

# **CLAIM CHART EXHIBIT 2**

## **"VIOLA 5/27/03"**

# **INVALIDITY CLAIM CHART FOR U.S. PATENT NO. 5,838,906**

- BASED ON VIOLA VERSION 5/27/93, DATED MAY 27, 1993 [PA-NAT-78\VIOLA\1993-05-27 DX37 - EX AE TO INV CONTENTIONS\VIOLA\TOGO.TAR.Z]. THE BODY OF MY REPORT HAS A NARRATIVE DESCRIPTION THAT AUGMENTS AND SHOULD BE CONSIDERED PART OF THIS CHART, AND VISE-VERSA FOR THIS AND ALL MY CHARTS.**

| Claim Text from '906 Patent  | Viola 5/27/93  |
|--|--|
| <p><b>906-1.a:</b><br/>A method for running an application program in a computer network environment, comprising:</p>  | <p>Viola 5/27/93 discloses an application program. <i>See, e.g.,</i> :</p> <p>ViolaWWW is a web browser computer program that was implemented using the Viola toolkit/language system. (See, e.g., violaTOGO\apps\www.v.)<br/>By way of example, Viola's documentation states that "ViolaWWW implements an X-Windows browser client to the World-Wide Web system" (See, e.g., violaTOGO\apps\violaWWWAbout.hmml.)</p> <p>Viola 5/27/93 discloses a computer network environment. <i>See, e.g.,</i> :</p> <p>ViolaWWW operates in a computer network environment, including local area networks or the World Wide Web. ViolaWWW operates in computer network environment consisting of clients and servers wherein a server — such as a file server or HTTP server — publishes hypermedia documents such as HMML and HTML files to client workstations. As Viola's documentation notes, "ViolaWWW implements an X-Windows browser client to the World-Wide Web System." (See, e.g., violaTOGO\apps\violaWWWAbout.hmml.)</p> |
| <p><b>906-1.b:</b><br/>providing at least one client workstation and one network server coupled to said network environment, wherein said network environment is a distributed hypermedia environment;</p> | <p>Viola 5/27/93 discloses a client workstation. <i>See, e.g.,</i> :</p> <p>ViolaWWW operated on a client workstation. By way of example, ViolaWWW executed on Sun SPARCstations running a SunOS operating system. This client workstation could interoperate with servers, such as file servers or HTTP servers. Viola's documentation notes that "ViolaWWW implements an X-Windows browser client to the World-</p>  |

| Claim Text from '906 Patent  | Viola 5/27/93   |
|--|---|
|  | <p>Wide Web...” (See, e.g., violaTOGO\apps\violaWWWAbout.hmml.)</p> <p>Viola 5/27/93 discloses a network server. <i>See, e.g., :</i></p> <p>ViolaWWW operated with network servers. Viola's documentation notes that “ViolaWWW implements an X-Windows browser client to the World-Wide Web System” (See, e.g., violaTOGO\apps\violaWWWAbout.hmml.) Servers that ViolaWWW operate with include file servers or HTTP servers. These servers transmit hypermedia documents, such as HMML or HTML files, and a client workstation running ViolaWWW receives them.</p> <p>Viola 5/27/93 discloses a distributed hypermedia environment. <i>See, e.g., :</i></p> <p>ViolaWWW operates in a distributed hypermedia environment, including local area networks or the World Wide Web. ViolaWWW operates in computer network environment consisting of clients and servers wherein a server — such as a file server or HTTP server — publishes hypermedia documents such as HMML and HTML files to client workstations. As Viola's documentation notes, “ViolaWWW implements an X-Windows browser client to the World-Wide Web System.” (See, e.g., violaTOGO\apps\violaWWWAbout.hmml.)</p> |
| <p><b>906-1.c:</b><br/>executing, at said client workstation, a browser application, that parses a first distributed hypermedia document to identify text formats included in said distributed hypermedia document and for responding to predetermined text formats to initiate processing specified by said text formats;</p> | <p>Viola 5/27/93 discloses a browser application. <i>See, e.g., :</i></p> <p>ViolaWWW is a web browser computer program that was implemented using the Viola toolkit/language system. (See, e.g., violaTOGO\apps\www.v.)</p> <p>By way of example, Viola's documentation states that “ViolaWWW implements an X-Windows browser client to the World-Wide Web system” (See, e.g., violaTOGO\apps\violaWWWAbout.hmml.)</p> <p>Viola 5/27/93 discloses that the browser application parses a hypermedia</p>   |

