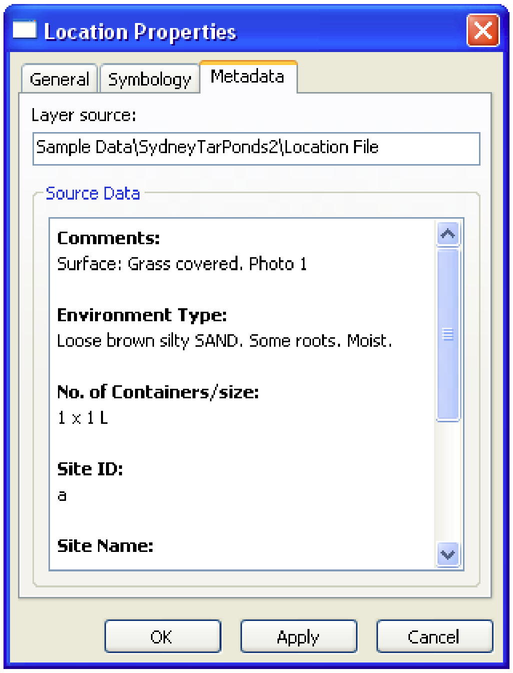
Label style: Dynamic

Label colour: □

□ Show

- Location shape, size and colour can be controlled from within the Symbology tab.
- Controls for label properties are also provided.

Metadata Tab



Location Properties

Layer source:

Sample Data\SydneyTarPonds2\Location File

Source Data

Comments:
Surface: Grass covered. Photo 1

Environment Type:
Loose brown silty SAND. Some roots. Moist.

No. of Containers/size:
1 x 1 L

Site ID:
a

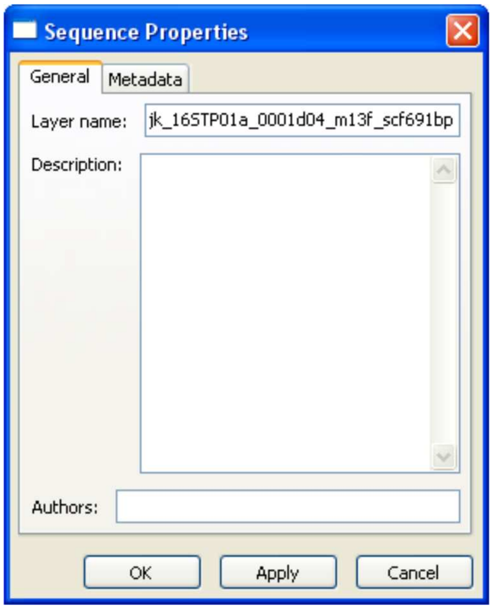
Site Name:

- Metadata from the location file (e.g., comments) is visible within the Metadata tab.

Sequence Layer Properties

The Sequence Properties dialog provides controls to modify sequence layer metadata (e.g., name, description) and also displays sequence file metadata. Components of the Sequence Layer properties dialog are explained in greater detail below.

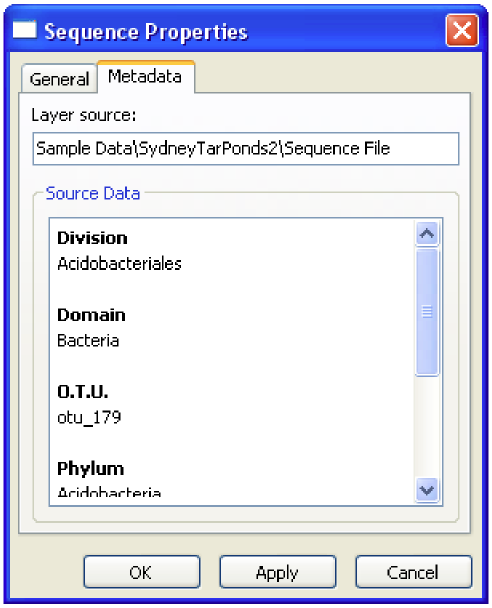
General Tab



Layer name: jk_16STP01a_0001d04_m13f_scf691bp

The General tab contains three editable layer metadata fields: *Layer name*, *Description*, and *Authors*. Changing the layer name also changes the name of the layer within the Layer Tree Controller.

