Übung 8 – Mensch-Maschine-Interaktion 2

Exercise 8: Information Visualization

Implementing a Tree Map Visualization (team work, 3-5 students)

Write a small Java application which visualizes given graphs by a tree map. A tree map visualizes a tree by representing each node by an area. If a node contains subnodes, its area is divided in subareas. An example you find at:
http://www.peets.com/selector_coffee/coffee_selector.asp
More information about tree maps you find at:
ftp://ftp.cs.umd.edu/pub/hcil/Reports-Abstracts-Bibliography/2001-06html/2001-06.pdf

For the representation of the graphs GraphML (Graph Markup Language, http://graphml.graphdrawing.org/) is used, a XML based format. It describes a graph by nodes and edges. Each edge connects two nodes.
Example:

```
<node id="n0"/>
<node id="n1"/>
<edge id="e0" source="n0" target="n1"/>
```

You can use the example file at the exercises website: http://www.medien.informatik.uni-muenchen.de/fileadmin/mimuc/mmi2_ss04/uebung/tree.xml

For processing the GraphML graphs use the *gravisto Graph VisualizationToolkit* (http://www.gravisto.org). It supports to load GraphML files and provides Java classes representing the nodes and edges. For loading a GraphML file create a new GraphMLReader and a new graph (adjacent) and load the file as follows:

```
Graph graph = new AdjListGraph();
try {
xmlReader.read(fileName, graph);
} catch (IOException e) {
}
```

You get the first node of the graph (which is the root node in *tree.xml*) with:

```
root = (Node)graph.getNodes().get(0);
```

Important methods of class Node:

```
// returns neighbours connected by outgoing edges:
public Collection getOutNeighbors()
// returns the number of outgoing (or undirected) edges:
public int getOutDegree()
```

You can use the gravisto jar library file on the exercises website:
http://www.medien.informatik.uni-muenchen.de/fileadmin/mimuc/mmi2_ss04/uebung/gravisto.jar

Create the tree map by painting on the content pane of your frame. On the exercises website, you can find a template for your Java class: http://www.medien.informatik.uni-muenchen.de/fileadmin/mimuc/mmi2_ss04/uebung/TreeMapTemplate.java

Tasks

a) As a first step build an application which loads GraphML files and visualizes only the first level of the tree.
b) The size of each area should now be as large as the number of nodes in the subtrees in relation of the overall number of nodes.
c) Now visualize the whole tree. For each level of the tree, the areas representing a node are subdivided alternatingly vertically and horizontally.
d) Find another way to distribute the available space between the nodes.
e) Try to find possible alternative solutions to visualize a hierarchical graph by a compact representation on a given area.

Submit your solution of **task c)** as a single Java file called *Vorname_Nachname.java*. Submit a short solution of **tasks d) and e)** as a single pdf file called *vorname_nachname.pdf*. Please send both documents via email to Andreas Pleuss (**Andreas.Pleuss@ifi.lmu.de**).
You may solve the tasks in team work (**3-5 students**). Solution deadline is **2nd of July 2004**.