

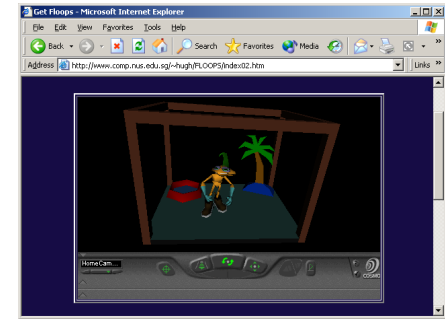


## Chapter 12

# Blending...



## FLOOPS



## Blending languages



- ✓ SWIG - Simplified Wrapper and Interface Generator
- ✓ S/W tool that connects C and C++ with scripting languages
- ✓ Generates wrapper code that scripting languages use to access C/C++.
- ✓ Perl, Python, Tcl/Tk, Ruby, Java, OCAML, C# ...



## Today though...



- ✓ Perl and Tk
- ✓ C and Tk
- ✓ VRML and Java
- ✓ Java and Tk



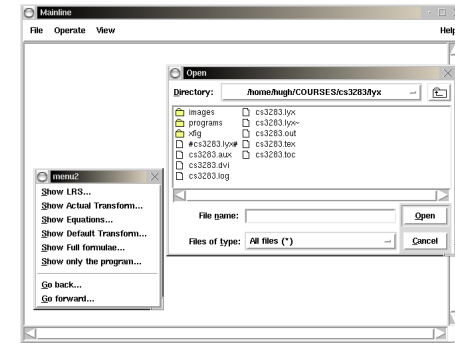
## Perl and Tk



- ✓ Have to install the Tk module for perl
- ✓ Download Tk-804.026.tar.gz, and then
  - ✓ perl Makefile.pl XFT=1
  - ✓ make
  - ✓ make install



## GUI



## Perl/Tk code



Perl has a Tk module:

CODE LISTING	mainline.pl
<pre> use Tk; my \$currentslice = 0; my \$currentpp = 0; my \$disptype = 2; my \$main = new MainWindow;  &lt;&lt;SetupMenu&gt;&gt; &lt;&lt;SetupFileMenu&gt;&gt; &lt;&lt;SetupEditMenu&gt;&gt; &lt;&lt;SetupViewMenu&gt;&gt;  \$main-&gt;configure(-menu =&gt;\$menubar);  &lt;&lt;SetupScrolledMainArea&gt;&gt;  MainLoop; &lt;&lt;FileOpenDialogBox&gt;&gt; </pre>	



## Menu bar



CODE LISTING	SetupMenu.pl
<pre> \$menubar = \$main-&gt;Menu; \$filemenu = \$menubar-&gt;cascade(-label=&gt;"File" ); \$editmenu = \$menubar-&gt;cascade(-label=&gt;"Operate" ); \$viewmenu = \$menubar-&gt;cascade(-label=&gt;"View" ); \$helpmenu = \$menubar-&gt;cascade(-label=&gt;"Help"); \$helpmenu-&gt;command(-command =&gt; \&amp;about_choice,                   -label =&gt; "About TkMenu...",                   -underline =&gt; 0); </pre>	



## Menu items



CODE LISTING

SetUpFileMenu.pl

```

$filenmenu->command(-command => sub { fileDialog( $main, 'open' );
    printf "Opening %s\n", $thisfile;
    readfile($thisfile);
    writefile($thisfile . ".ppx");},
    -label => "Open...",
    -underline => 0);
$filenmenu->separator;
$filenmenu->command(-label => "Exit",
    -command => \&exit_choice,
    -underline => 1);

```



## Edit menu



CODE LISTING

SetUpEditMenu.pl

```

$editmenu->command(-command => sub {Tp($currentslice,1,1);},
    -label => "Crank with widening...",
    -underline => 0);
$editmenu->command(-command => sub {Tp($currentslice,1,10);},
    -label => "Crank with widening (10X)...",
    -underline => 0);
$editmenu->command(-command => sub {Tp($currentslice,0,1);},
    -label => "Crank...",
    -underline => 0);
$editmenu->command(-command => sub {Tp($currentslice,0,10);},
    -label => "Crank (10X)...",
    -underline => 0);
$editmenu->command(-command => sub {Cousot($currentlice);},
    -label => "Cousot...",
    -underline => 0);
$editmenu->separator;
$editmenu->command(-command => sub {widening($currentlice);},
    -label => "Widen...",
    -underline => 0);