Metadata Tab



Layer source: Sample Data\SydneyTarPonds2\Sequence File

Division
Acidobacteriales

Domain
Bacteria

O.T.U.
otu_179

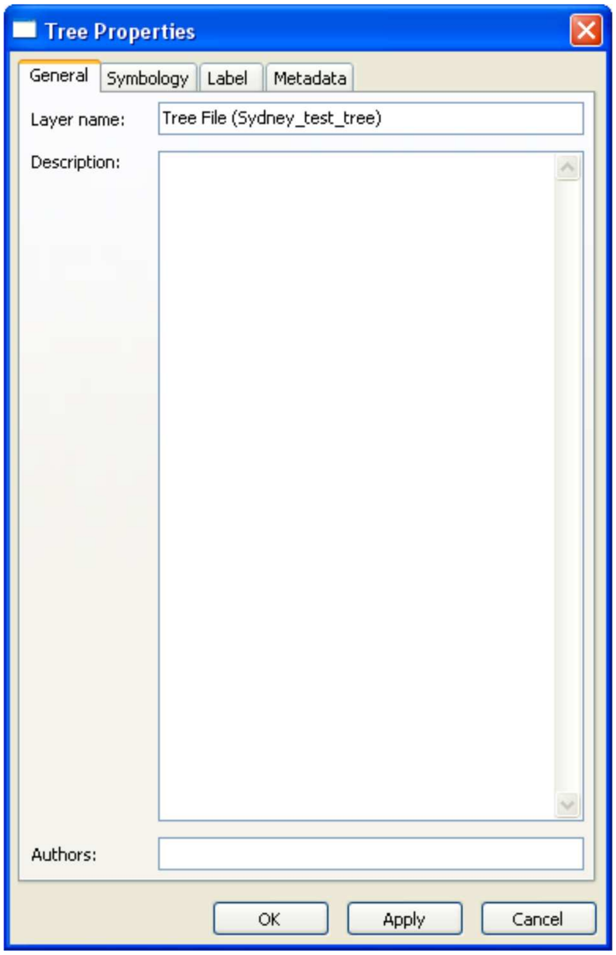
Phylum
Acidobacteria

- Metadata from the sequence file (e.g., division) is visible within the Metadata tab.

Tree Layer Properties

GenGIS supports 2D and 3D trees (e.g., phylogenetic or hierarchical cluster tree). Different display properties (e.g., colour, line width) can be assigned to trees through the Tree Layer Properties dialog. GenGIS supports loading multiple trees during a single session. Components of the Tree Layer properties dialog are explained in greater detail below.

