EXHIBIT A

JDX 274

PA-NAT-00000115

```
<!DOCTYPE hmml SYSTEM>
<HMML>
<GROUP>www</GROUP>
<TITLE>ViolaWWW Notes</TITLE>
<H1>ViolaWWW Alpha Release Notes</H1>
<P>
This is an alpha release of violaWWW. The primary intent of this
release is to provide a demo for the Design team to evaluate, to
provide user interface feedbacks, etc. The general look & feel is
expected to be <ITALIC>Motif'</ITALIC>ed soon.
<P>
<EMPH>As is</EMPH>, this browser has trouble working with the HTML
documents from the network, unless they have the necessary SGML document
type declaration. This problem is being dealt with. Meanwhile, you can
try out the documents listed below.

<H2>A few things to try out:</H2>
<P>
Under the rotating globe menu, the ViolaWWW ''about'' and ''help'' documents
are fairly good representative of HMML.
<P>
Under the <ITALIC>Local</ITALIC> bookmarks menu:
<DL>
<DT><A HREF="/home/wei/viola/docs/gnnElements.hmml">gnnElements.hmml</A>
<DD>A demo GNN header logo thingy.
<DT><A HREF="/home/wei/viola/docs/testAll.html">testAll.html</A>
<DD>Sample of HTML document.
<DT><A HREF="/home/wei/viola/docs/violaBrief.hmml">violaBrief.hmml</A>
<DD>A very long (too long) and <EMPH>drafty</EMPH> document on viola.
Has embedded GUIs.
<DT><A HREF="/home/wei/viola/inetRC/inetRC.hmmlb">InternetRC (all)</A>
<DD>An even longer document. Practically the whole InetRC catalog section.
<DT><A HREF="/home/wei/viola/gallery/collection.hmml">Art Gallery</A>
<DD>Was a demo for GIF, but the GIF code used in viola is broken...
So the pictures in it are now temporarily in XBM.
<DT><A HREF="/home/wei/viola/docs/violaChier.hmml">violaChier.hmml</A>
<DD>Shows a tree graph, with hot-link tree nodes.
<DT><A HREF="/home/wei/viola/apps/doodle.v">Doodle</A>
<DD>A doodling "program" as WWW "doc".
<DT><A HREF="/home/wei/viola/apps/vwq.v">Viola Survey</A>
<DD>Demos a fill-in/mail-out forms. Currently implemented as viola object,
but will be abstracted to be portable SGML, as in <CMD>&lt;ENTRY&gt</CMD>.
A related idea is the <A HREF="/home/wei/viola/apps/ORABKOrder.v">
ORA Book Order Demo</A>.
<DT><A HREF="/home/wei/viola/apps/wave.v">Wave</A>
<DD>Sine wave ploting as WWW doc. Just a GUI toy, but demonstrates WWW
as transport infrastructure not just for static documents.
<DT><A HREF="/home/wei/viola/apps/clock.v">Clock</A>
<DD>A real working time telling clock as WWW ''doc''.
<DT><A HREF="/home/wei/viola/docs/testPlot.hmml">testPlot.hmml</A>
<DD>Demos a highly cool wire-frame ploting widget embeded in a document.
<EMPH>If you try this, you must exit viola properly (do not exit viola
with control-C), else the X server will be anguished.</EMPH>
</DL>
<P>
Please ignore the <ITALIC>Public</ITALIC> and <ITALIC>Private</ITALIC>
bookmarks for now.
<P>
Well, development continues! If you have any comments, suggestions,
bug-reports, whatever, at all, please feel free to send them my way.

<ADDRESS><P>Pei
<P><KBD>pei@ora.com</KBD>
</ADDRESS>
</HMML>

```

```
<!DOCTYPE html SYSTEM "html.dtd">
<HTML>
<TITLE>Test misc</TITLE>
<H1>Header 1</H1>
<H2>Header 2</H2>
<H3>Header 3</H3>
See the <A HREF="/etc/motd">motd</A> for useless messages.
<XMP>
LISTING   HTML_listing   0    0 10 10 25
TITLE     HTML_title     0    0 10 10 25
</XMP>
Here is an ordered list:
<OL>
<LI>socks
<LI>scarf
<LI>pillows
</OL>
And an unordered list.
<UL>
<LI>rubber ball.
<LI>I recollect that, when a stripling, my first exploit in
squirrel-shooting was in a grove of tall walnut-trees that shades
one side of the valley.  I had wandered into it at noontime, when
all nature is peculiarly quiet, and was startled by the roar of
my own gun, as it broke the Sabbath stillness around and was
prolonged and reverberated by the angry echoes.
<LI>If ever I should
wish for a retreat whither I might steal from the world and its
distractions, and dream quietly away the remnant of a troubled
life, I know of none more promising than this little valley.
</UL>
Definition lists:
<DL>
<DT>First term
<DD>Definition paragraph starts here.
<DT>Second term
<DD>Second definition paragraph starts here.
</DL>
Menu:
<MENU>
<LI>cinamon crepe
<LI>zinc coated pop corn.
<LI>rasberry spring water
</MENU>
Here is a dir (x2):
<DIR>
<LI>socks 0
<LI>scarf 1
<LI>pillows 2
<LI>apple 3
<LI>kiwi 4
<LI>banana 5
<LI>pinball 6
</DIR>
<ADDRESS>Angel666
<P><KBD>angel666@alpha.div</KBD>
</ADDRESS>
</HTML>
```

```
<!DOCTYPE hmml SYSTEM>
<TITLE>Test</TITLE>
<H1>List No. 5</H1>
<P>
The <CMD>&lt;VOBJF&gt;</CMD> tag can be used to insert viola applications.
Using this capability allows you embed in your document what you can
access or build using viola's programming, and GUIs. Of course too much
violaism reduces the portability of your document on the World Wide Web,
but anyway...
</P>
<P>
Here are some examples.
<H2>Clock</H2>
<VOBJF>/home/wei/viola/apps/clock.v</VOBJF>
<H2>Vicon</H2>
<VOBJF>/home/wei/viola/apps/vicon.v</VOBJF>
<P>
This can be a handy menu to tuck away at a corner of the screen.
</P>
<H2>Query</H2>
<VOBJF>/home/wei/viola/apps/vwq.v</VOBJF>
<P>
This application is intended to gather user information.
</P>
<H2>Wave fun</H2>
<VOBJF>/home/wei/viola/apps/wave.v</VOBJF>
<H2>Noodle Doodles</H2>
<VOBJF>/home/wei/viola/apps/doodle.v</VOBJF>
<P>
So I was bored...
</P>
<P>
The end.
</P>
```

```
<!DOCTYPE hmml SYSTEM>
<HMML>
<TITLE>RandoMyster's Waves</TITLE>
<DL>
<DT>Title<DD>Brush Motion
<DT>Artist<DD>Rando "random art does not apologize" Myster
<DT>Date<DD>1993, March 7, before breakfast
</DL>
<VOBJF>colorwavy.v</VOBJF>
<P>
Warning: playing with color too much will take all the color resources...
(Yes, this is a bug).
</HMML>
```

```
<!DOCTYPE hmml SYSTEM>
<HMML>
<GROUP>www</GROUP>
<TITLE>XPlot</TITLE>
<H1>An equation and vector-objects plotting program</H1>
<P>
The <CMD>xplot</CMD> program was written by Scott Silvey at the
Experimental Computing Facility of UCB. In an experiment,
<CMD>xplot</CMD> was made accessible by other program, such as viola.
<P>
Here's a viola front-end to <CMD>xplot</CMD>, which is embedded
into <ITALIC>this</ITALIC> document.
<VOBJF>/home/wei/viola/apps/plot.v</VOBJF>
<EXAMPLE>
file /home/wei/vplot/off/x29.geom
equation .4 * sin(.4 * x + y)
equation .4 * sin(1 - x*x - y*y)
equation .4 * exp(1 - x*x - y*y) * sin(1 - x*x - y*y)
equation .4 * exp(1 - x*x - y*y) * sin(1 - x^4 - y^4)
</EXAMPLE>
</HMML>