```



## View menu



CODE LISTING

SetUpViewMenu.pl

```

$viewmenu->command(-command => sub { $disptype=0; display($currentslice,0);},
    -label => "Show LRS...",
    -underline => 0);
$viewmenu->command(-command => sub { $disptype=1; display($currentslice,1);},
    -label => "Show Actual Transform...",
    -underline => 0);
$viewmenu->command(-command => sub { $disptype=2; display($currentslice,2);},
    -label => "Show Equations...",
    -underline => 0);
$viewmenu->command(-command => sub { $disptype=3; display($currentslice,3);},
    -label => "Show Default Transform...",
    -underline => 0);
$viewmenu->command(-command => sub { $disptype=4; display($currentslice,4);},
    -label => "Show Full formulae...",
    -underline => 0);
$viewmenu->command(-command => sub { $disptype=5; display($currentslice,5);},
    -label => "Show only the program...",
    -underline => 0);
$viewmenu->separator;
$viewmenu->command(-command => sub { if ($currentslice>0) {
    $currentlice=$currentlice-1;
    if ($currentpp=0) { $codesize;
    } $currentpp=$codesize;
    } $currenttype=$currentpp-1;
    display($currentslice,$disptype);
    -label => "Go back..."},
    -underline => 0);
$viewmenu->command(-command => sub { if ($currentslice<=$maxlice){
    $currentlice=$currentlice+1;
    $currentpp=$currentpp+1;
    if ($currentpp=$codesize) {
    } $currentpp=0;
    } display($currentlice,$disptype);
    -label => "Go forward..."},
    -underline => 0);

```



## Dialog box



CODE LISTING

FileOpenDialogBox.pl

```

sub exit_choice {
    exit;
}

sub fileDialog {
    my $w = shift;
    my $operation = shift;
    my $types = my $file;
    @types = ("Code files", ".pp"),
    ["Work files", ".ppx"],
    ["All files", "*.*"];
    $file = $w->getOpenFile(-filetypes => \@types);
    if (defined $file and $file ne "") {
        $thisfile = $file;
    }
}

```



## Code



<http://www.comp.nus.edu.sg/~cs3283/ftp/original.pl>  
It may be run by typing **“perl original.pl”**.



## C and Tk



- ✓ There are various ways...
- ✓ Call C within Tk
- ✓ Call Tk within C
- ✓ Communication



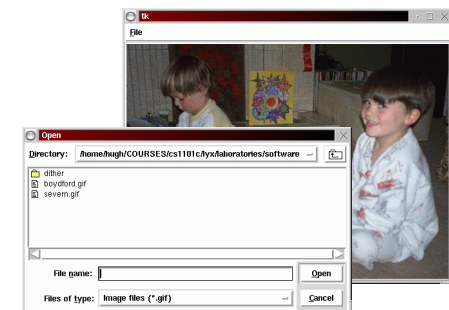
## C and Tk



- ✓ This one is a C program
- ✓ It loads and runs a Tk program along with its interpreter.
- ✓ Tk in turn can call-back C procedures if needed



## Viewer







## VRML software



CODE LISTING	defaulthtml.txt	Page 1/1
<pre> &lt;html&gt; &lt;head&gt; &lt;title&gt;Sample 3DVNT Page&lt;/title&gt; &lt;/head&gt; &lt;center&gt;&lt;H1&gt;Sample 3DVNT Page &lt;/H1&gt;&lt;/center&gt; &lt;center&gt;&lt;embed src="root.wrl" height="600" width="700"&gt; &lt;/center&gt; &lt;center&gt;&lt;applet code="View.class" width="100" height="10" mayscript&gt; &lt;PARAM name="segment" value="MACS"&gt; &lt;PARAM name="port" value="9876"&gt; &lt;PARAM name="host" value="opo.uspac.fj"&gt; &lt;/applet&gt; &lt;/center&gt; OK? &lt;/html&gt; </pre>		