General Tab



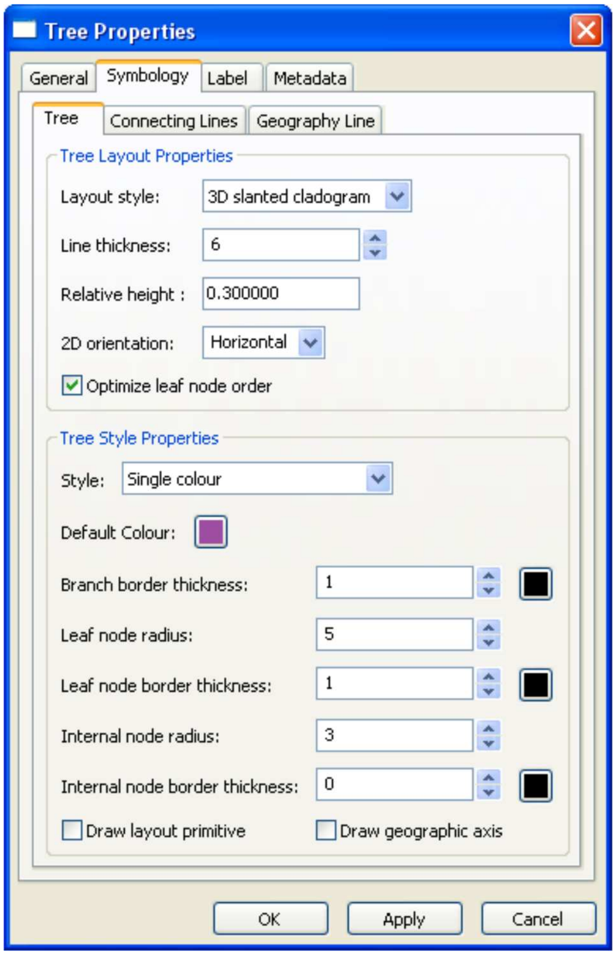
Layer name: Tree File (Sydney_test_tree)

Description:

Authors:

The General tab contains three editable layer metadata fields: *Layer name*, *Description*, and *Authors*. Changing the layer name also changes the name of the layer within the Layer Tree Controller.

Symbology > **Tree** Tab



Tree Layout Properties

Layout style: 3D slanted cladogram

Line thickness: 6

Relative height : 0.300000

2D orientation: Horizontal

☑ Optimize leaf node order

Tree Style Properties

Style: Single colour

Default Colour:

Branch border thickness: 1

Leaf node radius: 5

Leaf node border thickness: 1

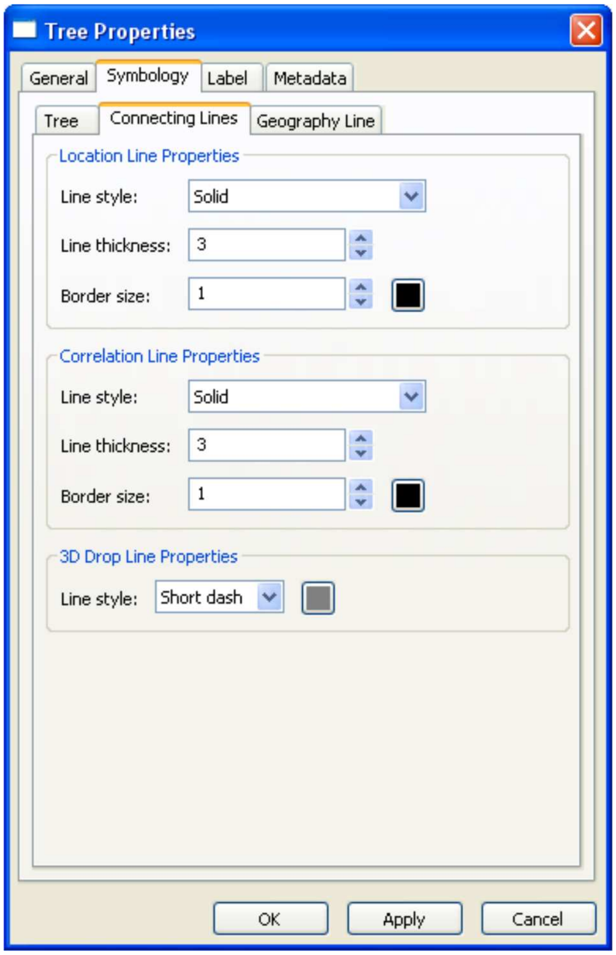
Internal node radius: 3

Internal node border thickness: 0

□ Draw layout primitive □ Draw geographic axis

- Tree layout and style properties can be set from within the Tree tab.