```

```
\class {vpane}
\name {plot}
\script {
        switch (arg[0]) {
        case "build":
                /* arg[1]       sourcefile
                 * arg[2]       parent
                 * arg[3]       name
                 * arg[4]       width
                 * ret          documentObject or 0
                 */
                docObj = plot.info("clone");
                send(docObj, "build", arg[1], arg[2], arg[3], arg[4]);

                if (isBlank(docObj) == 1) {
                        www.mesg.tf("show",
                              concatenate("Failed to get ", arg[1]));
                        cursorShape("idle");
                        return 0;
                } else {

                        www.mesg.tf("show", "");
                        www.udi.tf("show", arg[1]);

                        return docObj;
                }
        break;
        case "TTYObjP":
                return nthChild(0);
        break;
        case "canvas_WIDP":
        case "canvas_widthP":
        case "canvas_heightP":
                return send(nthChild(2), arg[0]);
        break;
        }
        usual();
}
\children {plot.TTY plot.head plot.body plot.cmd}
\width {500}
\height {500}
\
\class {txtLabel}
\name {plot.head}
\parent {plot}
\label {XPlot}
\maxHeight {18}
\font {normal}
\BGColor {LemonChiffon2}
\BDColor {LemonChiffon3}
\FGColor {black}
\
\class {hpane}
\name {plot.body}
\parent {plot}
\children {plot.body.view plot.body.ctrl}
\script {
        switch (arg[0]) {
        case "canvas_WIDP":
        case "canvas_widthP":
        case "canvas_heightP":
                return send(nthChild(0), arg[0]);
        break;
        case "TTYObjP":
                return send(parent(), "TTYObjP");
        break;
        }
        usual();
}
\BGColor {black}
\
\class {field}
\name {plot.body.view}
\parent {plot.body}
\BGColor {LightSteelBlue4}
```

```
\BDColor {LemonChiffon2}
\FGColor {black}
\script {
        switch (arg[0]) {
        case "canvas_WIDP":
                return get("window");
        break;
        case "canvas_widthP":
                return get("width");
        break;
        case "canvas_heightP":
                return get("height");
        break;
        case "clear":
                clearWindow();
                return;
        break;
        case "expose":
        case "config":
                usual();
                if (get("window") != 0) {
                        TTYObj = send(parent(), "TTYObjP");
/*                      send(TTYObj, "initClient");
*/
                        after(1, TTYObj, "initClient");
                }
/*
                width = send(parent(), "canvas_widthP");
                height = send(parent(), "canvas_heightP");
                send(TTYObj, "output",
                        concat("window ", get("window"), " ",
                                get("width"), " ", get("height")));
                print(">>>",
                        concat("w ", get("window"), " ",
                                get("width"), " ", get("height")), "\n");
*/
                return;
        break;
        }
        usual();
}
\
\class {vpane}
\name {plot.body.ctrl}
\parent {plot.body}
\children {plot.body.ctrl.ops plot.body.ctrl.cam plot.body.ctrl.quit}
\script {
        switch (arg[0]) {
        case "TTYObjP":
                return send(parent(), "TTYObjP");
        break;
        case "SUPER_HACK":
                return send(nthChild(0), "SUPER_HACK");
        break;
        }
        usual();
}
\BGColor {LemonChiffon2}
\maxWidth {60}
\gapH {2}
\
\class {menu}
\name {plot.body.ctrl.ops}
\parent {plot.body.ctrl}
\label {Ops}
\script {
        switch (arg[0]) {
        case "buttonRelease":
/*              send(send(parent(), "TTYObjP"), "output", "quit");
                quit();
*/
        break;
        case "x29":
                TTYObj = send(parent(), "TTYObjP");
                send(TTYObj, "output", "domain -10 10 -10 10");
                send(TTYObj, "output",
                        "file /home/wei/vplot/off/x29.geom");
                send(TTYObj, "output", "expose");
                return;
        break;
        case "TomCat":
                TTYObj = send(parent(), "TTYObjP");
                send(TTYObj, "output", "domain -10 10 -10 10");
                send(TTYObj, "output",
                        "file /home/wei/vplot/off/TomCat");
                send(TTYObj, "output", "expose");
                return;
        break;
        case "eq1":
                TTYObj = send(parent(), "TTYObjP");
                send(TTYObj, "output", "interval .2");
                send(TTYObj, "output", "domain -2 2 -2 2");
                send(TTYObj, "output",
                        "equation .4 * sin(x * y)");
                send(TTYObj, "output", "expose");
                return;
        break;
        case "eq2":
                TTYObj = send(parent(), "TTYObjP");
                send(TTYObj, "output", "interval .2");
                send(TTYObj, "output", "domain -2 2 -2 2");
                send(TTYObj, "output",
                        "equation .4 * sin(1 - x*x - y*y)");
                send(TTYObj, "output", "expose");
                return;
        break;
        case "eq3":
                TTYObj = send(parent(), "TTYObjP");
                send(TTYObj, "output", "interval .2");
                send(TTYObj, "output", "domain -2 2 -2 2");
                send(TTYObj, "output",
                    "equation .4 * exp(1 - x*x - y*y) * sin(1 - x*x - y*y)");
                send(TTYObj, "output", "expose");
                return;
        break;
        case "eq4":
                TTYObj = send(parent(), "TTYObjP");
                send(TTYObj, "output", "interval .2");
                send(TTYObj, "output", "domain -2 2 -2 2");
                send(TTYObj, "output",
                    "equation .4 * exp(1 - x*x - y*y) * sin(1 - x^4 - y^4)");
                send(TTYObj, "output", "expose");
                return;
        break;
        case "SUPER_HACK":
                set("visible", 0);
                set("visible", 1);
                return;
        break;
        }
        usual();
}
\BDColor {white}
\BGColor {grey45}
\FGColor {white}
\gapH {3}
\gapV {3}
\maxHeight {20}
\height {20}
\menuConfig {
. {file /home/wei/vplot/off/x29.geom} {send(self(), "x29");}
. {file /home/wei/vplot/off/TomCat} {send(self(), "TomCat");}
. {equation .4 * sin(x * y)} {send(self(), "eq1");}
. {equation .4 * sin(1 - x*x - y*y)} {send(self(), "eq2");}
. {equation .4 * exp(1 - x*x - y*y) * sin(1 - x*x - y*y) } {send(self(), "eq3");
}
. {equation .4 * exp(1 - x*x - y*y) * sin(1 - x^4 - y^4) } {send(self(), "eq4");
}
. {Quit} {quit();}
}
```

```
\
\class {hpane}
\name {plot.body.ctrl.cam}
\parent {plot.body.ctrl}
\children {plot.body.ctrl.cam.x plot.body.ctrl.cam.y plot.body.ctrl.cam.z}
\script {
        switch (arg[0]) {
        case "TTYObjP":
                return send(parent(), "TTYObjP");
        break;
        case "expose":
                send(parent(), "SUPER_HACK");
        break;
        }
        usual();
}
\gapH {3}
\gapV {3}
\maxHeight {200}
\height {200}
\
\class {slider}
\name {plot.body.ctrl.cam.x}
\parent {plot.body.ctrl.cam}
\script {
        switch (arg[0]) {
        case "_shownPositionV":
                send(TTYObj, "cx", arg[1]);
        break;
        case "expose":
                usual();
                TTYObj = send(parent(), "TTYObjP");
                return;
        break;
        }
        usual();
}
\BGColor {grey45}
\
\class {slider}
\name {plot.body.ctrl.cam.y}
\parent {plot.body.ctrl.cam}
\script {
        switch (arg[0]) {
        case "_shownPositionV":
                send(TTYObj, "cy", arg[1]);
        break;
        case "expose":
                usual();
                TTYObj = send(parent(), "TTYObjP");
                return;
        break;
        }
        usual();
}
\BGColor {grey45}
\
\class {slider}
\name {plot.body.ctrl.cam.z}
\parent {plot.body.ctrl.cam}
\script {
        switch (arg[0]) {
        case "_shownPositionV":
                send(TTYObj, "cz", arg[1]);
        break;
        case "expose":
                usual();
                TTYObj = send(parent(), "TTYObjP");
                return;
        break;
        }
        usual();
}
\BGColor {grey45}
\
\class {txtButton}
\name {plot.body.ctrl.quit}
\parent {plot.body.ctrl}
\label {Quit}
\script {
        switch (arg[0]) {
        case "buttonRelease":
/*              send(send(parent(), "TTYObjP"), "output", "quit");
*/
                quit();
        break;
        }
        usual();
}
\BDColor {white}
\BGColor {grey45}
\gapH {3}
\gapV {3}
\maxHeight {20}
\height {20}
\
\class {txtEdit}
\name {plot.cmd}
\parent {plot}
\script {
        switch (arg[0]) {
        case "buttonPress":
                send(send(parent(), "TTYObjP"), "initClient");
                return;
        break;
        case "keyPress":
                c = key();
                if (c == '\r') {
/*
                        cursorShape("busy");
*/
                        print("issue command:", currentLine(), "\n");
                        send(get("parent"), "search", currentLine());
                        insert('\r');
                        insert(c);

                        if (cursorRow() != 0) {
                                /* a kludgy safe guard to simulate
                                 * carriage-return then ctrl-p
                                 */
                                insert("\r\16\1");
                        }
                } else {
                        insert(c);
                }
                return;
        break;
        case "busy":
        case "idle":
                cursorShape(arg[0]);
                return;
        break;
        }
        usual();
}
\maxHeight {18}
\BGColor {grey45}
\FGColor {white}
\
\class {TTY}
\name {plot.TTY}
\parent {plot}
\path {/home/wei/vplot/vplot}
\script {
/*\path {/home/wei/vplot/vplot}\path {/usr/ucb/yes}\path {/home/wei/simplex/t}
*/
/*      print(": ", arg[0], ", ", arg[1], "\n");*/
/*      print("TTY: ", arg[0], "\n");
*/
        switch (arg[0]) {
```

```
        case "cx":
                output(concat("cx ", arg[1]));
                return;
        break;
        case "cy":
                output(concat("cy ", arg[1]));
                return;
        break;
        case "cz":
                output(concat("cz ", arg[1]));
                return;
        break;
        case "output":
                output(arg[1]);
                return;
        break;
        case "input":
                in = input(0);
/*              print("####### plot.tty input: {", in, "}\n");
*/
/*
                if (in == "BEGIN") {
                        print("plot.tty: sending initialization cmds\n");

                        output("bgcolor LemonChiffon1");
                        output("fgcolor black");

                        w = send(parent(), "canvas_WIDP");
                        width = send(parent(), "canvas_widthP");
                        height = send(parent(), "canvas_heightP");
                        output(concat("w ", w, " ", width, " ", height));
                }
*/
                return;
        break;
        case "init":
                usual();
                print("plot.tty: init\n");
                initialize();
                set("inDelimStr1", "\n");
                set("inDelimStr2", "\r");
                set("outDelimStr", "\n");
                startClient();

                return;
        break;
        case "initClient":
                print("plot.tty: sending initialization cmds\n");

                w = send(parent(), "canvas_WIDP");
                width = send(parent(), "canvas_widthP");
                height = send(parent(), "canvas_heightP");