| Claim Text from '906 Patent | Viola 5/27/93   |
|-----------------------------|---|
|                             | <p>document. <i>See, e.g., :</i></p> <p>ViolaWWW running on the client workstation can receive hypermedia document files (i.e., HTML and HMML documents) that contain enabling information from a network server (e.g., a file server or HTTP server) over a distributed hypermedia network environment. Examples of such documents include those in violaTOGO\docs.</p> <p>The hypermedia document files received from the network server contain predetermined text formats which enable a browser application to display at least a portion of a distributed hypermedia document within a browser-controlled window. For example, the hypermedia documents downloaded by ViolaWWW may contain HMML tags or HTML tags. In particular, the testPlot.hmml hypermedia document contains the HMML tags (i.e., text formats) TITLE, H1 and ITALIC. (See, e.g., testPlot.hmml.) As another example, the testAll.html file contains HTML tags, such as TITLE and H1.</p> <p>The hypermedia document downloaded from the remote network server is parsed by ViolaWWW to identify the tags. ViolaWWW then initiates processing specified by the tags. For example, ViolaWWW displays the text marked by the H1 tag in large, bold, header text and the text marked by the ITALIC tag in italics. (See, e.g., apps\VWHandler_hmml.v, src\sgmlsA2B.c and src\sgml.c, violaTOGO\libWWW.)</p> <p>Viola 5/27/93 discloses a hypermedia document with text formats. <i>See, e.g., :</i></p> <p>ViolaWWW running on the client workstation can receive hypermedia document files (i.e., HTML and HMML documents) that contain text formats from a network server (e.g., a file server or HTTP server) over a distributed hypermedia network environment. Examples of such documents include those in violaTOGO\docs.</p> <p>The hypermedia document files received from the network server contain predetermined text formats which enable a browser application to display</p> |

| Claim Text from '906 Patent  | Viola 5/27/93   |
|--|---|
|  | <p>at least a portion of a distributed hypermedia document within a browser-controlled window. For example, the hypermedia documents downloaded by ViolaWWW may contain HMML tags or HTML tags. In particular, the testPlot.hmml hypermedia document contains the HMML tags (i.e., text formats) TITLE, H1 and ITALIC. (See, e.g., testPlot.hmml.) As another example, the testAll.html file contains HTML tags, such as TITLE and H1.</p> <p>The hypermedia document downloaded from the remote network server is parsed by ViolaWWW to identify the tags. ViolaWWW then initiates processing specified by the tags. For example, ViolaWWW displays the text marked by the H1 tag in large, bold, header text and the text marked by the ITALIC tag in italics. (See, e.g., apps\VWHandler_hmml.v, src\sgmlsA2B.c and src\sgml.c, violaTOGO\libWWW.)</p>   |
| <p><b>906-1.d:</b><br/>utilizing said browser to display, on said client workstation, at least a portion of a first hypermedia document received over said network from said server,</p> | <p>Viola 5/27/93 discloses that a hypermedia document is received from the server. <i>See, e.g., :</i></p> <p>A client workstation running the ViolaWWW browser receives hypermedia documents from a server. Examples of servers from which ViolaWWW receives hypermedia documents include file servers or HTTP servers. Examples of documents that ViolaWWW receives can be found in \violaTOGO\docs, and include testPlot.hmml and testAll.html. Code files evidencing ViolaWWW's ability to retrieve documents from servers include apps\VWHandler_hmml.v and src\cl_generic.c.</p> <p>Viola 5/27/93 discloses that the browser displays a hypermedia document. <i>See, e.g., :</i></p> <p>ViolaWWW displays hypermedia documents, including HMML and HTML hypermedia documents. Examples of hypermedia HTML and HMML documents that ViolaWWW displayed — either by retrieving them from a local directory or retrieving them from a server location — are stored in violaTOGO\docs, such as testPlot.hmml and testAll.html.</p> |

| Claim Text from '906 Patent   | Viola 5/27/93  |
|---|--|
|   | <p>ViolaWWW could display other hypermedia documents as well. ViolaWWW displays HTML and HMML documents by parsing them to identify HMML or HTML tags, and then initiates processing specified by the tags and displays the hypermedia document. For example, ViolaWWW will display the text marked by the H1 tag in large, bold, header text and the text marked by the ITALIC tag in italics. Other tags are discussed in more detail elsewhere in this chart.</p>   |
| <p><b>906-1.e:</b><br/>wherein the portion of said first hypermedia document is displayed within a first browser-controlled window on said client workstation,</p>  | <p>Viola 5/27/93 discloses that a hypermedia document is displayed in a browser window. <i>See, e.g., :</i></p> <p>ViolaWWW displays hypermedia documents, including HMML and HTML hypermedia documents, in the ViolaWWW browser window. Examples of hypermedia HTML and HMML documents that ViolaWWW displayed — either by retrieving them from a local directory or retrieving them from a server location — are stored in violaTOGO\docs, such as testPlot.hmml and testAll.html. ViolaWWW could display other hypermedia documents as well.</p> <p>ViolaWWW displays HTML and HMML documents by parsing them to identify HMML or HTML tags, and then initiates processing specified by the tags. The documents are displayed in the ViolaWWW browser window.</p> |
| <p><b>906-1.f:</b><br/>wherein said first distributed hypermedia document includes an embed text format, located at a first location in said first distributed hypermedia document, that specifies the location of at least a portion of an object external to the first distributed hypermedia document,</p> | <p>Viola 5/27/93 discloses an embed text format at a first location in a hypermedia document. <i>See, e.g., :</i></p> <p>ViolaWWW running on the client workstation can receive hypermedia documents (e.g., HTML and HMML documents) from a network server (e.g., a file server or HTTP server) over the distributed hypermedia network environment. The hypermedia document received from the network server contains text formats which enable a browser application to display at least a portion of a distributed hypermedia document within a browser-controlled window. These text formats include embed text formats. For example, an HMML file can include an embed text format</p>  |