Symbology > **Connecting Lines** Tab



Location Line Properties

Line style: Solid

Line thickness: 3

Border size: 1

Correlation Line Properties

Line style: Solid

Line thickness: 3

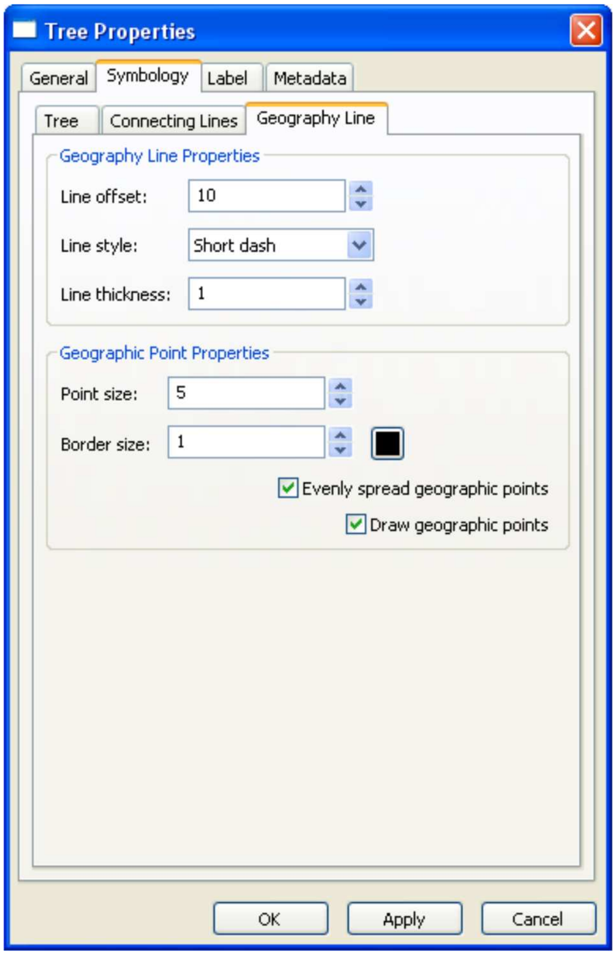
Border size: 1

3D Drop Line Properties

Line style: Short dash

- Location, correlation and 3D drop line properties can be set from within the Connecting Lines tab.

Symbology > **Geography Line** Tab



Geography Line Properties

Line offset: 10

Line style: Short dash

Line thickness: 1

Geographic Point Properties

Point size: 5

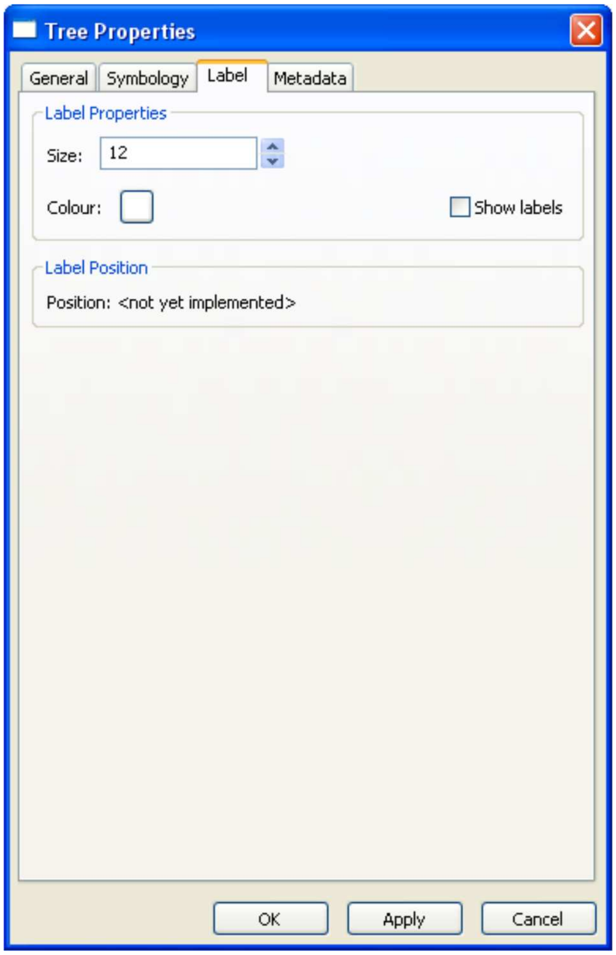
Border size: 1

☑ Evenly spread geographic points

☑ Draw geographic points

- Geography line and geographic point properties can be set from within the Geography Line tab.