                output(concat("window ", w, " ", width, " ", height));
                output("domain -2 2 -2 2");
                output("interval .2");
                output("equation 0");
                output("bgcolor LemonChiffon1");
                output("fgcolor yellow");
                output("expose");
                output("expose");

                print("  w = {", w, "}\n");
                return;
        break;
        }
        usual();
}
\height {1}
\maxHeight {1}
\BGColor {LemonChiffon2}
\
\name {doodle}
\class {vpane}
\children {doodle.ctrl doodle.canvas}
\width {400}
\height {400}
\
\class {hpane}
\name {doodle.ctrl}
\parent {doodle}
\children {doodle.ctrl.ops doodle.ctrl.brush doodle.ctrl.color}
\maxHeight {20}
\BGColor {grey45}
\
\class {menu}
\name {doodle.ctrl.ops}
\parent {doodle.ctrl}
\menuConfig {
. {backup}              {send("doodle.canvas", "backup");}
. {redraw               {send("doodle.canvas", "redraw");}
. {record on}           {send("doodle.canvas", "record on");}
. {record off}          {send("doodle.canvas", "record off");}
. {dump}                {send("doodle.canvas", "dump");}
. {clear stroke list}   {send("doodle.canvas", "clear");}
. {clearWindow}         {send("doodle.canvas", "clearWindow");}
. {save}                {send("doodle.canvas", "save");}
. {effect A, x10}       {send("doodle.canvas", "effect A", 10);}
. {effect A, x100}      {send("doodle.canvas", "effect A", 100);}
. {effect A, x200}      {send("doodle.canvas", "effect A", 200);}
. {effect A, x1000}     {send("doodle.canvas", "effect A", 1000);}
. {effect B}            {send("doodle.canvas", "effect B");}
. {effect C}            {send("doodle.canvas", "effect C");}
. {effect D}            {send("doodle.canvas", "effect D");}
. {effect E wave}       {send("doodle.canvas", "effect E");}
}
\label {Operation}
\maxWidth {100}
\FGColor {white}
\BGColor {grey45}
\BDColor {white}
\
\class {menu}
\name {doodle.ctrl.brush}
\parent {doodle.ctrl}
\menuConfig {
. {line}                {send("doodle.canvas", "setBrush", 0);}
. {point}               {send("doodle.canvas", "setBrush", 1);}
. {larger point}        {send("doodle.canvas", "setBrush", 2);}
. {even larger point}   {send("doodle.canvas", "setBrush", 3);}
. {very large point}    {send("doodle.canvas", "setBrush", 4);}
. {monster point}       {send("doodle.canvas", "setBrush", 5);}
. {textual 1}           {send("doodle.canvas", "setBrush", 6);}
. {textual 3}           {send("doodle.canvas", "setBrush", 7);}
}
\label {Brushes}
\maxWidth {100}
\FGColor {white}
\BGColor {grey45}
\BDColor {white}
\
\class {menu}
\name {doodle.ctrl.color}
\parent {doodle.ctrl}
\menuConfig {
. {black}               {send("doodle.canvas", "setColor", "black");}
. {white}               {send("doodle.canvas", "setColor", "white");}
. {red}                 {send("doodle.canvas", "setColor", "red");}
. {blue}                {send("doodle.canvas", "setColor", "blue");}
. {green}               {send("doodle.canvas", "setColor", "green");}
\label {Colors}
\maxWidth {100}
\FGColor {white}
\BGColor {grey45}
\BDColor {white}
\
\class {field}
```

```
\name {doodle.canvas}
\parent {doodle}
\script {
        switch (arg[0]) {
        case "mouseMove":
                if (penDownP) {
                        x0 = x1;
                        y0 = y1;
                        x1 = arg[1];
                        y1 = arg[2];
                        if (brush == 0) {
                                drawLine(x0, y0, x1, y1);
                                if (record == 1) {
                                        stroke[count] = concat("drawLine(",
                                                        x0, ",", y0, ",",
                                                        x1, ",", y1, ");");
                                        count = count + 1;
                                }
                        } else if (brush == 1) {
                                drawLine(x1, y1, x1, y1);
                                if (record == 1) {
                                        stroke[count] = concat("drawLine(",
                                                        x1, ",", y1, ",",
                                                        x1, ",", y1, ");");
                                        count = count + 1;
                                }
                        } else if (brush == 2) {
                                drawFillRect(x1, y1, x1 + 2, y1 + 2);
                                if (record == 1) {
                                        stroke[count] = concat("drawFillRect(",
                                                        x1, ",", y1, ",",
                                                        x1 + 2, ",",
                                                        y1 + 2, ");");
                                        count = count + 1;
                                }
                        } else if (brush == 3) {
                                drawFillRect(x1, y1, x1 + 5, y1 + 5);
                                if (record == 1) {
                                        stroke[count] = concat("drawFillRect(",
                                                        x1, ",", y1, ",",
                                                        x1 + 5, ",",
                                                        y1 + 5, ");");
                                        count = count + 1;
                                }
                        } else if (brush == 4) {
                                drawFillRect(x1, y1, x1 + 10, y1 + 10);
                                if (record == 1) {
                                        stroke[count] = concat("drawFillRect(",
                                                        x1, ",", y1, ",",
                                                        x1 + 10, ",",
                                                        y1 + 10, ");");
                                        count = count + 1;
                                }
                        } else if (brush == 5) {
                                drawFillRect(x1, y1, x1 + 100, y1 + 100);
                                if (record == 1) {
                                        stroke[count] = concat("drawFillRect(",
                                                        x1, ",", y1, ",",
                                                        x1 + 100, ",",
                                                        y1 + 100, ");");
                                        count = count + 1;
                                }
                        } else if (brush == 6) {
                                c = nthChar(brushText, idx);
                                drawText(x1, y1, 1, c);
                                idx = idx + 1;
                                if (record == 1) {
                                        stroke[count] = concat("drawText(",
                                                        x1, ",", y1, ",",
                                                        "1, ", c, ");");
                                        count = count + 1;
                                }
                        } else if (brush == 7) {
                                c = nthChar(brushText, idx);
                                drawText(x1, y1, 3, c);
```

```
                                        idx = idx + 1;
                                        if (record == 1) {
                                                stroke[count] = concat("drawText(",
                                                        x1, ",", y1, ",",
                                                        "3, ", c, ");");
                                                count = count + 1;
                                        }
                                }
                                return;
                        }
                break;
                case "buttonPress":
                        penDownP = 1;
                        x0 = mouseX();
                        y0 = mouseY();
                        x1 = x0;
                        y1 = y0;
                break;
                case "buttonRelease":
                        penDownP = 0;
                break;
                case "setColor":
                        set("FGColor", arg[1]);
                break;
                case "setBrush":
                        brush = arg[1];
                        if (brush == 6) {
                                idx = 0;
/*
                                brushText = loadFile("~/aliceInWonderland.txt");
                                brushText = loadFile("/home/wei/viola/apps/doodle.v");
*/
                                brushText = loadFile("/usr/dict/words");
                        } else if (brush == 7) {
                                idx = 0;
                                brushText = loadFile("/usr/dict/words");
                        }
                break;
                case "clear":
                        count = 0;
                        clearWindow();
                        return;
                break;
                case "clearWindow":
                        clearWindow();
                        return;
                break;
                case "save":
                        tt = "";
                        for (i = 0; i < count; i = i + 1) {
                                tt = concat(tt, stroke[i], "\n");
                        }
                        print("Saving to doodle.out\n");
                        saveFile("/usr/tmp/doodle.out", tt);
                        return;
                break;
                case "dump":
                        for (i = 0; i < count; i = i + 1) {
                                print(i, "\t", stroke[i], "\n");
                        }
                        return;
                break;
                case "backup":
                        count = count - 1;
                        return;
                break;
                case "redraw":
                        tt = "";
                        for (i = 0; i < count; i = i + 1) {
                                tt = concat(tt, stroke[i]);
                        }
                        clearWindow();
                        interpret(tt);
                        return;
                break;
                case "record on":
```

```
                        record = 1;
                        return;
                break;
                case "record off":
                        record = 0;
                        return;
                break;
                case "effect A":
                        if (count <= 0) return;
                        n = arg[1];
                        for (i = 0; i < n; i = i + 1) {
/*
                                idx = random(1) % count;
                                print("idx=", idx, "\n");
                                print("> ", idx, "\t", stroke[idx], "\n");
                                hue = (random(0) % 50) * 2;
                                set("FGColor", concat(hue, " ",hue, " ",hue));
*/
                                set("FGColor", "grey90");
                                idx = random(1) % count;
                                interpret(stroke[idx]);

                                set("FGColor", "grey80");
                                idx = random(1) % count;
                                interpret(stroke[idx]);

                                set("FGColor", "grey70");
                                idx = random(1) % count;
                                interpret(stroke[idx]);
                        }
                        return;
                break;
                case "effect B":
                        if (count <= 0) return;
