***IHM***

# Contents



# 1 Where to store the global variables

In order to enable multiples entry points from our sources (for instance one main function for a GUI and on other for a command-line application), we define a static class that old the main independents objects.

```
package nectar;
import javax.swing.*;

public class Main {
    static final String confFile = "etc/nectar.xml";
    static UI_Camera camera = null;
    static public JPanel gui = null;
    static Semaphore semSelect = null;
}
```

As it, we may develop several GUI that derive from the same interface (wich as to be defined). For now we use casting to call the GUI's functions from the other objects :

```
((GUI_GraphicalEmbryo) Main.jPanel).plotEvtCounter(counter);
```

# 2 Entry point for GUIs

Here is the `Main` class. It give us a first `JPanel` empty object and the `matchPanel` function that allow to fill it with any GUI object.

```
package nectar;
import javax.swing.*;
import java.awt.*;

public class Main extends javax.swing.JFrame {

    // add a JPanel into a container (build the GUI object tree)
    static void matchPanel(Container target, JPanel pluggin) {
        GroupLayout logPanelLayout = new GroupLayout(target);
        target.setLayout(logPanelLayout);
        logPanelLayout.setHorizontalGroup(
            logPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
            .addComponent(pluggin, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,
            );
        logPanelLayout.setVerticalGroup(
            logPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
            .addComponent(pluggin, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,
            );
    }
```

```
    public Main() {
        initComponents();

        if (Main.gui != null) {
            Main.matchPanel(getContentPane(), Main.gui);
        }
        setVisible(true);
    }

    private void initComponents() {
        setDefaultCloseOperation(javax.swing.WindowConstants.EXIT_ON_CLOSE);
        javax.swing.GroupLayout layout = new javax.swing.GroupLayout(getContentPane());
        getContentPane().setLayout(layout);
        layout.setHorizontalGroup(
            layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
            .addGap(0, 785, Short.MAX_VALUE)
        );
        layout.setVerticalGroup(
            layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
            .addGap(0, 427, Short.MAX_VALUE)
        );
        pack();
    }

    // Creates the main windows
    public static void main(String args[]) {
        java.awt.EventQueue.invokeLater(new Runnable() {
            public void run() {
                if (Main.gui == null) {
                    UT_GraphicalEmbryo.ut(); // the default entry point.
                }
                new Main();
            }
        });
    }
}
```

The `Main.main()` function will be our entry point call by all the GUI application (unit tests too). For instance :

```
public class GUI_Log extends JPanel {
    ...
    // Unitary test for GUI_Log
    public static void main(String args[]) {
        Main.jPanel = new GUI_Log();
        Main.main(null);
    }
```