Bug ID S-41: Graph Slowdown Bug

Table of contents

1 Problem..... 2
2 Solution..... 2
3 Modified file list..... 2
4 Code explanation..... 2
4.1 org.simBio.sim.analyzer.graph.Graph..... 2
4.2 org.simBio.sim.analyzer.graph.Axis..... 2

Copyright © 2002-2008 simBio.org All rights reserved.

1. Problem

When the matsuoka_et_al_2004/model.xml simulation runs, the graph cycles are displayed quickly to begin with, but gradually slow down over time.

2. Solution

This model has seven graphs with X-axes of different lengths, but when simBio checks if repainting is needed, it checks only one graph before repainting all the graphs. Unnecessary repaints slow down the simulation. The solution is to check each graph and only repaint those which need to be repainted.

3. Modified file list

- org.simBio.sim.analyzer.graph.Graph
- org.simBio.sim.analyzer.graph.Axis

4. Code explanation

4.1. org.simBio.sim.analyzer.graph.Graph

A call to the new method isRepaintNeeded() has been added to setAreaChanged(). If isRepaintNeeded() returns true, then the graph will be redrawn.

```
public void setAreaChanged() {
        if (isRepaintNeeded()){
                super.setAreaChanged();
                bufferRebuildFlag = true;
        }
}
```

The isRepaintNeeded() method checks the X and Y axes of the graph for scale changes using the method isScaleChanged(), which has been added to the Axis class.

```
private boolean isRepaintNeeded() {
        if ( axisX.isScaleChanged() ) return true;
        if ( axisY.isScaleChanged() ) return true;
        return false;
}
```

4.2. org.simBio.sim.analyzer.graph.Axis

In the p method of Axis, which is called before setAreaChanged(), when the data is less than min or greater than max, then the scale is changed by a value of delta, which is added to minNode and maxNode.

```
else {
        // change scale
        if (data < min) {
                double delta = (min - max) * extendRate;
```


Copyright © 2002-2008 simBio.org All rights reserved.

```
                if ( this.extendMode == this.MODE_AUTOSCROLL ) {
                        maxNode.addValue(delta); // reflect to "max"node
                }
                minNode.addValue(delta); // reflect to "min"node
        } else {
                // Data > max
                double delta = (max - min) * extendRate;
                if (this.extendMode == this.MODE_AUTOSCROLL) {
                        minNode.addValue(delta); // reflect to "min"node
                }
                maxNode.addValue(delta); // reflect to "min"node
        }
```

If the above code is executed, then the value of min no longer matches that of minNode (which has been changed by delta), and the value of max no longer matches that of maxNode. In this case, the new method isScaleChanged() returns true.

```
boolean isScaleChanged() {
        if (min != minNode.getValue() ) return true;
        if (max != maxNode.getValue() ) return true;
        return false;
}
```

The values of min and max are then updated to those of minNode and maxNode when prepareRepaint() is called.

After the code changes above, the graphs in matsuoka_et_al_2004/model.xml are repainted only when necessary and so the simulation runs at a normal speed.

Copyright © 2002-2008 simBio.org All rights reserved.