                        for (i = 0; i < count;) {
                                set("FGColor", "blue");
                                interpret(stroke[i]);
                                i = i + 1;

                                set("FGColor", "green");
                                interpret(stroke[i]);
                                i = i + 1;
                        }
                        return;
                break;
                case "effect C":
                        if (count <= 0) return;
                        for (i = 0; i < count;) {
                                set("FGColor", "blue");
                                interpret(stroke[i]);
                                i = i + 1;

                                set("FGColor", "grey");
                                interpret(stroke[i]);
                                i = i + 1;
                        }
                        return;
                break;
                case "effect D":
                        if (count <= 0) return;
                        for (i = 0; i < count;) {
                                set("FGColor", "grey80");
                                interpret(stroke[i]);
                                i = i + 1;

                                set("FGColor", "grey70");
                                interpret(stroke[i]);
                                i = i + 1;

                                set("FGColor", "grey60");
                                interpret(stroke[i]);
                                i = i + 1;
                        }
                        return;
                break;
```

```
        case "effect E":
                if (count <= 0) return;
                for (i = 2; i < count;) {
                        set("FGColor", "grey80");
                        interpret(stroke[i + 1]);

                        set("FGColor", "grey50");
                        interpret(stroke[i]);

                        set("FGColor", "grey80");
                        interpret(stroke[i - 1]);

                        set("FGColor", "white");
                        interpret(stroke[i - 2]);
                        i = i + 1;
                }
                return;
        break;
        case "init":
                usual();
                record = 1;
                penDownP = 0;
                return;
        break;
        }
        usual();
}
\
\class {field}
\name {colorwavy}
\parent {}
\script {
        switch (arg[0]) {
        case "graph":
                theta0 = theta0 + .8;
                theta1 = theta1 + .5;

                c0 = c0 + dir;
                if (c0 <= 0) {
                        dir = random(0) / 20000000;
                } else if (c0 > 99) {
                        dir = random(0) / -20000000;
                }
                for (x = x0; x < x1; x = x + 2) {
                        theta0 = theta0 + .1;
                        y = sin(theta0 * f) * r + r;
                        theta1 = theta1 + .1;

                        set("FGColor", concat(c0, ' ', c0, ' ', c0));
                        drawLine(x, y, x, sin(theta1 * f) * r + r);
                }
                if (view == 1) after(100, self(), "graph");
                return;
        break;
        case "randomizeBGColor":
                if (view) {
                        set("BGColor", concatenate(random(0) / 20000000, " ",
                                        random(0) / 20000000, " ",
                                        random(0) / 20000000));
                        clearWindow();
                        after(10000, self(), "randomizeBGColor");
                }
                return;
        break;
        case "VIEW_ON":
                view = 1;
                after(500, self(), "graph");
                after(10000, self(), "randomizeBGColor");
        break;
        case "expose":
                send(send(parent(), "findTop"), "VW_relay", self(),"VIEW_ON");
                send(send(parent(), "findTop"), "VW_relay", self(),"VIEW_OFF");
                view = 1;
                after(500, self(), "graph");
                f = 10.0;
                x0 = 0;
                xx = width();
                x1 = xx;
                r = height() / 2.0;
                after(10000, self(), "randomizeBGColor");
        break;
        case "VIEW_OFF":
                view = 0;
        break;
        case "buttonPress":
                send(self(), "VIEW_ON");
        break;
        case "keyPress":
                send(self(), "VIEW_OFF");
        break;
        }
        usual();
}
\width {100}
\height {100}
\
\class {field}
\name {wave}
\parent {}
\children {wave.sb}
\script {switch (arg[0]) {
case "graph":
/*    degree = float(arg[1]) / 100.0 * 360.0;
      print("degree=", degree, "\n");
*/
      f = float(arg[1]);
      xx = width();
      r  = height() / 2.0;
      theta = 0;
      for (x = 20; x < xx; x = x + 2) {
            theta = theta + .1;
            y = sin(theta * f) * r + r;

/*          print("x=", x, "  y=", y, " \n");
            print("theta=", theta, "  f=", f, " \n");
*/
            drawLine(x, y, x, y + 1);
      }
      return;
break;
}
usual();
}
\width {149}
\height {200}
\BDColor {55 55 55}
\BGColor {blue}
\CRColor {99 99 99}
\FGColor {99 99 99}
\
\class {slider}
\name {wave.sb}
\parent {wave}
\script {
switch (arg[0]) {
case "_shownPositionV":
      usual();
      send(parent(), "graph", arg[1]);
      return;
break;
}
usual();
}
\width {20}
\height {200}
\BDColor {55 55 55}
\BGColor {25 25 25}
\CRColor {99 99 99}
\FGColor {99 99 99}
\direction {t}
\
\name {clock}
\class {vpane}
\parent {}
\width {200}
\height {210}
\children {clock.dial clock.mesg}
\
\name {clock.dial}
\class {XPMBG}
\parent {clock}
\script {
        print("@@@@@@ clock: ");
        for (i =0; i < arg[]; i++) print(arg[i], ", ");
        print("\n");

        switch (arg[0]) {
        case "tick":

                date = date();
                clock.mesg("update");
                second = int(nthWord(date, 6));
                minute = int(nthWord(date, 5));
                hour = int(nthWord(date, 4));
                if (hour >= 12) hour = hour - 12;

                secondD = (second / 60.0 * 360.0) - 90.0;
                minuteD = (minute / 60.0 * 360.0) - 90.0;
                hourD = (hour / 12.0 * 360.0) - 90.0 + (minute / 60.0 * 30.0);

                secondX = secondR * cos(secondD) + centerX;
                secondY = secondR * sin(secondD) + centerY;
                minuteX = minuteR * cos(minuteD) + centerX;
                minuteY = minuteR * sin(minuteD) + centerY;
                hourX = hourR * cos(hourD) + centerX;
                hourY = hourR * sin(hourD) + centerY;

                if (lminuteX != minuteX) {
                        clearWindow();
                        clock.dial("render"); /* brutally redraw */
                        drawLine(centerX, centerY, minuteX, minuteY);
                        drawLine(centerX, centerY, hourX, hourY);
                        invertLine(centerX, centerY, lsecondX, lsecondY);
                }
                invertLine(centerX, centerY, lsecondX, lsecondY);
                invertLine(centerX, centerY, secondX, secondY);

                lsecondX = secondX;
                lsecondY = secondY;
                lminuteX = minuteX;
                lminuteY = minuteY;
                lhourX = hourX;
                lhourY = hourY;
                if (view) after(1000, "clock.dial", "tick");
                return;
        break;
        case "render":
                usual();
                for (i = 1; i <= 12; i = i + 1) {
                        x = letterR * cos((i / 12.0 * 360) - 90) +
                                centerX - 10;
                        y = letterR * sin((i / 12.0 * 360) - 90) +
                                centerY - 5;
                        drawText(x, y, i, str(i));
                }
                return;
        break;
        case "VIEW_ON":
                view = 1;
                return;
        break;
        case "VIEW_OFF":
                view = 0;
                return;
        break;
        case "expose":
```

```
                clearWindow();
                lminuteX = 0; /* forces redrawing */
                lhourX = 0; /* forces redrawing */
        break;
        case "config":
                usual();
                send(self(), "resize", arg[3], arg[4]);
                return;
        break;
        case "resize":
                if (arg[1] < arg[2])
                        radius = arg[1] / 2.0;
                else
                        radius = arg[2] / 2.0;

                centerX = arg[1] / 2.0;
                centerY = arg[2] / 2.0;
                secondR = radius * 0.95;
                minuteR = radius * 0.9;
                hourR   = radius * 0.6;
                letterR = (radius - 9) * 0.94;

                after(2000, "clock.dial", "tick");
                lminuteX = 0;

/*              system(concat(environVar("VIOLA"), "/play ",
                                environVar("VIOLA_DOCS"), "/cuckoo.au"));
*/
        break;
        }
        usual();
}
\label {/* XPM */
static char * marble2.xpm [] = {
"112 118 17 1",
"  c #3F3F3F3F7F7F",
". c #3F3F7F7F3F3F",
"X c #3F3F7F7F7F7F",
"o c #3F3F3F3F3F3F",
"O c #00003F3F7F7F",
"+ c #00007F7F7F7F",
"@ c #00007F7F3F3F",
"# c #7F7F7F7FBFBF",
"$ c #7F7F7F7F7F7F",
"% c #3F3FBFBF7F7F",
"& c #7F7FBFBF7F7F",
"* c #7F7FBFBFBFBF",
"= c #3F3F7F7FBFBF",
"- c #00003F3F3F3F",
"; c #3F3FBFBFBFBF",
": c #7F7F7F7F3F3F",
"? c #7F7F3F3F7F7F",
" .X . . .XoXoXoXo . .Oo+o . .Xo.XXoXO. .XoX .oo . o@oo+oXXXX#$XXX%$X&*$&X&X$XX$
%X&X$XXXX$XXX$X&$X&=&X$%$XX$XXXX$",
"X .Xo+oXoXXo-X .XX-Xo. . . @ . Xoo .oX. XX. -X- -. - @ .o .XXX. .XX$*$*$#X*X$;$
**$%$X&X$%XXX&X#%#X$XXXXX$%XX.XX.",
".o+oXoXoXXoX .X . . @ o@ . .X .Xo+o@ -X.. X. oo.o-oXo . . .-Xo @ XXXX&X&X&X$%$X
&#%$X&XXXXXX$X&X$%XXX$X$XXXXX . .",
"oXoX-XoX. .-Xo.O.X-XoXXoXoXXoXXXo. . @ X. Xo@ o -o-. -. . . @XoX.XX$X$X*$%#X$&
X$X$=$X$X$XXX$XXX$X$%XXX.Xo.X.XXX",
".O.XoXoX XXoXoXoXoXoX.XXXX.XX. .X @oXo.XoXXoXX. .oXo @oo-X-XooXXXXXXX%X$X&X$%X#
X%$X&X$XX$X$%X&X$XXX$XX$XX XoXXoX",
" . .O. @oXo+ .XO. .X-XoX. XoXX. o. .O. XX.O. . Xo--oXoO. o.Oo+ooX.X$XX$%#X*$$%$
$X$XXXXX%XX$XXXXX%XXX$XXX.X.$XoX.",
".OoXoXoXo+XoX-X. . .XXXXX.XXoX .O. X. .XoX.XX@XX . -XoX-.Oo.O.oX. .X.XX$%$%#%$X
*X&XXXX$X.XX.$X$X$XX.XX.XX XXXXX ",
"o@o@ @ -Xo . . . @X . . . : XoX.XoX.XXX.X XoX . .Oo.O.o oX .oO.OXXX.XXXX$&#&#%$
X$XXX.XX. oXXXXXXXX$o. X$X$. . .X",
"- . -XoXoX-X- @ . . XX.XX.XX.XoXX.X XX. +.O. .XXoXo+o .+o+o+oXo. . XoX.$X%$X&X$
%XXX$XXXXXXXXXXX$XX. .X.XXXX.X. .",
".oO.oo+o.O. .oo-oO.o. XoX XoXXXoX .X.XXXoX. .XoXXoXoX-XoXoXoXo+ .X.XX.XX$X&XXXX
$XX.X.X.XXXX:XX$XX. o . X$X. XoXo",
"oO.o+oo+o o- - . . +o+oXo.XXoX.XX. X . .XoX@ .XoX-X. .oX.X.XoX.X . XoXX$X&=$X$X
XXo . X$X$XX X.XXo o@oXXX.X$XX. X",
"o.oOo+o .-Xoo.oO.OooX. .X . X. X. .. X. XXo . -XoXoX.OXo+oXOXoO.oX. XXX$XX$X$%$
```

```
XXX.XXXXXXoXX. oXo-o-o-XXXXX. .Xo",
"- .oXo+oo -oOoo. o.- .X @ .X.XXX X @ .XXoX. -.o. .X-XoX. . .X.XoXXoX.X.XX$X&XX$
$XXoXoXoXXX$o .oo@oo-oXoXX$. Xo+o",
"o- -. .O.-Xo-oOo.OoXoO. .oX-X .X.oXo. . . o.oOoO.O. .XoO. @Xo @XoXXoXX XXXXX$&=
%$XXoX. .oX.Xo-o- oOo-XXX.XXX.XoX",
"O.-X-XooOooo o-o-oo-.o @o-X .X . Xo- .X .X-Xo. . . . -.oXooXX. XXoXXo.X.XX$XXX$
XXX. . . XoX . - .oXooXoX$XXX .XX",
". o-oO.O.oO.. @o . o -ooOo.X .XX. @ -XoXo . -X-o+oXoXoOo+oX-XoX. X.Xo+oXX.XX$%X
XXXoXoXX-X .o@ o.Oo+oXXXXXX?.X . ",
"O.- @o .-oo- -o-o-oo.Oo-.OoX. . +oo@o .X-X-Xo-XoXo-.- . . .XoX-X.X . .X .XXXXX$
X. . .O.X. . o@ -o. @ .XoXoX.X. .",
"o .o @o oOoo@o oo - ooXoo. -X-X. . o@ . .oXoX. -. X . X-XoX@ X.X .XX. . . .X$XX
X. .XoXo o@ o-ooX @ . . X.XoXoX- ",
"@oO.- .-.- o o-.O.o-.OoX- @ooX .Xo+ . .Oo- @ oXoX-.OX. . @XXoX .XoXoXO. . .X.X.