| Claim Text from '906 Patent  | Viola 5/27/93   |
|--|---|
|  | <p>called VOBJF. (See, e.g., violaTOGO\docs\testPlot.hmml; docs\violaChier.hmml.) The VOBJF text format is located at a first location in its hypermedia document.</p> <p>Viola 5/27/93 discloses that the embed text format specifies the location of an object. <i>See, e.g., :</i></p> <p>The VOBJF embed text format specifies the location of an object. For example, testPlot.hmml includes a VOBJF tag that shows the tag's syntax, including that it specifies the location of an object based on a filepath location in which the object can be found:<br/> &lt;VOBJF&gt;/usr/work/viola/apps/plot.v&lt;\VOBJF&gt;<br/> (See violaTOGO\docs\testPlot.hmml.)</p> <p>Viola 5/27/93 discloses an object that is external to a hypermedia document. <i>See, e.g., :</i></p> <p>In one example, when a Viola applet is embedded in a ViolaWWW web page, using the VOBJF tag, at least a portion of an object external to a hypermedia document, the default grid, appears in the ViolaWWW window upon browser launch (i.e., violaTOGO\docs\testPlot.hmml). The data for the default grid is specified in the file plot.v by the command:<br/> output("equation 0");<br/> (See apps\plot.v.)</p> |
| <p><b>906-1.g:</b><br/> wherein said object has type information associated with it utilized by said browser to identify and locate an executable application external to the first distributed hypermedia document, and</p> | <p>Viola 5/27/93 discloses that the object has associated type information. <i>See, e.g., :</i></p> <p>For example, the file plot.v contains type information associated with the object.<br/> /path{/usr/work/vplot/vplot}<br/> (See violaTOGO\apps\plot.v.)<br/> The type information is used by the ViolaWWW to identify and locate the</p>  |

| Claim Text from '906 Patent | Viola 5/27/93   |
|-----------------------------|---|
|                             | <pre> vplot executable application. switch (pid = vfork()) { ... case 0: \* Child *\ ... execv(GET_path(self), args); (See src\cl_TTY.c.) </pre> <p>Viola 5/27/93 discloses that the browser uses type information to identify and locate an executable application. <i>See, e.g., :</i></p> <p>For example, the file plot.v contains type information associated with the object.</p> <pre> /path{/usr/work/vplot/vplot} (See violaTOGO\apps\plot.v.) </pre> <p>The type information is used by the ViolaWWW to identify and locate the vplot executable application. ViolaWWW then invokes the executable application.</p> <pre> switch (pid = vfork()) { ... case 0: \* Child *\ ... execv(GET_path(self), args); (See violaTOGO\src\cl_TTY.c.) </pre> <p>Viola 5/27/93 discloses that the executable application is external to the hypermedia document. <i>See, e.g., :</i></p> <p>For example, the vplot executable application is external to the testPlot.hmml hypermedia document. (See violaTOGO\docs\testPlot.hmml) (<i>see, e.g., PA-NAT-00000078\vplot\</i>)</p> |



