A General-Purpose Computer Graphics Development Environment

G. Anastassakis

Department of Informatics
University of Piraeus

2010

Requirements

- Cross-platform
  - ✓ Focus on computer graphics, not on platform
- Intuitive
  - ✓ Makes you understand what's actually going on
- Simple and easy to program
  - ✓ Focus on graphics, not on the development process
- Windows, icons, menus, pointers (WIMP)
  - ✓ Computer graphics look good inside windows
  - ✓ Interactive user-interface elements always come in handy
- Flexible and extendible
  - ✓ Don't reinvent the wheel
  - ✓ Make your own wheels reusable

Design guidelines

- Cross-platform
  - ✓ Java-based
- Intuitive
  - ✓ Shows axes, allows zooming and panning, and more
- Simple and easy to program
  - ✓ Standard skeletal structure with placeholders
- Windows, icons, menus, pointers (WIMP)
  - ✓ Main window with controls and messages area
  - ✓ General-purpose cross-hair cursor
- Flexible and extendible
  - ✓ Use libraries
  - ✓ Structure smart and plan ahead

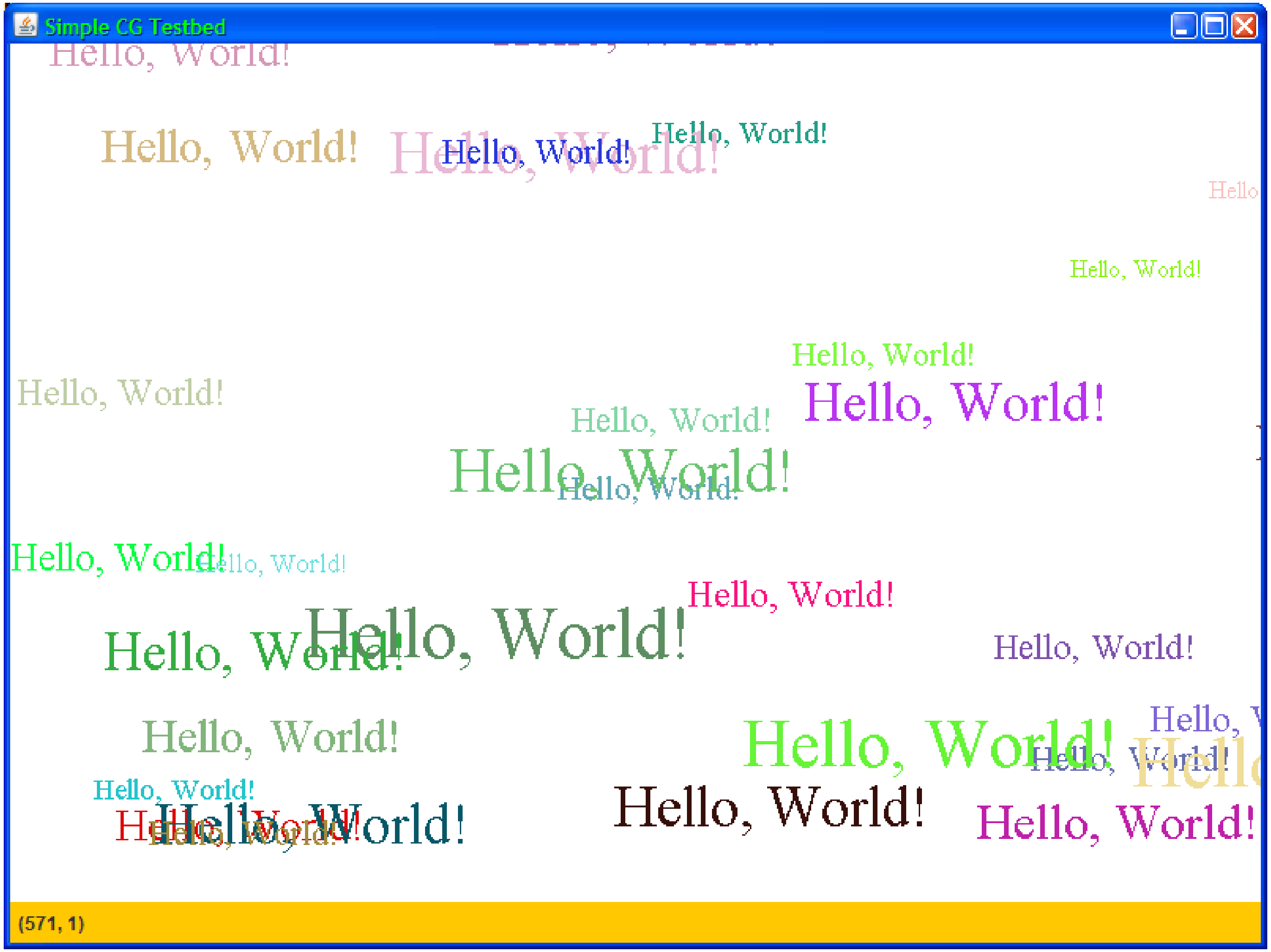
Development tools

- Text editor
  - ✓ Notepad, Wordpad
  - ✓ gedit, Kate
  - ✓ Notepad++
  - ✓ Etc.
- J2SE SDK (JDK)
  - ✓ Java compiler (javac.exe, javac)
  - ✓ Java application launcher (java.exe, java)
- Free to use your preferred IDE
  - ✓ ... but don't ask me questions about how to use it

Simple CG testbed

- Very simple structure
  - ✓ Only capable of performing a single drawing task
  - ✓ Drawing task is programmed directly into the source code
- Contains a canvas and a status bar
  - ✓ Canvas provides space for the drawing task
  - ✓ Status bar displays mouse coordinates on canvas
- Canvas implemented as an overriden JPanel
  - ✓ paint(Graphics g) method calls drawTask (Graphics g) method which is a placeholder for drawing code

Simple CG testbed's graphical user-interface



Simple CG testbed source code (1 of 5)

```java
package simplecgtestbed;

import java.awt.BorderLayout;
import java.awt.Color;
import java.awt.Container;
import java.awt.Cursor;
import java.awt.Dimension;
import java.awt.FlowLayout;
import java.awt.Font;
import java.awt.Graphics;
import java.awt.event.MouseEvent;
import java.awt.event.MouseMotionAdapter;
import javax.swing.JFrame;
import javax.swing.JLabel;
import javax.swing.JPanel;