 o. . .Xoo .oXo+o. o@ @ . XXXoXo.",
" .oOooOoO.-o-.o-o oXoo-o.oo+o@Xo-X . .. .oXoX. .O.oXo+oXoXXXX.Xo+o+oXo+oXoX XX
. -X-oOo-o- @o-o -.OoXoXoX. o+oXo",
".Ooo+oo@o o+o oX-. -oX .OoXoXoXoXoXX. . @ . .OXXoXoX-XoX@ XX. Xo+oXoX. . .X.XoX
oXo-. -.o @oOOo .o . . . -o.-o.Oo",
".o-oo- o-o-o .-oXooX-oXoXoX-XoX-XX. XXXoXoX.XoXXoX- . . . . $.Xo . . .X-XX XoXo
X-ooo-o -ooO.o@o- @ @ .X.X .o. ",
"oO.O.-o- . -o .OoOo-oooXoOo. . . . X.XXXoX- X. . @ .-Xo+oX..O. @o-.O.O.Xo. . -.
 . @ .-oo+ooOo - . . .OXoXo+oX-o-",
".o oOoXo-o- @o-oooo -Xo.o. o .XXX.XXoXX. X. .o+oXoXoXoX. XoX-XoXo . . X .O.o -
o+oo- oOoo-o-.- @oO.-XoXo+oXX.X .",
"O.-.o-oOoOoo -X-X-.-ooO. o. @ oXoX . X. X. .O. . @oX-.O. @ . . @ @ .-X.XoX. @o-
XoX .-.o- @ oO.o @oXoX. .X. . X-o",
".OoOoX-.o+o+o.ooo o .oo-Xo .o- .XXoX.XX. X. .+oXoX-X . -. .X-Xo . . . XoXoXoXoX
oX-XoX +oXo+o. @o . Xo+oX .XXo. .",
"oo. @o +ooXo+oO- -.- .OoX.o- .- . XoXoXX.XXXoX-.O. .-oXoXoX-Xo+o-X-X.X.Xo+ooXo+
oX. . . . .oO. oX-X.O.XoX. .XXo+o",
"-X-o @ooX-oO. o.-o o- . .Oo -XoXoXo+oXoX . @ . . -XoXX-X--oXoXoXoXoX-X X .O.oXo
X-Xo+o- @ +oXo+oXo-o. . .OXo. .oX",
"ooX-o @ o+o. . -oX-. -o+o @oXoXo+oX. X.XoXXoX-.O. .Xo+o. . -oX-X-o+oXX. . .OoO.
 . . . .o. . .oX-XoX- @ . . .O.-",
"X-oX-. @oOOo-OoX-o-o-Xoo+o @o- @oX- .X-Xo+oXo+oo @oXoXoO. .XooXoXo. . .X-Xoo. .
Oo. .-o+ . X-X-oX-Xo-. . X.Xo. . ",
"-o+o @o @ . o.-o-o-o-oOoo@o oXo ooXo-XoXoX-Xo.O.- .O.XoXo+o-XoO.o -. .O. .XO.Oo
oX-o- . . @ .oXo @o . .X. X. @ .O",
"@ o@o @oo+o- - -o-o- o.O. o@ @o+.O. .o@oO.oO.Oo .- . @ @o . oXo @ . @ . . . . .
- . .oO. . . .O.o-o+oX- . XXXoXo",
"o@o @o @ o .O.o@ oOo-X-o-X- . .o .-X-Xo .- -o. @oX-. . . . @ .-XoXoXoXoXo+. . -
o+o@ -. -XoXoXoO.Oo-Xo.-X. .oX.X.",
" ooXoOoo@ @oXo o-XoXo-XoXoo@oOo+oXo . @oO.o.O.- o-XoX-XoX-. . .oO.ooXoX-XoXo+.X
oXooXo . . .O. . o-XoX-oX .XX. Xo",
".O.- .- .- o+o@Xo@ @ .o-oO. -. @oO. oXooX- o+o.O.o+ooXoXo . .O. o @ .oXoXoXo .
O.O.O. @o-XoXo.O.Oo.O.XooX.o .X. ",
" -oX-oX- .-Xo .- oXo.OoX-oo-Xo-oXo-X.-o+ooXo-oo-oOooX-XoX-X-XoXoXo. o@ @ .X.XoX
oXooXoX X. .X. .o.-Xo .X-X .+oXXX",
"@oXo@ -.oOoo+o .-X- o o-. . -oX-o-oo XooOoo@ .OoXo@ oXo+o . oX- .Oo. .o .Oo @ -
X-o+oX.XoXXoX-X .OXoX-XoX.oX.Xo. ",
"o -oo. -XoX-o.O. oo@ @oXooOo.Oo-oOo@o-oXo@ oo-.o-o @o+ooXo.-X-o-o. oXoO. .XoXo.
oXooX .XX.O.XoXoXo. .o+oXoX . X.X",
".-.OoO.o- -o+o-Xo+o . o-Xoo-oo-Xo.- o @oo oo o oO-. o. @oO. -o-o oXo.-Xo+oo+o -
X-X-.X-X XXoXX. .+oOoXo.O.oX.Xo+o",
" oXo@oOo.-o o-. @o+o-o+ooO.OoOo-oOo.OooOo-o.-Xo@oo-.Ooo .o-ooo-o@ .o -oXoX. .o@
 ooXoXoX..XX. . . .X-oXoXXoX .XoX",
"@o- oX- - @oOoo- oo+o o-.- o@oo- o-o.O.o-Xo o-Xo -o @oOo- -o -o -o- @XoO.Oo@ Xo
-.O.O.XoO. .XXoX.X .oXoX. X.Xo+o.",
". .-X-o.- oo.O. @ .oOo- o @o @ .-oX- o-oo-o-Xoo .-o- .-o -o.oO-oo-oo -XooXoXo-X
oO.oXoX. XoXX.X @ . XoXXX. @ .X .",
" - .- oO.- - o-o.Oo+o.o-.- -Xo- .-o @o@ - .oo .-ooO.o- -o-o-oo-oXo @oXo+o+o-XoO
.oO. - . .O. X . XX. X@oXXXoX.oX-",
".-. o@ ooO.o-. -oo.o - oOoo.oO.oOo@o -oo-oOo @oo- o-o o-o-o-Oo- @o-oX-XoX-oXo@o
XoXo@Xo+. . .X.X. .X. XXoXoX. .oo",
"oX-o+o-X-. - -ooOoOo@ -.-oO-Xo @o oO.o-Xo o.oo -o-o-o-.oOo- o@o oXoX-oX-oo+oXoO
o+o- oXo . .O. o .X X. .XXXo o- .",
"-oX-oo-.oOoo.o-oo@o -.oOo-oo @o @o@o- o- @o- -ooOo- o+o-. .-o .@ -. .O. @ o-oXo
X-oXo@o-X-XoXoXX. .XX.XXXoX.o@o- ",
"X-o-X- o-.oOoo- - .-Xooo@ -.- .o - o-.O.ooOo-.-oo-o@oo oOoO.-o -.OoX-. oo-. @o-
oX-o+oXoXo+oX-XooXoX $oXXX oo .oo",
".Oo.-o-XoOooO. o.-oo - -oo -XoX-X.o+o o- o-.o- -o- oX-@ .oo -.O. . @ o@ -X-o @
.O.oOOo+oXoXo. X. XoX.XXoX.-.oOo+",
" .- o -o-o-o- -oo-oOo.o-X-ooX-oXoO. -.-o.o- -o-oOo@o- oo- @o o. -.o @oXoooo+o @
 oO.O.oXo-X- -.O. .XoX :Xo o-o.-o",
```

```
"o-o-o@o -. -.o.- oo.-oOo. @ - @oXoX-o-o- o.-oOoo-o- o-. -o @ @ .oO. -o+o+o-. o
.o-.oX-XoXooXoXoX. . .XX .oXoo -o",
"oOooOo-o-o-oOo-o-o - -o-Xo .o.oOo@ ooOoo-o- o@o-o-o-. -oo@o . .Oo.oOo@ o+o .Oo+
o+o - o-X- -.Oo@ XX XX. .oXo.-o. ",
"o@o-o- @oOo-o oO.o-ooo- o@ @ - .oo-X-o--oXo--o - -. -oo+o - Oo-o -o@oo@ o@oO.oo
Xo-. @oXo @oo. o@oX.XXX. oo- oOo-",
"o - -oo-oo---o-o .- @o@o .oO.oOo+o-o-o-o--o--oo@ -oX@ o-oX-oXo@ . oOo @o .o @
-Xo . .O.-oX- @ o+oXoXoX .O.o@oo+",
"-ooO.o-o- - oo-o@oOo-o -o-X-. -.o ooo-o-oo-oo--o -.o- o-oOooOo- -o@o-.- o@ @ .
o o@ -oo . o.oo-Xoo . . .-o o @ o",
" .-. -oOo-o--o -oo-. @o @ ooO. - @- -oo--o- -o-o-. - .-Xo -X- Xo.Oo @ o@ oXooO.
-X- . X-X-oX-XO.o-X-oO.oOoX-.o-o.",
"o-o @o -oX- .-oOoOoX- o@o-X@ .o+oo @o -ooOo-.- -oOo.-oo- -oo -X-o.-o -Xo-X-. .-
 oo@o-XooXO. .o X-o @oO.o-.o-Xo+o",
"O.-. .-.-oo- o@ooo-oo@ oOoooX- o-@o o@oOo-o - .-o@ - -Xo-X-X-.oO.Oo+ooo- o@ -Xo
@ . X@ . .O. . @o .o - -o-oooO",
"oOo-o -ooOoo- o-X-o+oo-.o+o-. .- o @o-o. .-.o@ o -Xo. -Xooo@ oO.oooo@o+oo@ o. -
XoO.-XoXoXXoXoX- @oX- -.o.oO. @ o",
"o@oX-. -o--o.- -o-o- - o+ooO. -o-o-ooOo-- oO oX-oX- -.-o- @oOo.- -X- -oXo .- .
.-XoXoXoXo. .-oXo -o.- -o .oo- .-",
"o -o-o-Xo- - -o@o o.-Xo@o -.o-Xo+ooOo-oo-o-.-X-X-oX-Xo -Xo- - o@ o@ .o- @o @o@
 oXoXoXo+oO. .Oo@XoX-ooo@oOoOo-oX",
"@o-o- o-o-o-oo- -.Oo -Xoo@ -Xo-ooOo@oo-oO.- o@ o-X-oo+o.- . - -Xo-Xo- -Xo @oo o
+o@ . . o@o-o+o. o- o- - oo@oo@o-",
"oO.oo@-ooOooX-o-oO.-Xoo+o o.-XoOo@oo- -o-oOo+oo- o.OooOo.- .- o-oX-o.-Xo-Xo-X-.
. -. @o+o . o- @oX.Xo.o-.O. -oO.",
"oo-X-o -X-oX-oXoXo .-Xo @o-X-oXo-o- --o-Xo-oo+oo@oO.o+. -XoO.o+o+oo+o-o+ooX-o o
@ o. -Xo .-o-Xo. - oXoO. o .o-.o ",
"@ o-oOo@oo-o -o+o+oO.o+o -X- .-ooOo-oo-o-oX- o- o+oo+o . -.oOo-.oO.oO. o+o-Xo@
o@ .O. . o+oo@ -. @o .o @oX-X- @o",
"o-.-Xo@ oOoX-Xo-. . . .X.oo+o -X-.o @o-o- -.o@o@oo+oo+.- -Xo.O .oO.o-o+oXo- -o
XoO. . .+o o+ooXoXoX-X-. . . o. o",
".Ooo-. -.Oo+o- oO. .XXo @ . ..o-o o.oO. .oX-oO. -oo+o oX-Xo@ o.- @ o+oO. o+o. @
o+o. . . .+ooXo+oXoOoo ooXoX. @ .",
" .-X- @ o. oXo-.oO.Oo. . -X- +oXo@ - .o- -oo+o o@Oo.O.OoXoXoO.Oo. -. @ o. . -.
oo+o+o@ . o.Oo+oX .-X.O.O. XoX .O",
".Oo@oo.O-.-oOo-o-oo . .O.oXoXoOooXo- @ @o+oo.- oXoXoX. @o-Xo-o@OoX- o.+ @ @ .
@ o. . .-o+o-. .X. Xo -ooXoX-X. X",
"XoX- -X-o o+o.- o-o-.OoXo -.- .o+o-Xo@o.o-o@ O.o+o+oXo-Xo -o+o o.-oXo+ ooXo. o+
o.O. . o . oXoXo +. .Xo+ XXoXXX.X",
"@ .oXo- .-ooOoo-oX- .oX-.oX- .OXoX.-XoX- Xo .- -ooo- @o .-Xoo@o- @o @o. -.OoXo.
O. .Xo+. -Xo-XoX. o+oX.X. .XXoX X",
"o o+o-Xo-X- @oO.Ooo@ .o . o.o@oX. XoX- .oX@oX-. .O. . o@ .o+o @ - .X Xo+o . @ .
 o@ .X X. @ .X. @ . .X XXXXX.XX. ",
". . @ o.oOo. . o.- ooO.O.-XoX . X.XoXo.O. XoXoX @oX.O. . o+o@ .-. . @oO. . . .O
. . . . . . .XoX . .X-XoX. . . XoXX",
". @ oXoX . .O.O. o.O.o -oXo+ooXo. . .XoX.X-Xo+. o+oO. .-Xo. o-X . XX . . @ @ .
.OoXoX-XoX .X.X-XoXoXXoXoX. .XXX ",
"-o. .O. XX. XoXoXO. .Oo. . . .o+ @XoXXoXXXoXoX.XoXoXoXoXo+o+ooXoXo.XoX .XoXoX-
XoX-XoX. .+ X .XX.O.oX-X. .X .X. ",
" .O. . X. XXoXoX. . .Xo+o+o. @ . . Xo+. . .X-X- .Oo. @ .-XoXo-X-X-X .+o+oXXoXoX
o@ . . XoXo. .XXoXoXoX. oXoXoXoXo",
"o o.- .oXXoXXoXX XXXXX. .oXoXoX-X.X. .XX.X XXoX. XX-X .Oo .O. -. o. oXoXoXo+oX-
XoXo+oX-Xo+o+. .XXoX. . .- @o-X .",
"+.OoX- . .Xo . .X. .XX .X @ @ . oX X. . XX.. .X X.XoX. .+o+ooXo @ .+oXoXoXo @ o
Xo+oXoXoX. XoX-XoX. +o+o .- o -XX",
"ooo-oXo+o oXoXX oXoX.XXXX.XXoX . .oXXXXX. -XoX.X.X . +. -XoOo+o@ . oX-XX-X-Xo.
-Xo. . @ XoXo. -oO. . . @o @o@o. ",
"X- .Ooo @o+oXoX.XoXX .X$oXXXXoX@ . . .XXoXo .-X .XX@XoX-XoX. oXo @oX-XoX . - o+
o. X-XoXo.O.O. .XoXoXo+o o-oo . .",
"Oo-oo+o.oOoXoX XoXoX.XXXXXoXoX. XX.X. .XXo+o .o+o. Xo-Xo. -X-X-oXoXo @ . @oX.oX
oX-XoXoXo+oXo @ . @ .Oo. . @o+o -",
"o.o+oo - . .O.oXo @ oXoXoX. .X .XoX .O. o@ @oO.o @o.O. . X. . . @ @ . . @ oO. -
Xo .oX-Xo . .O. -XoXo. .O. oOoX-.",
".OooXoX-oO. . o+oXoXXXXX. X.O.XoXoX-XoX.- .O.o+oX @ . X-X-. -X-XoXo.O.XoXo.oXoX
o-.Oo-oXo+ . .oXXoXoXo+oXXX.o.-o ",
" .oXoo-Xo. XoO. X. . . XX. .X . @ .oX.oXoX-. @ . . @ .XoXXo+ooX . . . XXoX .- -
X .oXX- . .XoX . .o+oXoXooXo Xoo-",
"oXo @ oXoO. . .X XXXX.X. .X XXX. .O.- .XXoXo . @ @ . -XoXoOoX .Xo+oOo+oXX.XX. .
o.OXo-Xo+oXoXX.O.O. . . . .O.oXoX",
"Xo+oo@ o-oXo . ..X.X X X. X.X. .+o-Xo+o-Xo+o+oXoXoXoX. XoX. @XoO. .oo@ .XXX X.X
 oX-XoX. .OX. XoXoXoXo . .O. . -o",
" -oXoo-o -o@ .X XX X.X.XXoX . .OoXo+oX. .Xo+oX-Xo+o+oX. @ .- . .X o+o . .XX. Xo
X+o. . XoXXoXX. .XoXoXo+ . . .-Xo",
" .o-O.Oo-. oXXXXXoXXX . @ @o+ @oX-XoXo+o-XXoX.X . . .OoXoOoX- . .X-o.-o+ . Xo.X
```

```
 . XoXo+oX X. X. . oXoXoXo-X-XoX-",
"- . o-o-oOo+oXXoXXo.oX-oX- - . o+o. .+o. . @ oO.XX-o-XoX-. @ooXXXXX . . .XXo-o
 .XX. .O. .XoX. .O.o .XXoXoooo+oX",
" .-o+o -.- .X .XoX-XoO. -.oXo+o.Oo+ .Xo+. .X@ . . X .OooO. oXX.Xo+oO.-O. . oOo@
 oX X. . XoX .X- o @ -X. .O. @ ",
" . oOo@o -XoX-XoX-XoX- @ oX-XoX- oXXXXoXXo . -XoXo. .+ooXXXo+XXXoX. .oOo+o.o o
 -X.X @ XoXO.oXoXo@o . XoXoXoXo@ .",
" @o.Oo-Xo@ -. . . @ o+oo@ Xo-Xo @ . . -.X@ . . @X-Xo X-Xo+. . .XoXoO.o . -X-X
 ooO. . .+. Xo+oOo -Xo XoX-X-o X",
"X- .-Xo-oOooO. @ -o@ o-Xo.o-Xo-X- .oO.X. X Xo-@ . o- @oXoXoXoXoX .O.X. +o-. o.-
 -oXoXoX o+oXoXXoo oOo oXoXo-o-X ",
" .-XoX-X-o-.XoX-. .Oo@ .OoXoX-Xo .O o oo@ .-Xo -o+oX. .O.O.O. @XoX. -X-oX-o@ O.
 oXoXo+oX-XoXXXoXoOoO. -XoOoXoO. .",
" o--o+oo-X- -Xo+o- oXooX. @ o @oo @o@ o-Xo-.-Xo-XoXoXXoXoX-Xo-X-Xo. .-oXoXo.O
 . @ . . . @ oXXoXo o. o+oX-ooXoX",
".O. -oXoX-oXX. . X-o+o+o+oXo -XoX-XoOo-XoO. o- o-Xo+oX+ .Ooo+o. -X-- X-X-XoO.
 . . - @ ooX.oX .O. -o+ooXoXXoO ",
" .o-Xo-X- .oXo+o+o.O. oo @ . oXoX-o. -o.o --.o@ o+o . .X-.XoOoO.oo @o@ oXoXoX
 -XXoXo- .O.Oo+o@ . -. oXooXo+oXoX",
"@ @ -oO.o@ -X- .o+ . . .X- -XoX.O. o-.oOoOoo-oOo-oo@oO. .X -Xo.oO.O.- oXo@ .O.
 .oXoXo@ o-oXoXo @o -. .oO. oXoXo",
"o@oo.O.- o.Oo.o+ . . .X oXoXoXoX .o+oX-oX-.-o+ooO.O. -.O.O.Xo-+o .oo .-.O. . .
 . .O.- o-.Oo-Xo+oXoXo+o .o-Xo-X- ",
"o X @ oX-X-oX-XoXoX. . .+oO.oXo@o -oOo.Ooo . o+oo-oo- ooXo -Xoo+o- @XoX . .O. @
 @ o . @ o-Xo-XoXXXXX .XoXo-Xo .X",
"O.oXo.Oo-o @ oXoXoX-Xo+oXoX-XX .oXo.Oo. @oO.-o @ .O.-.Oo@ .o+ooXoXoX-X. . . .
 oo+o@ .o-o -oXoXoXX .XoX- oo -X",
"-o+oOo@oX-oXoX-X-o+oX . .X . .oOo@ . @ -o o-Xo-.O.o o-oXo-Xo -Xo+o+oX . X. .-o
 +ooXo-oO. .o+o-X-X.XoX . -Xo . ",
"Xoo@oOoo-X-o+oXoXo .oXX.X .XX . o. .o-X-.-ooOoooX-.O. -o -o+oXo .oX-XoXo+o @
 o+o-XoOo-. o-Xo oOoX .O. . oXoO.X",
"-XoO.o+o-oXoX- @-X-XX. XoXX .Xo@ . X- . -oOoO.o- @o Xo-. o.XoXo+o@ @ . .O. .Xo-
 . . -.o+o @XoX-X.oXoX. Xo . oXXXX",
"X-. -o-XoOo@ .-oXoXoXX.XXoX. X . X. .- .o -oo-o-o- . . -XoO. . oXoXo @oX. . oX
 .O.oX- o.OoXoXo @ @ .O. . .O.XoXo",
" .-Xo+oo+oXo-oXo-X.X . .XXoX.X. . .-.- @o-o-oo-o-o+o+o @ .X +.- @ @oXo+oX.O.o+
 ooOo-Xo+oo+oX-X. . .O. .XOX. XX-.",
".Oo @oOoo@oO.Oo+ooX. . XoXXXXoXX.XoXo .-o -oo+o-o-oXoXo. -XoXoXoXooX- . . X. .
 .Oo@ooX- @o o.Oo+oXoXo+oX. X. . @",
"oo@o o.O.Oo.Xo+o-Xo .oooXoXoXXoo Xo @o- -o--o-o- -oO.oOoXo+o+o+o-X- .-XoX.oXX X
 X. o+oXo. -X- . . @ .XoXoXXX +oX ",
"+o @oOoo-o -Xo-ooO. - -.o . . . .oXooO.o-oo-o-o . . XoX-oO. . oX .o@ oX-X-XoXX.
 Xo@ . +oXoXo+o-XoXoX-X-X. . o+oX",
"o@ooO.-X-XoXoO.O.oX-.o -o-Xo.Xoo -oOoo-oo--oOo+oO.O.Xo@ @oXoXo+o.OXo.O. ooXXX.
 .oXo .-Xo-X-. . .O.XoXo. @ .-XoXo",
"@ -.oOoooO.- o-oo+o o-oX .oXoXX.XoX-.-o-o- o. o-o-ooO. ooXo+ . @ o- -. .XXoXXXo
 X .XoXoX .OoX- @ . .O.O. o-XoXoX-",
"o-oO.o-X-oOo@X.+oo+. @ .oX @ . - -o-o -o -.O.-X-oOo-Xo@ @ X.Xo+o-X.oX .X XXXoXX
 XXXX @ -. . -. .o+oXoXoO.+oXo@ -X",
"Oo@oO.Oo-.oOo oXoXoXoo@ .-XoXoX-.o+o-o--o-o Xo-o-o-oXo .XoXoXoo+ooX .XXXX.XXXoX
 oXX.X .Xo+oXoX-X . .XoXo. .-X .oO",
"oOoo@o-ooO.oX@o+o+oX+ . Xo-oX- @ o.Oo-o- ..O. -o-o-XX. X.Xo+o+o .O.XXXX X.XoXo
 XoX XXX . . @ .XoX. +o+oX- .oX-Xo",
".o@ - -X-X @o . . . . .O.O. @ o- .-o-o-o-oOXo-Xo- @ .O.XoXX-Xoo-XoXX XX.XoX ooo
 - ..XX.XX . .XX.XXX. X.XoXoX-Xo- ",
"O. -oXo-. . o+oXoXXX. -. o.Oo. .-o -o-o-oo. . .O. .oXoX . . -XoXoXXX.X X .oo.-
 o-o-oX$XX. XXX$XX$XXXXXoX.Oo.-X.-",
" -oo@ - . o@ o-X@ .Xo+oX-X-oX-oX .OoXo-o-Xo+oXoXoXo+oO.Xo+o. -o+oXoX .Xo. - -X-
 o-o oX XoXX$$$&#$X$X$XXXo .oOoOoX",
"@o-Oo@ -oXOoo+ooXoX-XoX . .OoX-o@ . @ . XoO. @XXXXo-XoXX. X-XoXXXXX.XoX o@ooooo
 -Xo-. .XXXX$X%#&***$X$. .oO. .-oo",
" .o+ooX-ooX-oX-o+oo-o-Xo@ooXoX-o @oXoXoXo.OoXoXoXo+o-. X. . Xo.o. oX-.oO XX. -
 XoXX .XX$XX.$X$****$XXXoXo@o-oX- ");

}
\BGColor {grey45}
\FGColor {white}
\BDColor {grey20}
\
\name   {clock.mesg}
\parent {clock}
\class  {txtLabel}
\script {
        switch (arg[0]) {
        case "time":
```