| Claim Text from '906 Patent   | Viola 5/27/93  |
|---|--|
|   | <p>Viola 5/27/93 also discloses executable applications in the form of scripts, should the Court consider scripts to be covered by its claim construction. <i>See, e.g.,</i></p> <p>For example, the doodle.v script exists as part of the codebase for Viola 5/27/93 (See violaTOGO\apps\doodle.v) (<i>see e.g.,</i> PA-NAT-00000078\viola\)</p>  |
| <p><b>906-1.h:</b><br/>wherein said embed text format is parsed by said browser to automatically invoke said executable application to execute on said client workstation in order to display said object and enable an end-user to directly interact with said object within a display area created at said first location within the portion of said first distributed hypermedia document being displayed in said first browser-controlled window.</p> | <p>Viola 5/27/93 discloses that the browser parses the embed text format. <i>See, e.g.,</i> :</p> <p>The file containing enabling information is downloaded from the remote network server and parsed by ViolaWWW to identify the text formats. ViolaWWW then initiates processing specified by the text formats. For example, ViolaWWW displays the text marked by the H1 tag in large, bold, header text and the text marked by the ITALIC tag in italics. (See, e.g., apps\VWHandler_hmml.v, src\sgmlsA2B.c and src\sgml.c, violaTOGO\libWWW.)</p> <p>For hypermedia documents containing embed text formats, these embed text formats are likewise identified by parsing the file. For example, when parsing testPlot.hmml or violaChier.hmml, violaWWW identifies the VOBJF tag while parsing. (See, e.g., src\sgml.c, apps\HMML_vobjf.v.)</p> <p>Viola 5/27/93 discloses automatic invocation of the executable application. <i>See, e.g.,</i> :</p> <p>When ViolaWWW parses the VOBJF tag, it automatically invokes the vplot executable application. The automatic invocation does not require action by the user.</p> <p>For example, when ViolaWWW parses the VOBJF tag in testPlot.hmml, the vplot application is automatically invoked as follows:</p> <pre>switch (pid = vfork()) {</pre> |

| Claim Text from '906 Patent | Viola 5/27/93   |
|-----------------------------|---|
|                             | <pre> ...     case 0: \* Child *\ ...     execv(GET_path(self), args); (See src\cl_TTY.c.) </pre> <p>As another example, the doodle application is automatically invoked by the ViolaWWW browser upon parsing of the VOBJF tag as follows:</p> <pre> &lt;H2&gt;Noodle Doodles&lt;/H2&gt; &lt;VOBJF&gt;../apps/doodle.v&lt;/VOBJF&gt; </pre> <p>(See violaTOGO\docs\violaApps.hmm1)</p> <p>Viola 5/27/93 discloses that the executable application displays the object. <i>See, e.g., :</i></p> <p>For example, the vplot application displays the object as a grid inside the ViolaWWW window that is displaying testPlot.hmm1. The object is displayed at the location in the window corresponding to the VOBJF tag. (See violaTOGO\docs\testplot.hmm1.</p> <p>As another example, the doodle.v script displays the freehand drawing inside the ViolaWWW window. The drawing is displayed at the location in the window corresponding to the location specified in the script. (See violaTOGO\apps\doodle.v).</p> <p>Viola 5/27/93 discloses that the executable application enables direct interaction with the object. <i>See, e.g., :</i></p> <p>For example, the vplot executable application enables the user to directly interact with the object using the slider bars to rotate the object around the X, Y and Z axes. (See, e.g., violaTOGO\apps\plot.v, src\cl_slider.c,</p> |

| Claim Text from '906 Patent  | Viola 5/27/93   |
|--|---|
|  | <p>src\cl_client.c.)</p> <p>As another example, the doodle.v script enables the user to directly interact with the object using an input device to create a freeform drawing in the display area. (See violaTOGO\apps\doodle.v)</p> <p>Viola 5/27/93 discloses that interaction with the object is at a first location in the hypermedia document. <i>See, e.g., :</i></p> <p>For example, the vplot executable application enables the user to directly interact with the object using the slider bars to rotate the object around the X, Y and Z axes. (See, e.g., apps\plot.v, src\cl_slider.c, src\cl_client.c.) This interaction with the object occurs at the first location within the portion of the HMML document displayed in the ViolaWWW window. (See, e.g., docs\testPlot.hmml, apps\plot.v)</p> <p>Similarly, the doodle.v script enables the freeform drawing interaction described above which occurs at the first location within the portion of the HMML document displayed in the ViolaWWW window. (See violaTOGO\apps\doodle.v)</p> |
| <p><b>906-2.a:</b><br/> The method of claim 1, wherein said executable application is a controllable application and further comprising the step of: interactively controlling said controllable application on said client workstation via inter-process communications between said browser and said controllable application.</p> | <p>Viola 5/27/93 discloses interactive control via inter-process communications between a browser and an application. <i>See, e.g., :</i></p> <p>ViolaWWW browser can communicate with an application using inter-process communication. As noted in Viola's documentation, "... viola is message driven. Messages can originate from window system events and user-GUI interaction, generated by timers, interprocess communication, and objects' scripts." (See, e.g., violaTOGO\docs\testHSEP.hmml.) "Viola is message driven, and messages may be generated by a number of sources. A message is typically caused by the user interacting with a</p>  |

