JDesigner SBW Interface

Herbert M Sauro
hsauro@cds.caltech.edu

Version of November 10, 2001

1 Introduction

JDesigner is a tool for the visual construction of biochemical networks. This document describes the SBW interface which allows other applications to control JDesigner either remotely or locally on the same machine.

2 SBW Services and Methods

JDesigner exports three services, `jsys`, `model` and `visual`. `jsys` provides basic system functionality, for example loading and saving model files. The `model` service provides access to the properties of the model that is currently loaded. `visual` provides visual based methods which a remote process to control the appearance of the model and to reposition the network nodes and edges on screen.

Some messages require colour information as part of their argument list. These colour values are represented by 32-bit integers. Each colour value is constructed from three bytes representing the intensity of the three primary colours, red, green and blue, the top byte is not used and should be set to zero. Programatically colour values are defined as

```
struct rgb (
    BYTE bRed,    // red component of color
    BYTE bGreen,  // green component of color
    BYTE bBlue,   // blue component of color
    BYTE dummy    // Ignore
   );
```

Jarnac has a function called rgb() which given the three primary colours will return a colour value, eg `red = rgb (255, 0, 0);`

The following table describes the functions and methods associated with the exported services:

Service	Method	Description	Arguments	Return Values
jsys	`string getVersion()`	get version string	—	return version string
	`string loadFile (string)`	load a model file	file name	return status message as a string
	`string saveFile (string)`	save a model file	file name	return status message as a string
	`void newPathway()`	clear current model	—	—
model	`string getSBML()`	get SBML for current model	—	SBML string
	`int getnReactions()`	get the number of reactions	—	number of reactions
	`int getnFloats()`	get the number of floating species	—	number of floating species
	`int getnFixed()`	get the number of fixed species	—	number of fixed species
visual	`string setArcColour(string, int)`	set the colour of a particular reaction	reaction name, colour	—
	`void setGlobalArcColour (int)`	set the global reaction arc colour	colour	—
	`string setNodeColour(string, int)`	set the colour of a particular species	species name, colour	—
	`void setGlobalBoundaryNodeColour(int)`	set the global colour for fixed species	colour	—
	`void setGlobalFloatNodeColour (int)`	set the global colour for floating specie	colour	—
	`void setBackColour(int)`	set the background colour	colour	—
	`void setGlobalFont(string, int, int)`	set the global font characteristics	font name, font size, font colour	—
	`void redraw()`	Issue a repaint of the network	—	—

Table 1: *List of Services and Methods exported from JDesigner*

As noted before, colour integers are constructed from three bytes representing the intensity of the three primary colours, red, green and blue.

Example code in python

```
JDesigner.jsys.loadFile ("test.dat")
str = JDesigner.model.getSBML()
```