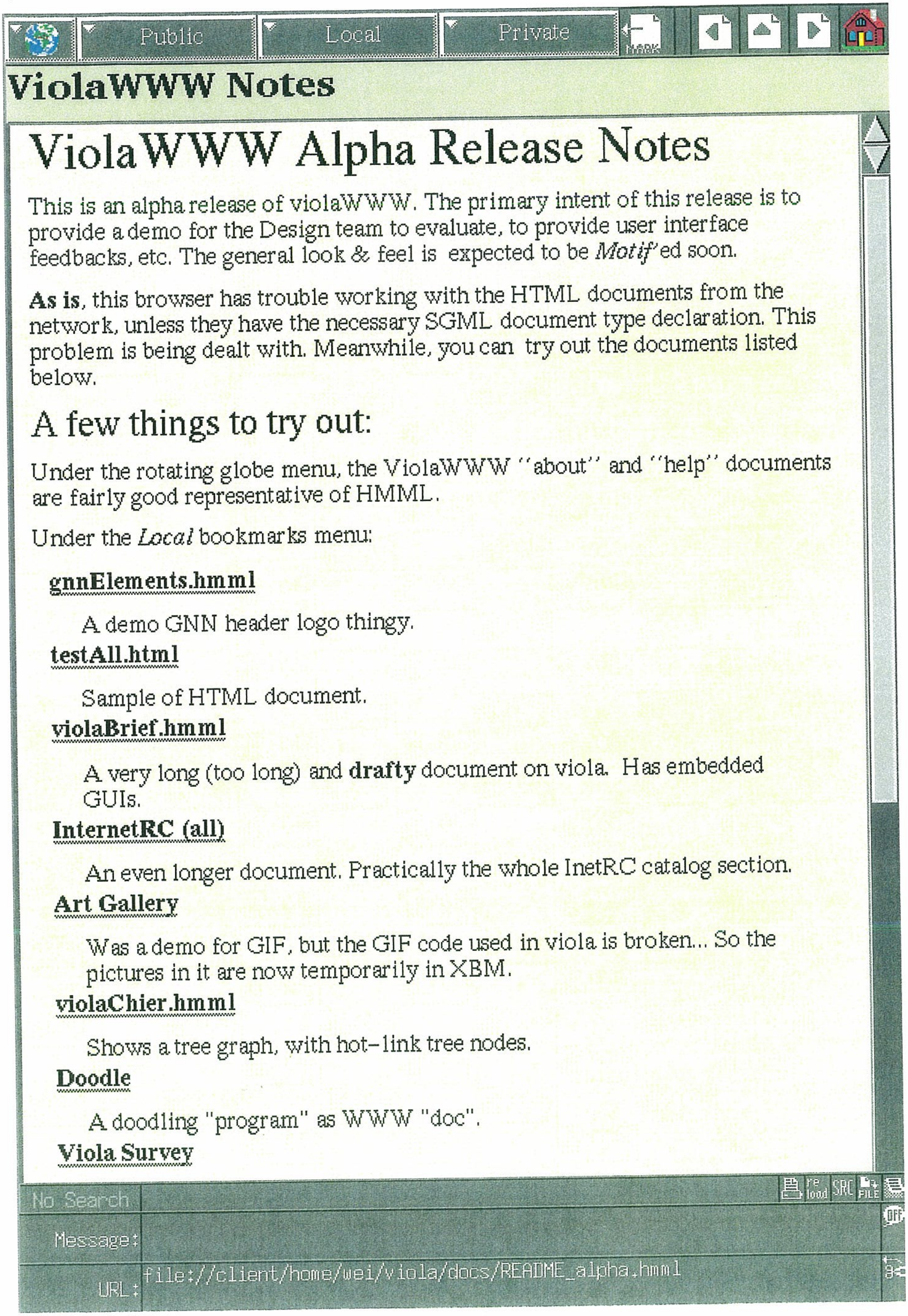
```
                return currentLine();
        break;
        case "update":
                set("label", date());
                render();
                return;
        break;
        }
        usual();
}
\label {Tue Feb 18 16:55:16 1992}
\BGColor {grey30}
\FGColor {lightyellow}
\BDColor {grey20}
\font {usual_small}
\maxHeight {17}
\
\class {vpane}
\name {vwq}
\children {vwq.about vwq.email vwq.name vwq.org vwq.listP vwq.blank vwq.send}
\width {400}
\height {400}
\script {
        switch (arg[0]) {
        case "queryAddress":
                return "a viola app...";
        break;
        case "queryTitle":
                return "a viola app title...";
        break;
        case "queryIsIndex":
                return 0;
        break;
        case "expose":
                bell(2);
                send(self(), "config", x(), y(), width(), height());
        break;
        }
        usual();
}
\
\class {txtDisp}
\name {vwq.about}
\parent {vwq}
\content {\f(3)ViolaWWW User Survey\f(1)

Please take a few moments to fill out this
form for a users survey. Thanks.
}
\BGColor {LemonChiffon3}
\FGColor {black}
\
\class {hpane}
\name {vwq.email}
\parent {vwq}
\children {vwq.email.title vwq.email.entry}
\maxHeight {20}
\
\class {txtLabel}
\name {vwq.email.title}
\parent {vwq.email}
\label {Your E-Mail address:}
\paneConfig {eastToWest}
\maxWidth {150}
\BGColor {grey45}
\
\class {txtEdit}
\name {vwq.email.entry}
\parent {vwq.email}
\script {
        switch (arg[0]) {
        case "keyPress":
                c = key();
                switch (c) {
                case '\t':
                case '\r':
                        if (shiftKeyP()) vwq.send("focus");
                        else vwq.name.entry("focus");
                        return;
                break;
                default:
                        insert(c);
                        return;
                break;
                }
        break;
        case "contentP":
                return get("content");
        break;
        }
        usual();
}
```

```
\BGColor {LemonChiffon1}
\FGColor {black}
\CRColor {red}
\
\class {hpane}
\name {vwq.name}
\parent {vwq}
\children {vwq.name.title vwq.name.entry}
\maxHeight {20}
\
\class {txtEdit}
\name {vwq.name.entry}
\parent {vwq.name}
\script {
        switch (arg[0]) {
        case "keyPress":
                c = key();
                switch (c) {
                case '\t':
                case '\r':
                        if (shiftKeyP()) vwq.email.entry("focus");
                        else vwq.org.entry("focus");
                        return;
                break;
                default:
                        insert(c);
                        return;
                break;
                }
        break;
        case "contentP":
                return get("content");
        break;
        case "init":
/*bugged                set("content", pipe("whoami"));
*/
                tmpWhoAmI = makeTempFile();
                system(concat("whoami > ", tmpWhoAmI));
                set("content", trimEdge(loadFile(tmpWhoAmI)));
                system(concat("rm -f ", tmpWhoAmI));
/*
                set("content", pipe("whoami"));
                finfo = nthLine(pipe("finger `whoami`"), 1, 1);
                cidx = findPattern(finfo, "In real life: ");
                if (cidx != -1) {
                        set("content", nthChars(finfo, cidx, 999));
                } else {
                        set("content", "");
                }
*/
        break;
        }
        usual();
}
\BGColor {LemonChiffon1}
\FGColor {black}
\CRColor {red}
\
\class {txtLabel}
\name {vwq.name.title}
\parent {vwq.name}
\label {Your name:}
\paneConfig {eastToWest}
\maxWidth {150}
\BGColor {grey45}
\
\class {hpane}
\name {vwq.org}
\parent {vwq}
\children {vwq.org.title vwq.org.entry}
\maxHeight {100}
\BGColor {grey45}
\
\class {txtLabel}
\name {vwq.org.title}
```

```
\parent {vwq.org}
\label {Your organization:}
\paneConfig {eastToWest}
\maxWidth {150}
\maxHeight {20}
\BGColor {grey45}
\
\class {txtEdit}
\name {vwq.org.entry}
\parent {vwq.org}
\script {
        switch (arg[0]) {
        case "keyPress":
                c = key();
                if (c == '\t') {
                        if (shiftKeyP()) vwq.name.entry("focus");
                        else vwq.listP.check("focus");
                        return;
                } else {
                        insert(c);
                        return;
                }
        break;
        case "contentP":
                return get("content");
        break;
        }
        usual();
}
\BGColor {LemonChiffon1}
\FGColor {black}
\CRColor {red}
\
\class {hpane}
\name {vwq.listP}
\parent {vwq}
\children {vwq.listP.check vwq.listP.title}
\maxHeight {20}
\
\class {toggle}
\name {vwq.listP.check}
\parent {vwq.listP}
\script {
        switch (arg[0]) {
        case "keyPress":
                c = key();
                if (c == '\t') {
                        if (shiftKeyP()) vwq.org.entry("focus");
                        else vwq.blank("focus");
                        return;
                } else {
                        send(self(), "buttonRelease");
                        return;
                }
        break;
        case "contentP":
                if (get("toggleState") == 1)
                        return "viola-announce@xcf.berkeley.edu";
                else return "";
        break;
        }
        usual();
}
\style {check}
\maxWidth {18}
\BGColor {grey45}
\
\class {txtLabel}
\name {vwq.listP.title}
\parent {vwq.listP}
\label {Check to be added to viola-announce@xcf.berkeley.edu}
\script {
        switch (arg[0]) {
        case "buttonPress":
        case "buttonRelease":
```

```
                        return send("vwq.listP.check", arg[0]);
                break;
                case "keyPress":
                        c = key();
                        if (c == '\t') {
                                if (shiftKeyP()) vwq.org.entry("focus");
                                else vwq.blank("focus");
                                return;
                        } else {
                                vwq.listP.check("buttonRelease");
                                return;
                        }
                break;
                }
                usual();
}
\font {normal}
\BGColor {grey45}
\
\class {txtEdit}
\name {vwq.blank}
\parent {vwq}
\content {Space for any comments you might have.}
\script {
        switch (arg[0]) {
        case "keyPress":
                c = key();
                switch (c) {
                case '\t':
                        if (shiftKeyP()) vwq.listP.check("focus");
                        else vwq.send("focus");
                        return;
                break;
                default:
                        insert(c);
                        return;
                break;
                }
        break;
/*      case "focus":
                mousePos = mouse();
                winPos = windowPosition();
                mx = mousePos[0];
                my = mousePos[1];
                dx = ((winPos[0] + width() / 2) - mx) / 10.0;
                dy = ((winPos[1] + height() / 2) - my) / 10.0;
                for (i = 0; i < 10; i = i + 1) {
                        mx = mx + dx;
                        my = my + dy;
                        setMouse(mx, my);
                }
                return;
        break;
*/
        case "contentP":
                return get("content");
        break;
        }
        usual();
}
\BGColor {LemonChiffon1}
\FGColor {black}
\CRColor {red}
\font {normal}
\
\class {txtButton}
\name {vwq.send}
\parent {vwq}
\label {Press to e-mail this off}
\script {
        switch (arg[0]) {
        case "buttonRelease":

                tmpUserInfo = makeTempFile();
                system(concat("whoami > ",
                                                tmpUserInfo));
                        tmpWhoAmI = makeTempFile();
                        system(concat("who am i > ", tmpWhoAmI));
/*bugged
                        "User Info::: ", pipe("grep `whoami` /etc/passwd"), "\n"
                        "WhoAmI::: ", pipe("who am i"), "\n",
*/

                        tt = concat(
                                "**** Survey extract ***************\n",
                                "User email : ", vwq.email.entry("contentP"), "\n",
                                "User name  : ", vwq.name.entry("contentP"), "\n",
                                "Add to     : ", vwq.listP.check("contentP"), "\n",
                                "Organiz.   : ", vwq.org.entry("contentP"), "\n",
                                "Who_Am_I   : ", trimEdge(loadFile(tmpWhoAmI)), "\n",
                                "User Info  : ", trimEdge(loadFile(tmpUserInfo)), "\n",
                                "Comments   : ", vwq.blank("contentP"), "\n",
                                "************************************\n"
                                );
                        print("\n\nIf this survey were for real, the following text woul
d be mailed:\n\n", tt, "\n\n");

                        tmp = makeTempFile();
                        saveFile(tmp, tt);
                        system(concat(MAILER, " ", "-s survey ",
                                        RECEIVER, " < ", tmp));

                        system(concat("rm -f ", tmp,
                                        " ", tmpUserInfo,
                                        " ", tmpWhoAmI
                                        ));
                break;
                case "keyPress":
                        c = key();
                        if (c == '\t') {
                                if (shiftKeyP()) vwq.blank("focus");
                                else vwq.email.entry("focus");
                                return;
                        } else if (c == '\r') {
                                send(self(), "buttonRelease");
                                return;
                        }
                break;
                case "init":
                        RECEIVER = "pei@pebble";
                        MAILER = "Mail";
                break;
                }
                usual();
}
\BGColor (grey45)
\BDColor (white)
\maxHeight (25)
\gapH (2)
\gapV (3)
\
\class {field}
\name {vicon}
\parent {}
\width {66}
\height {34}
\maxWidth {66}
\maxHeight {34}
\children {vicon.starter vicon.bitmap}
\
\class {XPM}
\name {vicon.bitmap}
\parent {vicon}
\width {65}
\height {33}
\gapH {1}
\gapV {1}
\label {/* XPM */
static char * viola [] = {
"64 32 9 1",
"       c white",
".      c gray35",
"X      c gray30",
"o      c #cc6b57",
"O      c black",
"+      c #ecd5ba",
"@      c gray50",
"#      c #9b5948",
"$      c #34b3ba",
"                                                                ",
"                                                               .",
"  XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX. ",
"  XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX .",
"  XXX        XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX. ",
"  XXX       XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX .",
"  XXXX     X XXXXXXooooooXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX. ",
"  XXXXX   X XXXXXoooooooooXXXXXooooXXXXXXXXXXXXXXXXXXXXXXXXXXX .",
"  XXXXXX X XXXXXoooooooooooXXooooooooXXXXXXXXXXXXXXXXXXXXXXXXX. ",
"  XXXXXXX XXXXXoooooOOooOoooooooooooooXXXXXXXXXXXXXXXXXXXXXXXX .",
"  XXXXXXXXXXXXoooooOooOOooooooooooooooXXXXXXXXXXXXXXXXXXXXXXXX. ",
"  XXXXXXXXXXXXoooooooooooooooooooooooooXXXXXXXXXXXXXXXXXXXXXXX .",
"  XXXXXXXXXXXoOOooooooOoooooooooooooooooooXXXXXXXXXXXXXXXXXXXXX. ",
"  XXXXXXXXXXXoOOO+o+o+O+o+o+o+o+o@O@@@O@O@@@O@O@@O@@O@@OOOOXXX .",
"  XXXXXXXXXXXoOOOo+ooO+Ooo+oo+o++OO@O@@OOO@O@O@OO@O@OOOOOOOXXX. ",
"  XXXXXXXXXXXoOOO+o+o+O+o+o+oo+o+@@O@O@@@O@O@@@O@@O@@O@OOOXXXX .",
"  XXXXXXXXXXXooOooooooooooooooooooOOOOOOOOOOOOOOOOOOOOOOOOXXXXX. ",
"  XXXXXXXXXXXXooooooooooooooooooooooooooXXXXXXXXXXXXXXXXXXXXXX .",
"  XXXXXXXXXXXX#ooooOooOOooooooooooooooo#XXXXXXXXXXXXXXXXXXXXXXX. ",
"  XXXXXXXXXXXX#oooooOOooOoooooooooooooooXXX$XXX$X$XX$XX$XXX$XXXX .",
"  XXXXXXXXXXXXX#oooooooooooo##oooooooo#XXX$XXX$X$X$X$XX$X$XXX. ",
"  XXXXXXXXXXXXXX##ooooooo#######ooo##XXXX$XXX$X$X$X$X$XX$X$XXX .",
"  XXXXXXXXXXXXXXX###ooo####XXXX####XXXXXX$XXX$X$X$X$X$XX$X$XXX. ",
"  XXXXXXXXXXXXXXXX########XXXXXXXXXXXXXXX$XXX$X$X$X$X$XX$X$XXX .",
"  XXXXXXXXXXXXXXXXX#####XXXXXXXXXXXXXXXXX$XXX$X$X$X$X$XX$$XXX. ",
"  XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX$X$XX$X$X$X$XX$X$XXX .",
"  XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX$X$XX$X$X$X$XX$X$XXX. ",
"  XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX$XXX$XX$XX$$X$X$XXX .",
"  XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX. ",
"  XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX .",
"  . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .  ",
" . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . ."};
}
\script {
        switch (arg[0]) {
        case "keyPress":
res.dialogWithButtons("show", "fobar fdsf", "OK", "");
        break;
        case "buttonPress":
                vicon.starter(arg[0]);
                return;
        break;
        }
        usual();
}
\
\class {menu}
```

```
\name {vicon.starter}
\parent {vicon}
\script {
        switch (arg[0]) {
        case "about":
                print("Viola is ..\n");
        break;
        }
        usual();
}
\BGColor {grey45}
\FGColor {white}
\BDColor {grey45}
\font {usual_small}
\menuConfig {
.       {World Wide Web Browser},    {www("render");}
.       {The Internet Catalog Resource Catalog},      {twi("render");}
.       {User Survey},                {vwq("render");}
.       {Clock},                      {clock("render");}
}
\content {
.    {Applications}
..      {World Wide Web Browser},    {www("render");}
..      {The Internet Catalog Resource Catalog},      {twi("render");}
..      {User Survey},                {vwq("render");}
..      {Clock},                      {clock("render");}

.    {Tools}
..      {Interpreter Shell},          {shell("render");}
..      {Object Editor},              {edit("render");}
..      {Font Selector},              {fontSelector("render");}
..      {Color Selector},             {colorSelector("render");}
..      {Samples},                    {samples("render");}
..      {Resource},                   {res("render");}

.    {About Viola},                   {vicon.starter("about");}
}
\
```



