The IST Programme
Project No. 10378

Sim**Bio**

The SimBio Visualisation Module Release Notes

http://www.simbio.de
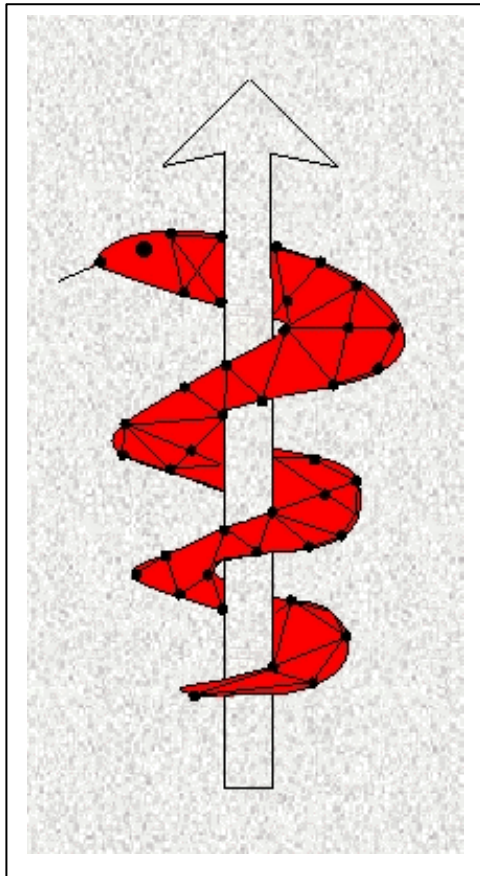
Deliverable D5b
Visualisation Release Notes

Status:	Final
Version:	1.0
Security:	Public
Responsible:	MPI
Authoring Partners:	MPI

Release History

Version	Date
1.0	30.04.01

The SimBio Visualisation Module Release Notes

Markus Svensén
Max-Planck Institute of Cognitive Neuroscience

1. Introduction

This document is the release notes for the first prototype of the visualisation module (VM) of the SimBio project [SimBio00]. The purpose of this module is, as its name suggests, to visualise the results from other SimBio modules, e.g. stress fields from bio-mechanical simulations or source localisation and influence from bio-electrical simulations.

The full functionality intended for implementation in the final version of the VM is described in its requirement specification [Svensen00]. The purpose of this document is to specify the functionality implemented in this initial prototype.

1.1. Overview of this Release

This release is the first prototype of the VM and only implements a part of the full functionality described in [Svensen00]. In summary, the basic and essential functionality is available, offering visualisation of different kinds of data (scalar, complex, vector and tensor valued data) defined on different topologies (images and geometrical objects), but more advanced visualisation techniques (overlays, animations) have not yet been implemented.

The purpose of this initial prototype is twofold. First, it will provide the other partners in the SimBio consortium with means for visualising initial simulation results. Second, it serves to demonstrate the progress made so far within this workpackage.

1.2. The Rest of this Document

The following chapter provides a detailed listing of the requirements met in this release. It also describes the dependencies of the VM on other software packages. Finally, it gives guidelines for how the VM is to be installed.

The last chapter outlines the plans for future releases.

2. Release Notes

This chapter provides the detailed release notes for the initial prototype of the SimBio visualisation module.

2.1. Requirements met in this Release

2.1.1. Visualisation of Voxel Volumes

The VM is able to display 2D slices of data (images) extracted from 3D, cubic voxel volumes along any of the three principal axes (centred normals of the sides of the cubic volume).

Scalar data are displayed as pixel images using colour maps selectable for each view.

The VM is capable of scaling scalar data to employ the full range of the available colour map.

RGB data are displayed as pixel images with the corresponding colours.

Scalar representations of multivariate data, such as magnitude of complex numbers and 3-element vectors and volume of 6-element tensors, are displayed just like scalar values.

Complex numbers are visualised by representing the real and imaginary components by different orthogonal colours (red and blue). This may be combined with intensity

indicating the magnitude.

It is possible to display 3-element vectors projected on the image plane.

It is possible to display 6-element tensors by projecting the corresponding ellipsoidal representations (see 6-element tensors in geometrical models) on the image plane.

The default visualisation for whole images of scalar values is a scaled grey scale map.

The default visualisation for images of 3-element vector values is the corresponding 2D projection in the image plane.

The default visualisation for images of 6-element vector values is the 2D projection of the corresponding ellipsoidal representation in the image plane.

2.1.2. Visualisation of Geometrical Objects

The VM is able to render 3D geometric objects (meshes, surfaces and volumes), with variable rotation and scale.

Scalar and RGB values associated with vertices are displayed through the colour of vertices and, where applicable, edge and surface elements. The colour of the latter will be interpolated between the colours of the vertices defining the surface element, in the scalar case using colour maps selectable for each view.

The VM is capable of scaling scalar data to employ the full range of the chosen colour map.