## The root.wrl file



CODE LISTING	root.wrl	Page 1/1
<pre> PROTO CLUSTER [] { ... } # Cluster definition PROTO KEYBOARD [] { ... } # Keyboard definition PROTO SCREEN [] { ... } # Screen definition PROTO GLOBE [] { ... } # Traffic sphere definition # Some setting up declarations Background { skyColor .4 .66 1 } NavigationInfo { type [ "EXAMINE", "ANY" ] speed 400 } Viewpoint { position 0 400 0 orientation 0 1 0 4 description "Camera1" } # Lines, floors and roofs DEF LINES Transform { ... } DEF FLOORS Transform { ... } DEF ROOFS Transform { ... } # and then the nodes DEF node1 Transform { ... } DEF node2 Transform { ... } # ... and so on ... </pre>		



## VRML nodes



CODE LISTING	node.wrl	Page 1/1
<pre> DEF node1 Transform {   translation 4350 150 4365   rotation 0 1 0 4.71238   children {     KEYBOARD {}     SCREEN {}     DEF node1box Transform {       children {         Shape {           appearance Appearance {             material DEF node1boxcolor Material {               diffuseColor 0.8 0.8 0.8             }           }           geometry Box { size 50 50 50 }         }       }     }     DEF node1sphere Transform {       scale 1 1 1       children {         Shape {           appearance GLOBE {}           geometry Sphere { radius 1 }         }       }     }   } } </pre>		



## Java



The java software maintains a link to a remote data collector, Uses the EAI to modify the images in the VRML view.



## Java 1Java 1



```

Mar 05, 09 11:51 View1.java Printed by Hugh Anderson Page 3/3
// Mar 05, 09 11:51 View1.java
import java.awt.*;
import java.awt.event.*;
import javax.swing.*;

public class View1 implements ActionListener {
    private JFrame frame;
    private JButton button;
    private JLabel label;

    public View1() {
        frame = new JFrame("TestApp");
        frame.setSize(300, 300);
        frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        frame.setLayout(new BorderLayout());

        label = new JLabel("Click the button");
        button = new JButton("Click Me");
        button.addActionListener(this);

        frame.add(label, BorderLayout.CENTER);
        frame.add(button, BorderLayout.SOUTH);

        frame.setVisible(true);
    }

    public void actionPerformed(ActionEvent e) {
        System.out.println("Button clicked");
    }

    public static void main(String[] args) {
        View1 view1 = new View1();
    }
}
Thursday August 26, 1999 1/2

```



## Java 2



```

Mar 05, 09 11:51 View1.java Printed by Hugh Anderson Page 2/3
// Mar 05, 09 11:51 View1.java
import java.awt.*;
import java.awt.event.*;
import javax.swing.*;

public class View1 implements ActionListener {
    private JFrame frame;
    private JButton button;
    private JLabel label;
    private JTextField textField;

    public View1() {
        frame = new JFrame("TestApp");
        frame.setSize(300, 300);
        frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        frame.setLayout(new BorderLayout());

        label = new JLabel("Click the button");
        button = new JButton("Click Me");
        textField = new JTextField(20);

        frame.add(label, BorderLayout.CENTER);
        frame.add(textField, BorderLayout.NORTH);
        frame.add(button, BorderLayout.SOUTH);

        button.addActionListener(this);

        frame.setVisible(true);
    }

    public void actionPerformed(ActionEvent e) {
        System.out.println("Button clicked");
    }

    public static void main(String[] args) {
        View1 view1 = new View1();
    }
}
Thursday August 26, 1999 2/3