| Claim Text from '906 Patent   | Viola 5/27/93   |
|---|---|
|   | <p>graphical user interface object, but it could also be generated by other objects, or by a timer facility." (See, e.g., violaTOGO\docs\violaBrief.hmml.)</p> <p>"In sum, the Viola language/toolkit system provides an environment where applications are composed of groups of objects, where objects interact, by message passing, with the user and with each other." (See, e.g., docs\violaBrief.hmml.)</p> <p>In one example, the ViolaWWW browser communicates with vplot using inter-process communication. (See, e.g., violaTOGO\apps\plot.v, violaTOGO\src\cl_TTY.c.) (See also main.c of vplot source code, available at, e.g., PA-NAT-00000078\vplot\MS_SUPP_1205_001\petra\vplot.)</p>  |
| <p><b>906-3.a:</b></p> <p>The method of claim 2, wherein the communications to interactively control said controllable application continue to be exchanged between the controllable application and the browser even after the controllable application program has been launched.</p> | <p>Viola 5/27/93 discloses ongoing inter-process communications. <i>See, e.g., :</i></p> <p>ViolaWWW browser can communicate with an application using inter-process communication. As noted in Viola's documentation, "... viola is message driven. Messages can originate from window system events and user-GUI interaction, generated by timers, interprocess communication, and objects' scripts." (See, e.g., violaTOGO\docs\testHSEP.hmml.)</p> <p>"Viola is message driven, and messages may be generated by a number of sources. A message is typically caused by the user interacting with a graphical user interface object, but it could also be generated by other objects, or by a timer facility." (See, e.g., violaTOGO\docs\violaBrief.hmml.)</p> <p>"In sum, the Viola language/toolkit system provides an environment where applications are composed of groups of objects, where objects interact, by message passing, with the user and with each other." (See, e.g., docs\violaBrief.hmml.)</p> <p>In one example, the ViolaWWW browser communicates with vplot using inter-process communication. (See, e.g., violaTOGO\apps\plot.v, violaTOGO\src\cl_TTY.c.) (See also main.c of vplot source code,</p> |

| Claim Text from '906 Patent   | Viola 5/27/93   |
|---|---|
|   | <p>available at, e.g., PA-NAT-00000078\vplot\MS_SUPP_1205_001\petra\vplot.) In this example, an end-user can continue interacting with an object, such as the default grid, in an ongoing basis using the slider bars to rotate the object around the X, Y and Z axes. (See, e.g., viola\apps\plot.v, src\cl_slider.c, src\cl_client.c.) These interactions are communicated between the browser and the vplot executable eapplication on an ongoing basis so that vplot can process those interactions.</p> <p>Similar communications also occur between the browser and the doodle.v script, with interactions between the user and the object being communicated between the script and the browser. (See violaTOGO\apps\doodle.v)</p> |
| <p><b>906-6.a:</b><br/>A computer program product for use in a system having at least one client workstation and one network server coupled to said network environment, wherein said network environment is a distributed hypermedia environment, the computer program product comprising:</p> | <p>Viola 5/27/93 discloses an application program in a computer network environment. <i>See</i> evidence recited for 906-1.a.</p> <p>Viola 5/27/93 also discloses a client workstation and a network server in a distributed hypermedia environment. <i>See</i> evidence recited for 906-1.b.</p>   |
| <p><b>906-6.b:</b><br/>a computer usable medium having computer readable program code physically embodied therein, said computer program product further comprising:</p>  | <p>Viola 5/27/93 discloses computer code physically embodied on a medium. <i>See, e.g., :</i></p> <p>The computer on which ViolaWWW executes includes computer usable media having computer readable program code physically embodied therein. By way of example, ViolaWWW executed on Sun SPARCstations running a SunOS operating system. ViolaWWW could be executed on the Sun SPARCstation with the command “viola -o www.” (See, e.g., violaTOGO\RUN.WWW.) Viola, including the www.v file that implements the ViolaWWW browser, contains computer readable code that is stored on the SPARCstation's computer usable media.</p>  |

| Claim Text from '906 Patent  | Viola 5/27/93  |
|--|--|
| <b>906-6.c:</b><br>computer readable program code for causing said client workstation to execute a browser application to parse a first distributed hypermedia document to identify text formats included in said distributed hypermedia document and to respond to predetermined text formats to initiate processes specified by said text formats; | Viola 5/27/93 discloses a browser application that parses a hypermedia document with text formats and responds to predetermined text formats to initiate processing specified by the text formats. <i>See</i> evidence recited for 906-1.c.  |
| <b>906-6.d:</b><br>computer readable program code for causing said client workstation to utilize said browser to display, on said client workstation, at least a portion of a first hypermedia document received over said network from said server,   | Viola 5/27/93 discloses a hypermedia document received from a server and a browser that displays the hypermedia document. <i>See</i> evidence recited for 906-1.d.   |
| <b>906-6.e:</b><br>wherein the portion of said first hypermedia document is displayed within a first browser-controlled window on said client workstation,   | Viola 5/27/93 discloses that the hypermedia document is displayed in a browser window. <i>See</i> evidence recited for 906-1.e.  |
| <b>906-6.f:</b><br>wherein said first distributed hypermedia document includes an embed text format, located at a first location in said first distributed hypermedia document, that specifies the location of at least a portion of an object external to the first distributed hypermedia document,  | Viola 5/27/93 discloses an embed text format at a first location in a hypermedia document; that the embed text format specifies the location of an object; and that the object is external to the hypermedia document. <i>See</i> evidence recited for 906-1.f.                            |
| <b>906-6.g:</b><br>wherein said object has type information associated with it utilized by said browser to identify and locate an executable application external to the first distributed hypermedia document, and  | Viola 5/27/93 discloses that the object has associated type information, that the browser uses the type information to identify and locate an executable application, and that the executable application is external to the hypermedia document. <i>See</i> evidence recited for 906-1.g. |
| <b>906-6.h:</b>  | Viola 5/27/93 discloses that the browser parses the embed text format; that the  |