ViolaWWW Notes

ViolaWWW Alpha Release Notes

This is an alpha release of violaWWW. The primary intent of this release is to provide a demo for the Design team to evaluate, to provide user interface feedbacks, etc. The general look & feel is expected to be *Motif*'ed soon.

As is, this browser has trouble working with the HTML documents from the network, unless they have the necessary SGML document type declaration. This problem is being dealt with. Meanwhile, you can try out the documents listed below.

A few things to try out:

Under the rotating globe menu, the ViolaWWW ''about'' and ''help'' documents are fairly good representative of HMML.

Under the *Local* bookmarks menu:

gnnElements.hmml

A demo GNN header logo thingy.

testAll.html

Sample of HTML document.

violaBrief.hmml

A very long (too long) and **drafty** document on viola. Has embedded GUIs.

InternetRC (all)

An even longer document. Practically the whole InetRC catalog section.

Art Gallery

Was a demo for GIF, but the GIF code used in viola is broken... So the pictures in it are now temporarily in XBM.

violaChier.hmml

Shows a tree graph, with hot-link tree nodes.

Doodle

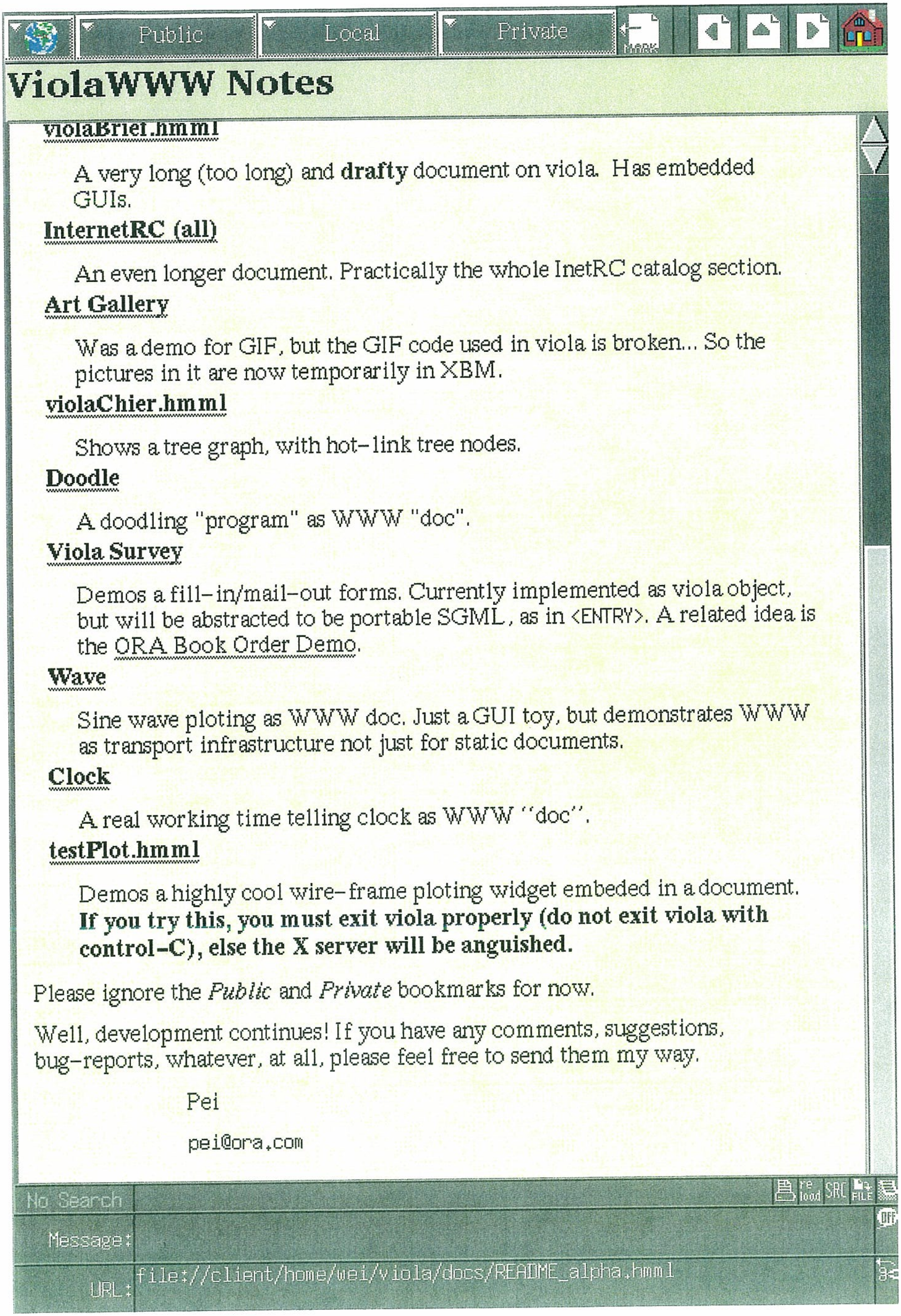
A doodling "program" as WWW "doc".

Viola Survey

No Search

Message:

URL: file://client/home/wei/viola/docs/README_alpha.hmml



ViolaWWW Notes

violaBrief.hmml

A very long (too long) and **drafty** document on viola. Has embedded GUIs.

InternetRC (all)

An even longer document. Practically the whole InetRC catalog section.

Art Gallery

Was a demo for GIF, but the GIF code used in viola is broken... So the pictures in it are now temporarily in XBM.

violaChier.hmml

Shows a tree graph, with hot-link tree nodes.

Doodle

A doodling "program" as WWW "doc".

Viola Survey

Demos a fill-in/mail-out forms. Currently implemented as viola object, but will be abstracted to be portable SGML, as in <ENTRY>. A related idea is the ORA Book Order Demo.

Wave

Sine wave ploting as WWW doc. Just a GUI toy, but demonstrates WWW as transport infrastructure not just for static documents.

Clock

A real working time telling clock as WWW ''doc''.

testPlot.hmml

Demos a highly cool wire-frame ploting widget embeded in a document. **If you try this, you must exit viola properly (do not exit viola with control-C), else the X server will be anguished.**

Please ignore the *Public* and *Private* bookmarks for now.

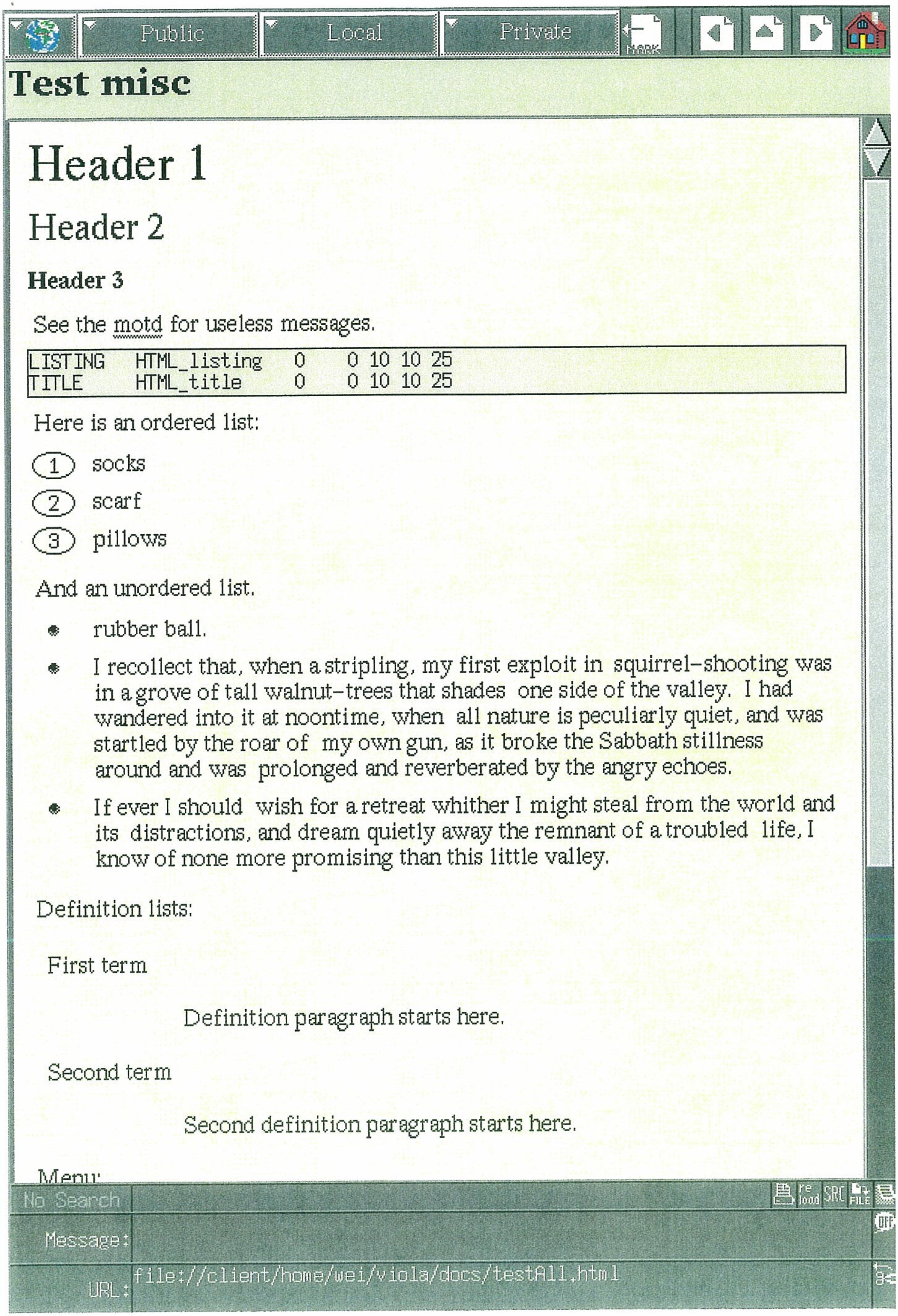
Well, development continues! If you have any comments, suggestions, bug-reports, whatever, at all, please feel free to send them my way.

Pei

pei@ora.com

Message:

URL: file://client/home/wei/viola/docs/README_alpha.hmml



Test misc

Header 1

Header 2

Header 3

See the motd for useless messages.

```
LISTING   HTML_listing   0     0 10 10 25
TITLE     HTML_title     0     0 10 10 25
```

Here is an ordered list:

1. socks
2. scarf
3. pillows

And an unordered list.

- rubber ball.
- I recollect that, when a stripling, my first exploit in squirrel-shooting was in a grove of tall walnut-trees that shades one side of the valley. I had wandered into it at noontime, when all nature is peculiarly quiet, and was startled by the roar of my own gun, as it broke the Sabbath stillness around and was prolonged and reverberated by the angry echoes.
- If ever I should wish for a retreat whither I might steal from the world and its distractions, and dream quietly away the remnant of a troubled life, I know of none more promising than this little valley.

Definition lists:

First term

Definition paragraph starts here.

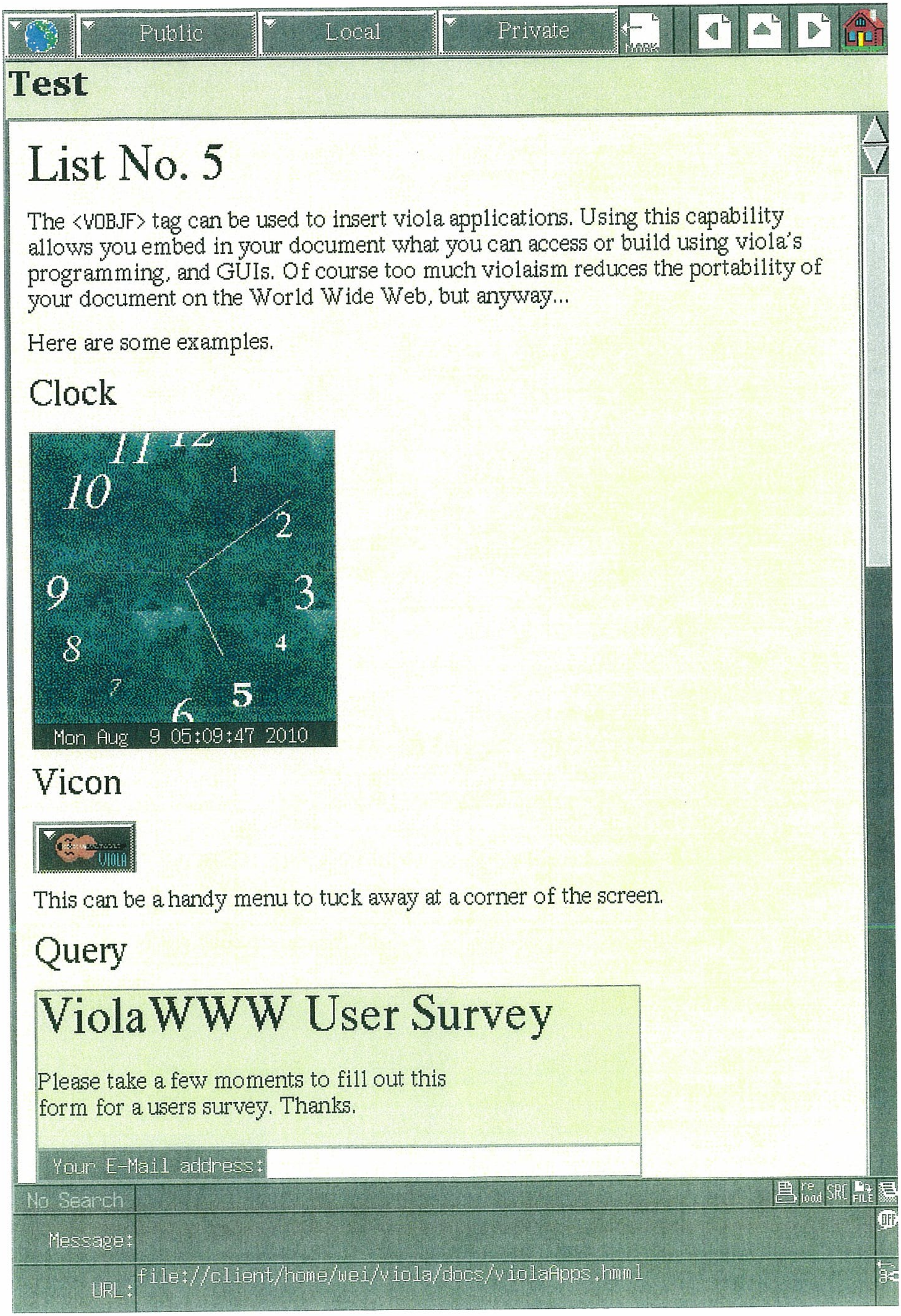
Second term

Second definition paragraph starts here.

Menu:

Message:

URL: file://client/home/wei/viola/docs/testAll.html



Test

List No. 5

The <VOBJF> tag can be used to insert viola applications. Using this capability allows you embed in your document what you can access or build using viola's programming, and GUIs. Of course too much violaism reduces the portability of your document on the World Wide Web, but anyway...

Here are some examples.

Clock

Vicon

This can be a handy menu to tuck away at a corner of the screen.

Query

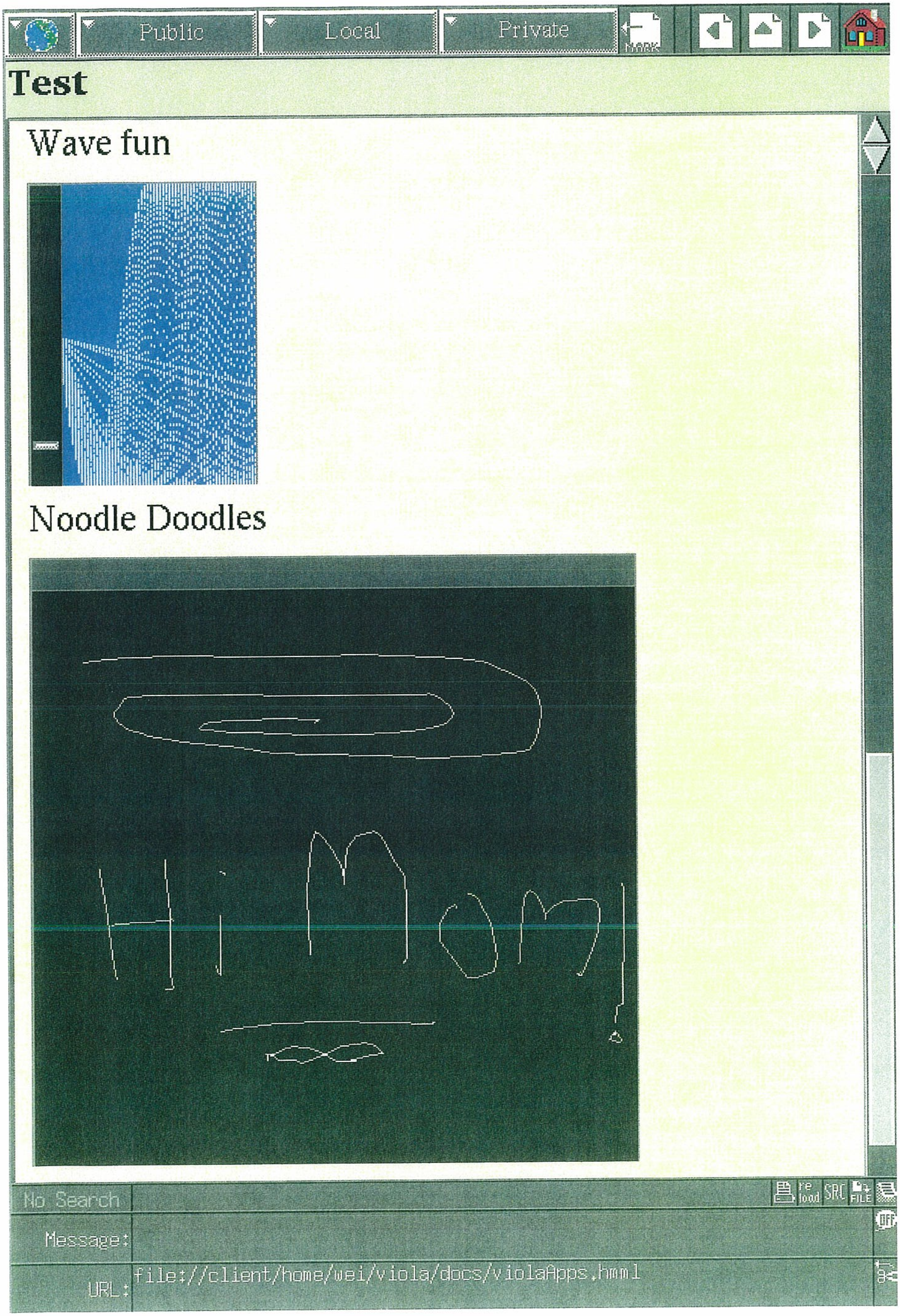
ViolaWWW User Survey

Please take a few moments to fill out this form for a users survey. Thanks.