Label Tab



Label Properties

Size: 12

Colour:

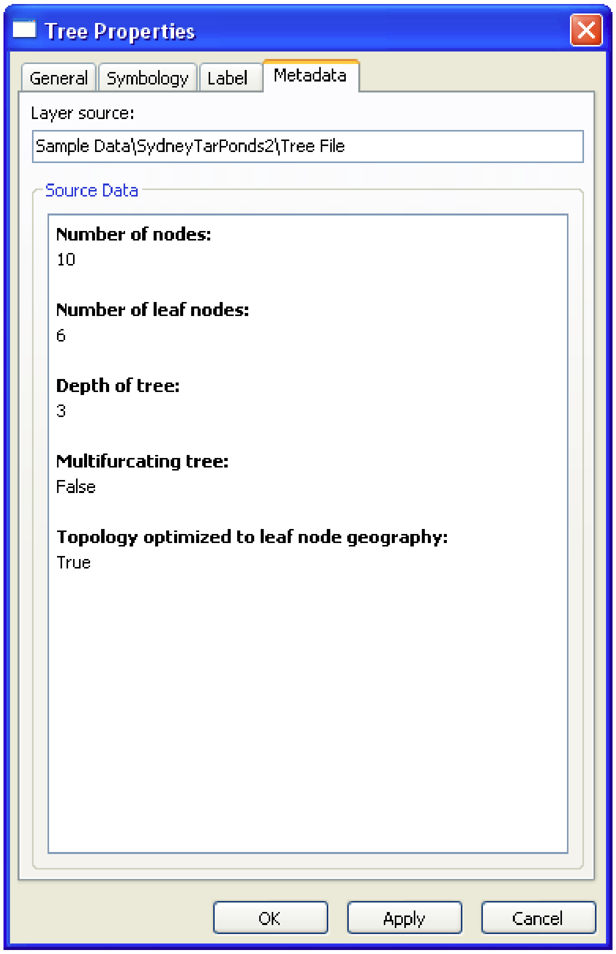
□ Show labels

Label Position

Position: <not yet implemented>

- Tree label size, colour and visibility can be set from within the Label tab.

Metadata Tab



Layer source:

Sample Data\SydneyTarPonds2\Tree File

Source Data

Number of nodes:
10

Number of leaf nodes:
6

Depth of tree:
3

Multifurcating tree:
False

Topology optimized to leaf node geography:
True

- Metadata from the tree file (e.g., number of nodes) is visible within the Metadata tab.

The Python console and API functions

What you can do with the console

The *Python Console* provides access to a standard Python interpreter (http://www.python.org/) . Python is a general-purpose high-level programming language with many packages (http://pypi.python.org/pypi/) available for phylogenetics, population genetics, and statistics. Data loaded into GenGIS is exposed to the *Python Console* allowing quantitative hypothesis testing to be performed directly within GenGIS. Results of analyses can be visualized within the *Viewport* to aid in interpretation of results and generation of new hypotheses.

Below we give several short examples of using this API. You can also find information about using the API on our tutorials page.