```



## Java 3



```

Mar 05, 09 11:51 View1.java Printed by Hugh Anderson Page 3/3
// Mar 05, 09 11:51 View1.java
import java.awt.*;
import java.awt.event.*;
import javax.swing.*;

public class View1 implements ActionListener {
    private JFrame frame;
    private JButton button;
    private JLabel label;
    private JTextField textField;
    private int count;

    public View1() {
        frame = new JFrame("TestApp");
        frame.setSize(300, 300);
        frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        frame.setLayout(new BorderLayout());

        label = new JLabel("Click the button");
        button = new JButton("Click Me");
        textField = new JTextField(20);

        frame.add(label, BorderLayout.CENTER);
        frame.add(textField, BorderLayout.NORTH);
        frame.add(button, BorderLayout.SOUTH);

        button.addActionListener(this);

        frame.setVisible(true);
    }

    public void actionPerformed(ActionEvent e) {
        System.out.println("Button clicked");
        count++;
        textField.setText(count + "");
    }

    public static void main(String[] args) {
        View1 view1 = new View1();
    }
}
Thursday August 26, 1999 3/3

```



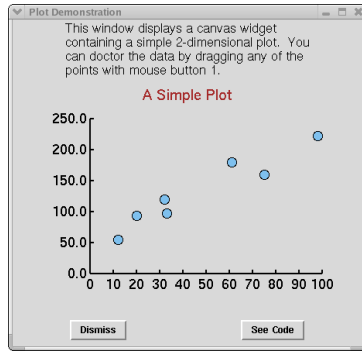
## Java/Tk code



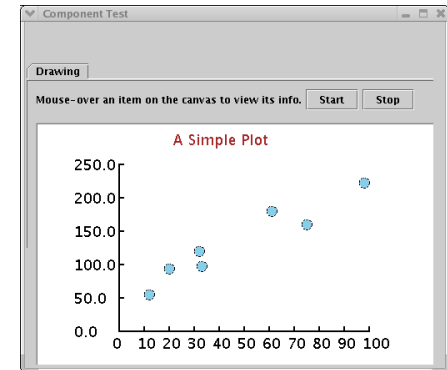
- ✓ As for C... we have
  - ✓ Java calls to Tk
  - ✓ Tk calls to Java
- ✓ Partial implementation of Tk inside Java/Swing



## Tk code



## Java/Tk code



## Tk code



```

CODE LISTING      tkplot.tcl      Page 1/1
...
canvas $c -relief raised -width 450 -height 300
set plotFont {Helvetica 18}
...
$c create line 100 250 100 50 -width 2
$c create text 225 20 -text "A Simple Plot" -font $plotFont -fill brown
...
for {set i 0} {$i <= 10} {incr i} {
    set x [expr {100 + ($i*30)}]
    $c create line $x 250 $x 245 -width 2
    $c create text $x 254 -text [expr 10*$i] -anchor n -font $plotFont
}
...
$c bind point <Any-Enter> "$c itemconfig current -fill red"
...
proc plotMove {w x y} {
    global plot
    $w move selected [expr $x-$plot(lastX)] [expr $y-$plot(lastY)]
    set plot(lastX) $x
    set plot(lastY) $y
}

```



## The JavaKit code



```

CODE LISTING      tkplot.java      Page 1/1
...
C = new canvas(jf, width, height, bgColor);
plotFont = "-font [fontName Helvetica fontSize 18 fontStyle plain]";
...
C.create("-item line -coords [100 250 100 50] -width 2.0");
C.create("-item text -coords [225 20] -text \"A Simple Plot\" + plotFont );
...
for(int i = 0; i <= 10; i++) {
    int x = 100 + (i*30);
    C.create("-item line -coords [ + x + * 250 * + x + * 245] -width 2.0");
    C.create("-item text -coords [ + (x+9) + * 264] -text * + (10*i) + * + plotFont );
}
...
C.bind("point", new command("enter") {
    public void action(EventInfo ei) { C.itemConfig("current", "-fill red"); });
}
...
public void plotMove(canvas w, double x, double y) {
    w.move("selected", x - lastX, y - lastY);
    lastX = x; lastY = y;
}

```





## Summary of topics



- ✓ Tk blended with C
- ✓ Tk blended with perl
- ✓ VRML, Java and the EAI
- ✓ Tk blended with Java