public class SimpleCGTestbed {

    ...

}
```

Simple CG testbed source code (2 of 5)

```java
public class SimpleCGTestbed {

    private final JLabel coords = new JLabel("(x, y)");

    public SimpleCGTestbed() {
        ...
    }

    public static void main(String[] args) {
        ...
    }

    private class Canvas extends JPanel {

        public void paint(Graphics g) {
            ...
        }

        public void draw(Graphics g) {
            ...
        }
    }
}
```

Simple CG testbed source code (3 of 5)

```
public SimpleCGTestbed() {

    JFrame frame = new JFrame("Simple CG Testbed");
    frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
    frame.setSize(800, 600);
    frame.setLocationRelativeTo(null);

    JPanel statusBar = new JPanel(new FlowLayout(FlowLayout.LEFT));
    statusBar.setBackground(Color.ORANGE);

    statusBar.add(coords);

    Canvas canvas = new Canvas();
    canvas.setBackground(Color.WHITE);
    canvas.setCursor(Cursor.getPredefinedCursor(Cursor.CROSSHAIR_CURSOR));
    canvas.addMouseMotionListener(new MouseMotionAdapter() {
        public void mouseMoved(MouseEvent e) {
            coords.setText("(" + e.getX() + ", " + e.getY() + ")");
        }
    });

    Container contentPane = frame.getContentPane();
    contentPane.setLayout(new BorderLayout());
    contentPane.add(statusBar, BorderLayout.SOUTH);
    contentPane.add(canvas, BorderLayout.CENTER);

    frame.setVisible(true);
}
```

Simple CG testbed source code (4 of 5)

```
public static void main(String[] args) {

    SimpleCGTestbed testbed = new SimpleCGTestbed();
}
```

Simple CG testbed source code (5 of 5)

```
private class Canvas extends JPanel {

    public void paint(Graphics g) {

        super.paint(g);

        draw(g);
    }

    public void draw(Graphics g) {

        // TODO: write code for your task here...
    }
}
```

Simple CG testbed project structure

```
SimpleCGTestbed/
  src/
    simplecgtestbed/
      SimpleCGTestbed.java
  classes/
    simplecgtestbed/
      SimpleCGTestbed$1.class
      SimpleCGTestbed$Canvas.class
      SimpleCGTestbed.class
```

Legend: created by you; automatically-generated.

Compilation and execution

- Compilation

```
javac -d classes src\simplecgtestbed\*.java
```

- Execution

```
java -cp classes simplecgtestbed.SimpleCGTestbed
```