| Claim Text from '906 Patent   | Viola 5/27/93   |
|---|---|
| <p>wherein said embed text format is parsed by said browser to automatically invoke said executable application to execute on said client workstation in order to display said object and enable an end-user to directly interact with said object within a display area created at said first location within the portion of said first distributed hypermedia document being displayed in said first browser-controlled window.</p> | <p>browser automatically invokes the executable application; that the executable application displays the object and enables an end-user to directly interact with it; and that interaction with the object is at a first location in the hypermedia document. <i>See</i> evidence recited for 906-1.h.</p> |
| <p><b>906-7.a:</b><br/>The computer program product of claim 6, wherein said executable application is a controllable application and further comprising:<br/>computer readable program code for causing said client workstation to interactively control said controllable application on said client workstation via inter-process communications between said browser and said controllable application.</p>                       | <p>Viola 5/27/93 discloses interactive control via inter-process communications between a browser and an application. <i>See</i> evidence recited for 906-2.a.</p>  |
| <p><b>906-8.a:</b><br/>The computer program product of claim 7, wherein the communications to interactively control said controllable application continue to be exchanged between the controllable application and the browser even after the controllable application program has been launched.</p>  | <p>Viola 5/27/93 discloses ongoing inter-process communications. <i>See</i> evidence recited for 906-3.a.</p>   |
| <p><b>906-11.a:</b><br/>The method of claim 3, wherein additional instructions for controlling said controllable application reside on said network server, wherein</p>   | <p>Viola 5/27/93 discloses additional instructions on the server <i>See, e.g.,</i> :<br/><br/>Viola's documentation states that "Through a communication facility such as the socket, a message may also be generated from another process on</p>   |

| Claim Text from '906 Patent   | Viola 5/27/93  |
|---|--|
| <p>said step of interactively controlling said controllable application includes the following substeps:</p>  | <p>the network." (See, e.g., violaTOGO\docs\violaBrief.hmm1.) By way of example, and as discussed elsewhere in this chart, ViolaWWW provides for automatically invoking the vplot executable application using the execv system call. (See violaTOGO\src\cl_TTY.c.) Vplot can be implemented as a distributed application wherein a portion of the computations — for example, graphics transformations — can be executed on a remote server by using the rexec socket-based system call or other Unix socket-based system calls in lieu of the execv system call. In this case, additional instructions would reside on the server.</p> <p>Furthermore, as discussed in my report and in Viola video 14.avi, and Viola video 15.avi, vplot can be interchanged with other applications, including distributed applications such as VIS. ViolaWWW works with a distributed application in the same way that it works with an executable application on the client workstation.</p>   |
| <p><b>906-11.b:</b><br/>issuing, from the client workstation, one or more commands to the network server;</p> | <p>Viola 5/27/93 discloses that the client issues commands to the server. <i>See, e.g., :</i></p> <p>Viola's documentation states that "Through a communication facility such as the socket, a message may also be generated from another process on the network." (See, e.g., violaTOGO\docs\violaBrief.hmm1.) By way of example, and as discussed elsewhere in this chart, ViolaWWW provides for automatically invoking the vplot executable application using the execv system call. (See violaTOGO\src\cl_TTY.c.) Vplot can be implemented as a distributed application wherein a portion of the computations — for example, graphics transformations — can be executed on a remote server by using the rexec socket-based system call or other Unix socket-based system calls in lieu of the execv system call. In this case, the client would issue commands to the server. These facilities provide for a client issuing commands to a server.</p> <p>Furthermore, as discussed in my report and in Viola video 14.avi, and Viola video 15.avi, vplot can be interchanged with any application,</p> |