Accessing location site and sequence data

Location site and sequence data can be accessed directly from the *Python Console*. You can access all location layers using:

```
locLayers = GenGIS.layerTree.GetLocationSetLayer(0).GetAllLocationLayers()
```

If you wish to get a list of only the active location layers (i.e., those which are checked), use:

```
activeLocLayers = GenGIS.layerTree.GetLocationSetLayer(0).GetAllActiveLocationLayers()
```

Properties of a location are accessed through its controller:

```
locController = locLayers[0].GetController()
```

A list of functions supported by the location controller can be obtained with:

```
dir(locController)
```

The metadata associated with a location is accessed as a python dictionary:

```
metadata = locController.GetData()
metadata.keys()
metadata['Site ID']
```

All sequence layers or all active sequence layers associated with a location layer can be accessed using:

```
seqLayers = locLayers[0].GetAllSequenceLayers()
activeSeqlayers = locLayers[0].GetAllActiveSequenceLayers()
```

Analogous to location data, data associated with a sequence is accessed through the sequence controller:

```
seqController = seqLayers[0].GetController()
metadata = seqController.GetData()
metadata.keys()
metadata['Sequence Id']
```

Filtering data

We have provided a simple function, *filterData*, for filtering data. This function is contained in *dataHelper.py*. As an example, all locations with a temperature greater than 20 can be obtained as follows:

```
import dataHelper
locLayers = GenGIS.layerTree.GetLocationSetLayer(0).GetAllLocationLayers()
filteredLocLayers = dataHelper.filter(locLayers, 'Temperature', 20, dataHelper.filterFunc.greater)
```

The function *filter* takes 4 parameters:

- the data to be filtered
- the field to filter on
- the value to filter on
- a filtering function which returns true for all items passing the filter

Filtering can be done on either strings or numeric values:

```
import dataHelper
seqLayers = locLayers[0].GetAllSequenceLayers()
filteredSeqLayers = dataHelper.filter(seqLayers, 'Phylum', 'Actinobacteria', dataHelper.filterFunc.equal)
```

Basic filtering functions are provided in *filterFunc.py*, but it is easy to write your own filtering functions. For example, the *equal* filter used above is simply:

```
def equal(val1, val2):
  return str(val1) == str(val2)
```

Creating custom data visualizations

Using the VisualLine, VisualMarker, and VisualLabel classes one can create custom data visualizations with GenGIS. The VisualLine class allows user defined lines to be drawn in the Viewport. Suppose we have two locations within our location set with ids of 'GBR' and 'ITA'.

We can draw a line between these locations as follows:

```
# get location layers
locLayers = GenGIS.layerTree.GetLocationSetLayer(0).GetAllLocationLayers()
```

```
# create a dictionary indicating the geographic coordinates of each location
locDict = {}
for loc in locLayers:
  locDict[loc.GetController().GetId()] = [loc.GetController().GetLongitude(), loc.GetController().GetLatitude()]

# get the 3D position of GBR and ITA
terrainController = GenGIS.layerTree.GetMapLayer(0).GetController()
gbrPt = GenGIS.Point3D()
terrainController.GeoToGrid(GenGIS.GeoCoord(locDict['GBR'][0], locDict['GBR'][1]), gbrPt)
```

```
itaPt = GenGIS.Point3D()
terrainController.GeoToGrid(GenGIS.GeoCoord(locDict['ITA'][0], locDict['ITA'][1]), itaPt)
```

```
# draw a solid red line with a width of 2 between these countries
line = GenGIS.VisualLine(GenGIS.Colour(1,0,0), 2, GenGIS.LINE_STYLE.SOLID, GenGIS.Line3D(gbrPt, itaPt))
lineId = GenGIS.graphics.AddLine(line)
GenGIS.viewport.Refresh()
```

The visual properties of this line can easily be changed at any time to reflect different aspects of your data:

```
line.SetColour(GenGIS.Colour(0,1,0))
line.SetThickness(5)
line.SetLineStyle(GenGIS.LINE_STYLE.SHORT_DASH)
GenGIS.viewport.Refresh()
```

We can later remove this line using:

```
GenGIS.graphics.RemoveLine(lineId)
GenGIS.viewport.Refresh()
```