Your E-Mail address:

Message:

URL: file://client/home/wei/viola/docs/violaApps.hmml



Test

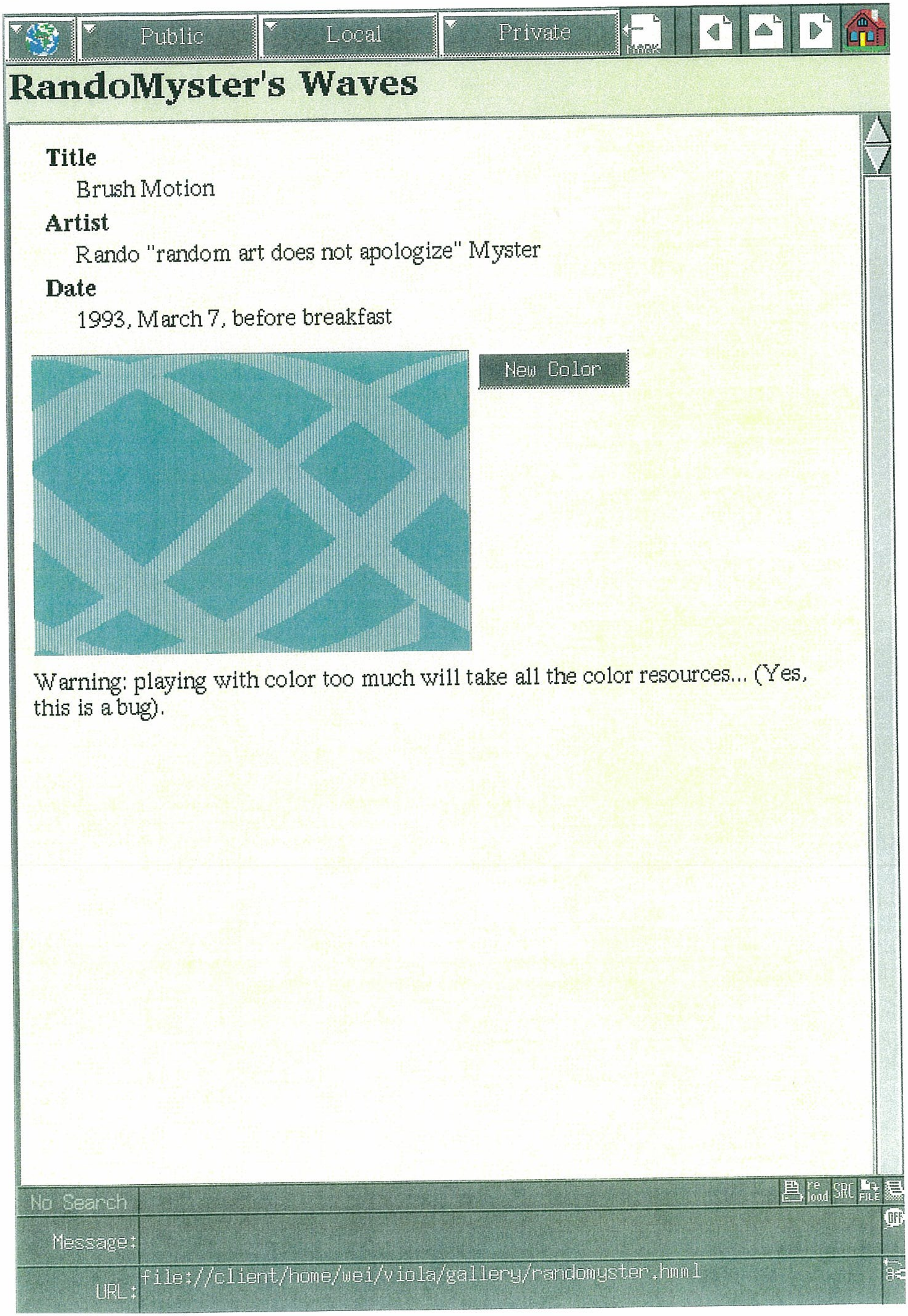
Wave fun

Noodle Doodles

No Search

Message:

URL: file://client/home/wei/viola/docs/violaApps.hmml



RandoMyster's Waves

Title

Brush Motion

Artist

Rando "random art does not apologize" Myster

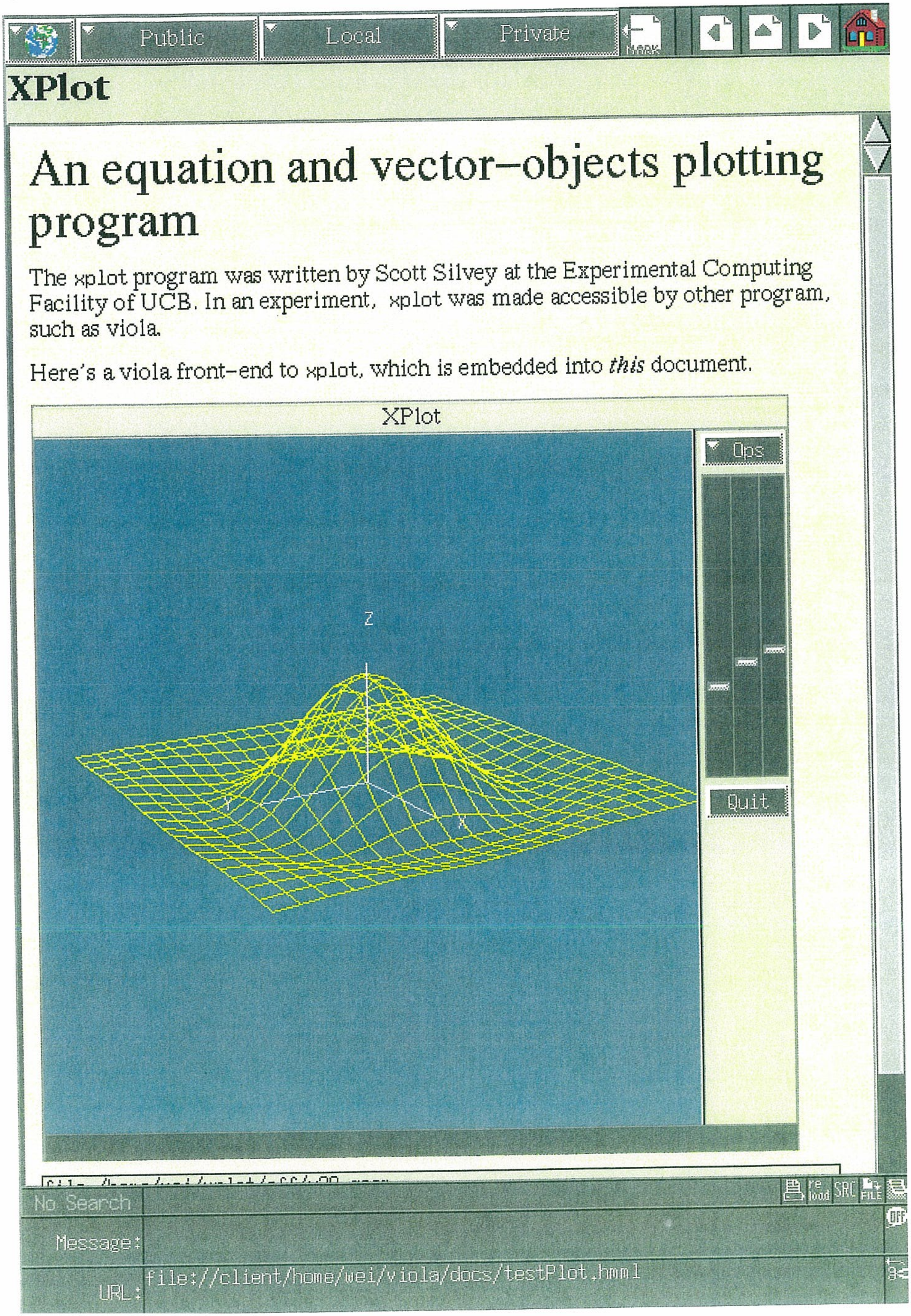
Date

1993, March 7, before breakfast

Warning: playing with color too much will take all the color resources... (Yes, this is a bug).

Message:

URL: file://client/home/wei/viola/gallery/randomyster.hmml



XPlot

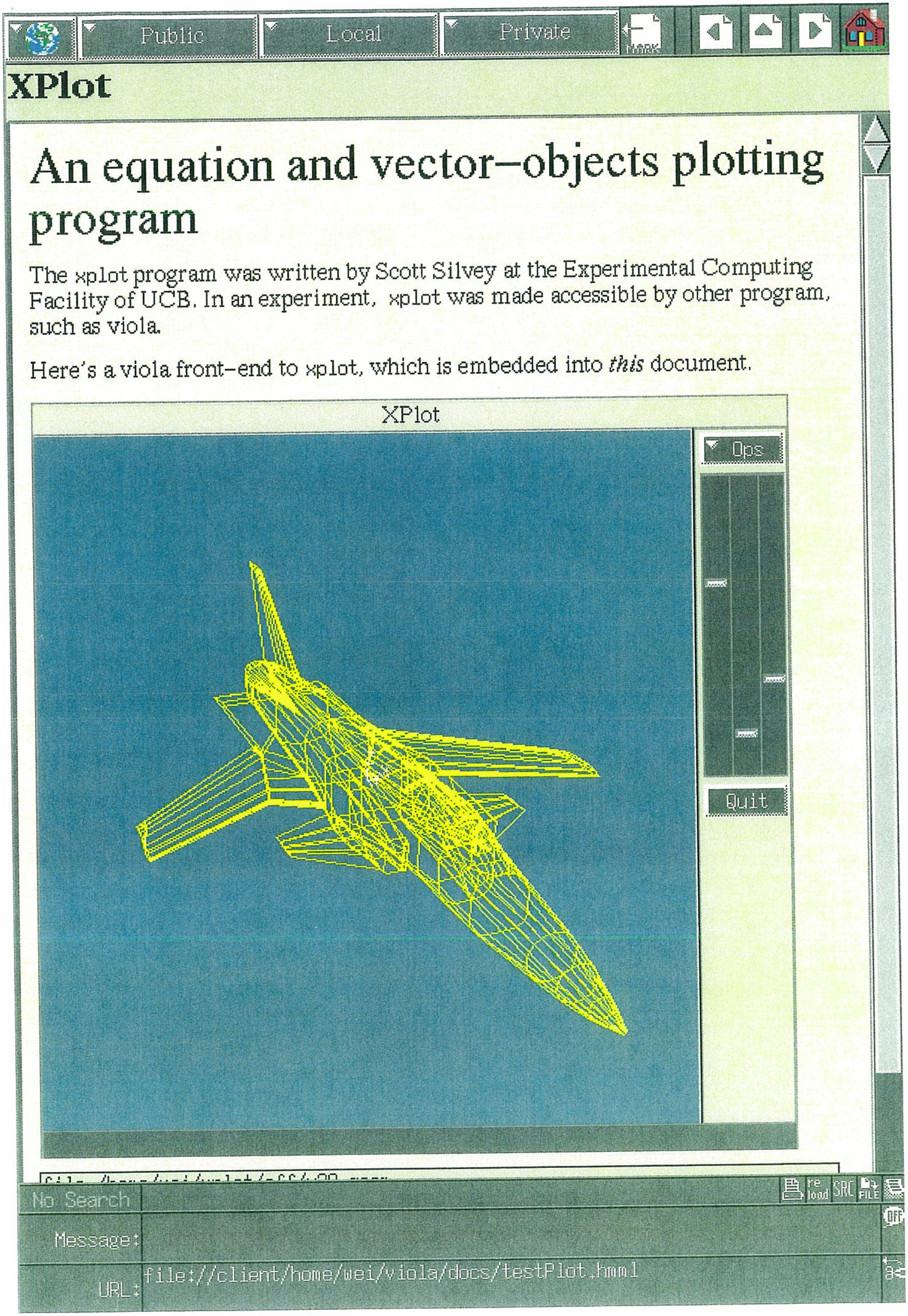
An equation and vector–objects plotting program

The xplot program was written by Scott Silvey at the Experimental Computing Facility of UCB. In an experiment, xplot was made accessible by other program, such as viola.

Here's a viola front–end to xplot, which is embedded into *this* document.

Message:

URL: file://client/home/wei/viola/docs/testPlot.hmml



XPlot

An equation and vector–objects plotting program

The xplot program was written by Scott Silvey at the Experimental Computing Facility of UCB. In an experiment, xplot was made accessible by other program, such as viola.

Here's a viola front-end to xplot, which is embedded into *this* document.

Message:

URL: file://client/home/wei/viola/docs/testPlot.hmml