| Claim Text from '906 Patent  | Viola 5/27/93  |
|--|--|
|  | including distributed applications such as VIS. ViolaWWW works with a distributed application in the same way that it works with an executable application on the client workstation   |
| <p><b>906-11.c:</b><br/>executing, on the network server, one or more instructions in response to said commands;</p>                               | <p>Viola 5/27/93 discloses that the server executes instructions in response to client commands. <i>See, e.g.,</i> :</p> <p>Viola's documentation states that "Through a communication facility such as the socket, a message may also be generated from another process on the network." (See, e.g., violaTOGO\docs\violaBrief.hmm1.) By way of example, and as discussed elsewhere in this chart, ViolaWWW provides for automatically invoking the vplot executable application using the execv system call. (See violaTOGO\src\cl_TTY.c.) Vplot can be implemented as a distributed application wherein a portion of the computations — for example, graphics transformations — can be executed on a remote server by using the rexec socket-based system call or other Unix socket-based system calls in lieu of the execv system call. In this case, the server would execute in response to commands from the client. These facilities provide for a server executing in response to the comments from the client.</p> <p>Furthermore, as discussed in my report and in Viola video 14.avi, and Viola video 15.avi, vplot can be interchanged with any application, including distributed applications such as VIS. ViolaWWW works with a distributed application in the same way that it works with an executable application on the client workstation</p> |
| <p><b>906-11.d:</b><br/>sending information from said network server to said client workstation in response to said executed instructions; and</p> | <p>Viola 5/27/93 discloses that the server responds with information to the client. <i>See, e.g.,</i> :</p> <p>Viola's documentation states that "Through a communication facility such as the socket, a message may also be generated from another process on the network." (See, e.g., violaTOGO\docs\violaBrief.hmm1.) By way of example, and as discussed elsewhere in this chart, ViolaWWW provides</p>   |

| Claim Text from '906 Patent   | Viola 5/27/93  |
|---|--|
|   | <p>for automatically invoking the vplot executable application using the execv system call. (See violaTOGO\src\cl_TTY.c.) Vplot can be implemented as a distributed application wherein a portion of the computations — for example, graphics transformations — can be executed on a remote server by using the rexec socket-based system call or other Unix socket-based system calls in lieu of the execv system call. In this case, the server would respond with information to the client. These facilities provide for the server responding to the command issued by the client with information to the client.</p> <p>Furthermore, as discussed in my report and in Viola video 14.avi, and Viola video 15.avi, vplot can be interchanged with any application, including distributed applications such as VIS. ViolaWWW works with a distributed application in the same way that it works with an executable application on the client workstation..</p>   |
| <p><b>906-11.e:</b><br/>processing said information at the client workstation to interactively control said controllable application.</p> | <p>Viola 5/27/93 discloses that the client uses information from the server to interactively control the application. <i>See, e.g., :</i></p> <p>Viola's documentation states that "Through a communication facility such as the socket, a message may also be generated from another process on the network." (See, e.g., violaTOGO\docs\violaBrief.hmm1.) By way of example, and as discussed elsewhere in this chart, ViolaWWW provides for automatically invoking the vplot executable application using the execv system call. (See violaTOGO\src\cl_TTY.c.) Vplot can be implemented as a distributed application wherein a portion of the computations — for example, graphics transformations — can be executed on a remote server by using the rexec socket-based system call or other Unix socket-based system calls in lieu of the execv system call. In this case, the client would use information from the server to control the application. These facilities provide for the server responding to the command issued by the client with information to the client. The client can then use that information to control the executable application.</p> |

| Claim Text from '906 Patent   | Viola 5/27/93  |
|---|--|
|   | <p>Furthermore, as discussed in my report and in Viola video 14.avi, and Viola video 15.avi, vplot can be interchanged with any application, including distributed applications such as VIS. ViolaWWW works with a distributed application in the same way that it works with an executable application on the client workstation.</p> |
| <p><b>906-13.a:</b><br/>The computer program product of claim 8, wherein additional instructions for controlling said controllable application reside on said network server, wherein said computer readable program code for causing said client workstation to interactively control said controllable application on said client workstation includes:</p> | <p>Viola 5/27/93 discloses additional instructions on the server <i>See evidence recited for 906-11.a.</i></p>   |
| <p><b>906-13.b:</b><br/>computer readable program code for causing said client workstation to issue from the client workstation, one or more commands to the network server;</p>  | <p>Viola 5/27/93 discloses that the client issues commands to the server. <i>See evidence recited for 906-11.b.</i></p>  |
| <p><b>906-13.c:</b><br/>computer readable program code for causing said network server to execute one or more instructions in response to said commands;</p>  | <p>Viola 5/27/93 discloses that the server executes instructions in response to client commands. <i>See evidence recited for 906-11.c.</i></p>   |
| <p><b>906-13.d:</b><br/>computer readable program code for causing said network sever to send information to said client workstation in response to said executed instructions; and</p>   | <p>Viola 5/27/93 discloses that the server responds with information to the client. <i>See evidence recited for 906-11.d.</i></p>  |
| <p><b>906-13.e:</b><br/>computer readable program code for causing said client workstation to process said information at</p>   | <p>Viola 5/27/93 discloses that the client uses information from the server to interactively control the application. <i>See evidence recited for 906-11.e.</i></p>  |

| Claim Text from '906 Patent  | Viola 5/27/93 |
|--|---------------|
| the client workstation to interactively control said controllable application. |               |
|  |               |