The VisualMarker class allows user defined markers to be drawn in the Viewport. It is similar to the VisualLine class. For example, we can draw a blue circle over London which is situated at a longitude 0.128W and a latitude of 51.51N as follows:

```
# get 3D position of geographic coordinate in Viewport
terrainController = GenGIS.layerTree.GetMapLayer(0).GetController()
pt = GenGIS.Point3D()
terrainController.LatLongToGrid(GenGIS.GeoCoord(-0.128, 51.51), pt)
marker = GenGIS.VisualMarker(GenGIS.Colour(0,0,1), 6, GenGIS.MARKER_SHAPE.CIRCLE, pt)
markerId = GenGIS.graphics.AddMarker(marker)
GenGIS.viewport.Refresh()
```

VisualLabels can be used to create orthographic (e.g., to indicate a legend or figure caption) or perspective text (e.g., to label points on a map). We can create a "Hello World!" label for a map as follows:

```
label = GenGIS.VisualLabel("Hello World!", GenGIS.Colour(0,0,0), 12, GenGIS.LABEL_RENDERING_STYLE.ORTHO)
label.SetScreenPosition(GenGIS.Point3D(20,20,1))
labelId = GenGIS.graphics.AddLabel(label)
GenGIS.viewport.Refresh()
```

Alternatively, we can use a VisualLabel to label our marker at London:

```
terrainController = GenGIS.layerTree.GetMapLayer(0).GetController()
label = GenGIS.VisualLabel("London", GenGIS.Colour(0,0,0), 12, GenGIS.LABEL_RENDERING_STYLE.PERSPECTIVE)
pt = GenGIS.Point3D()
terrainController.LatLongToGrid(GenGIS.GeoCoord(-0.128, 51.51), pt)
label.SetGridPosition(pt)
labelId = GenGIS.graphics.AddLabel(label)
GenGIS.viewport.Refresh()
```

A label can be removed with:

```
GenGIS.graphics.RemoveLabel(labelId)
GenGIS.viewport.Refresh()
```

By combining these graphical primative and encoding key aspects of your data to different visual properties (i.e., colour, size, shape) GenGIS can be used to identify interesting patterns within a wide-range of datasets. An example which uses these classes to visualizing a distance matrix indicating the rate of import and export of HIV-1 subtype B for different European countries as reported by Paraskevis et al. (2009) (http://www.retrovirology.com/content/6/1/49) is available here.

Creating fly-through movies

A collection of functions for creating fly-through movies are available in *movieHelper.py* found in the *scripts* directory. A useful movie is to rotate the map about its origin. Such a movie can be made using the `rotateAboutOrigin` function which takes the number of degrees to rotate and the time of the movie as parameters:

```
import movieHelper
movieHelper.rotateAboutOrigin(360, 10)
```

More general movies can be created by capturing the camera parameters at *key frames* using the function `getCameraParam` and then interpolating between these key frames using the function `linearInterpolateParams`:

```
import movieHelper
# move camera to first key frame (for example, use the toolbar to set a top down view)
keyFrame1 = movieHelper.getCameraParam()
# move camera to next key frame (for example, use the toolbar to set the default perspective view)
keyFrame2 = movieHelper.getCameraParam()
# set camera back to first key frame
movieHelper.setCameraParam(keyFrame1)
# smoothly move between these key frames in 5 seconds
movieHelper.linearInterpolateParams(keyFrame1, keyFrame2, 5)
```

For examples of creating custom movies which do not use the movieHelper API, have a look at the series of H1N1 movies we have developed. In particular, note that GenGIS.mainWindow.Yield() must be called occasionally for time series to run correctly.

By stitching together multiple key frames complex fly-through movies can be created. Commercial software such as Camtasia (http://www.techsmith.com/camtasia.asp) or open source software such as CamStudio (http://sourceforge.net/projects/camstudio/) can be used to record these movies.

Retrieved from "http://kiwi.cs.dal.ca/GenGIS/The_GenGIS_Manual"

- This page was last modified on 16 March 2012, at 20:52.