Scalar representations of multivariate data, such as magnitude of complex number and 3-element vectors and volume of 6-element tensors, are displayed just like scalar values.

Complex numbers associated with vertices will be visualised by representing the real and imaginary components by different orthogonal colours (red and blue), specifying the colour of the model surface according to the scheme governing scalar values.

3-element vectors will be displayed directly at the vertex position in the 3D space.

6-element tensors will be displayed directly at the corresponding position (see prev. para.) in the 3D space, using a 3D ellipsiodal representations, so called smarties.

2.2. Software Dependencies

The VM depends on several software packages, without which it will not be functional. All these software packages are either available free on the Internet or available within the SimBio project.

2.2.1. The SimBio-Vista Library

The SimBio-Vista library, which is a part of the software delivered from SimBio workpackage 1, subtask 1, provides means for creating, accessing, manipulating and storing data structures representing SimBio data sets (images and geometrical objects).

The SimBio-Vista library is built on the Vista library [Vista], with data structures for representing geometrical objects added. More information about the library and its content can be obtained from [Kruggel00] as well as the man-pages delivered with the library.

The library is available to project partners via download from the SimBio web-site.

2.2.2. OpenGL

The VM uses the OpenGL® library for the visualisation. This is library that provides low-level routines for creating 2D and 3D computer graphics. It was initially developed by SGI and today its architecture is under the supervision of a dedicated Architecture Review Board, formed by several computer hard- and software companies active in the field of computer graphics.

There exist several different implementations of the OpenGL library, and the VM should work with any of these. However, to be practically useful with the data sets that can be anticipated in the SimBio project, it should be combined with an OpenGL implementation that exploits dedicated graphics hardware. This will ensure that visualisation of data will be obtained rapidly. Such hardware is today available at a moderate cost, typically in the form of graphic cards fitting standard PC computers. Although not all such cards come with Linux® drivers, the number of cards that do so are steadily increasing. Also many other computer arhitectures that are available with a

UNIX® operating system (e.g. Irix®) includes dedicated, hardware supported OpenGL imlpementations.

The VM requires OpenGL version 1.2 or later.

2.2.3. Qt

Qt™ [Qt] is a library for building graphical user interfaces (GUIs) under X-windows™ and Microsoft® Windows®. It supplies the widgets (components), e.g. windows, buttons and other controls, which together form the GUI of the VM. The VM requires Qt version 2.2 or later, which includes dedicated components (windows) for OpenGL rendering.

Qt is available under different conditions on different architectures. For the X-windows system, which is the sole target windowing system for this release of the VM, there is a freely available version distributed under the GNU general public licence or the Qt public licence. There is also a commercial licence available, for the production of commercial software. For Microsoft Windows, there is only a commercial version available

The freely available version of Qt can be downloaded over the Internet.

2.2.4. QGLViewer

The QGLViewer [QGLViewer] package provide a sophisticated Qt widget for 3D OpenGL rendering. Apart from the 3D rendering area it also provides controls for rotation and zooming. Rotation can also be controlled directly by using the mouse.

The QGLViewer package is distributed under the GNU general public licence and is freely available on the Internet. The VM requires version 1.1 of the package.

2.3. Installing the VM

The VM software is supplied in a package that also includes configuration scripts designed to facilitate the installation. These scripts use tools like automake and autoconf, which are standard on most Linux systems, but which are also available under other UNIX dialects.

Using these installation scripts, assuming that the necessary tools are available, the only additional information that needs to be supplied are ways of accessing (paths to) the software package that are used by the VM, as described in previous section.

Detailed installation instructions are supplied as a part of the complete VM software package.

3. Future Releases

Future releases will at first aim for meeting all the basic requirements outlined in the requirements specification [Svensen00]. Of particular importance is the incorporation of dedicated visualisation modes for specific data objects, such as electrodes and dipoles, and the possibility to visualise several objects together.

Once all the initial requirements have been met, work will be concentrated at the inclusion of more advanced features, such as an interface to the ImageMagick library and the generation of animations. In addition, some of the fairly basic visualisation algorithms in the current release will be replaced by more advanced ditto, yielding an improved performance of the VM.

References

[SimBio00] The SimBio Consortium (2000), *SimBio: A generic environment for bio-numerical simulation*, `http://www.simbio.de/`

[Kruggel00] F. Kruggel and M. Svensén (2000), *File Format Conventions for the SimBio project*, `http://www.simbio.de/`

[Svensen00] M. Svensén and F. Kruggel (2000), *The SimBio Visualisation Module Requirements Specification*, `http://www.simbio.de/`

[Vista] The Vista web-site:
`http://www.cs.ubc.ca/nest/lci/vista/vista.html`

[OpenGL] The OpenGL web-site: `http://www.opengl.org/`

[Qt] The Qt web-site: `http://www.trolltech.com/products/qt/`

[QGLViewer] The QGL Viewer web-site: `http://www.qglviewer.de/`