# **INVALIDITY CLAIM CHART FOR U.S. PATENT NO. 7,599,985**

- BASED ON VIOLA 5/27/93, DATED MAY 27, 1993 [PA-NAT-78\VIOLA\1993-05-27 5/27/93 - Ex AE TO INV CONTENTIONS\VIOLA\TOGO.TAR.Z], (“VIOLA 5/27/93”). THE BODY OF MY REPORT HAS A NARRATIVE DESCRIPTION THAT AUGMENTS AND SHOULD BE CONSIDERED PART OF THIS CHART, AND VISE-VERSA FOR THIS AND ALL MY CHARTS.**

| Claim Text from '985 Patent   | Viola 5/27/93   |
|---|---|
| <p><b>985-1.a:</b><br/>A method for running an application program in a distributed hypermedia network environment, wherein the network environment comprises at least one client workstation and one network server coupled to the network environment, the method comprising:</p> | <p>Viola 5/27/93 discloses an application program. <i>See, e.g., :</i></p> <p style="padding-left: 40px;">ViolaWWW is a web browser computer program that was implemented using the Viola toolkit/language system. (See, e.g., violaTOGO\apps\www.v.)<br/>By way of example, Viola's documentation states that “ViolaWWW implements an X-Windows browser client to the World-Wide Web system” (See, e.g., violaTOGO\apps\violaWWWAbout.hmml.)</p> <p>Viola 5/27/93 discloses a distributed hypermedia environment. <i>See, e.g., :</i></p> <p style="padding-left: 40px;">ViolaWWW operates in a distributed hypermedia environment, including local area networks or the World Wide Web. ViolaWWW operates in computer network environment consisting of clients and servers wherein a server — such as a file server or HTTP server — publishes hypermedia documents such as HMML and HTML files to client workstations. As Viola's documentation notes, “ViolaWWW implements an X-Windows browser client to the World-Wide Web System.” (See, e.g., violaTOGO\apps\violaWWWAbout.hmml.)</p> <p>Viola 5/27/93 discloses a client workstation. <i>See, e.g., :</i></p> <p style="padding-left: 40px;">ViolaWWW operated on a client workstation. By way of example, ViolaWWW executed on Sun SPARCstations running a SunOS operating</p> |

| Claim Text from '985 Patent   | Viola 5/27/93   |
|---|---|
|   | <p>system. This client workstation could interoperate with servers, such as file servers or HTTP servers. Viola's documentation notes that "ViolaWWW implements an X-Windows browser client to the World-Wide Web..." (See, e.g., violaTOGO\apps\violaWWWAbout.hmml.)</p> <p>Viola 5/27/93 discloses a network server. <i>See, e.g., :</i></p> <p>ViolaWWW operated with network servers. Viola's documentation notes that "ViolaWWW implements an X-Windows browser client to the World-Wide Web System" (See, e.g., violaTOGO\apps\violaWWWAbout.hmml.) Servers that ViolaWWW operate with include file servers or HTTP servers. These servers transmit hypermedia documents, such as HMML or HTML files, and a client workstation running ViolaWWW receives them.</p> <p>Viola 5/27/93 discloses a distributed hypermedia environment. <i>See, e.g., :</i></p> <p>ViolaWWW operates in a distributed hypermedia environment, including local area networks or the World Wide Web. ViolaWWW operates in computer network environment consisting of clients and servers wherein a server — such as a file server or HTTP server — publishes hypermedia documents such as HMML and HTML files to client workstations. As Viola's documentation notes, "ViolaWWW implements an X-Windows browser client to the World-Wide Web System." (See, e.g., violaTOGO\apps\violaWWWAbout.hmml.)</p> |
| <p><b>985-1.b:</b><br/>receiving, at the client workstation from the network server over the network environment, at least one file containing information to enable a browser application to display at least a portion of a distributed hypermedia document within a browser-controlled window;</p> | <p>Viola 5/27/93 discloses a browser application. <i>See, e.g., :</i></p> <p>ViolaWWW is a web browser computer program that was implemented using the Viola toolkit/language system. (See, e.g., violaTOGO\apps\www.v.)<br/>By way of example, Viola's documentation states that "ViolaWWW implements an X-Windows browser client to the World-Wide Web</p>  |

| Claim Text from '985 Patent | Viola 5/27/93  |
|-----------------------------|--|
|                             | <p>system” (See, e.g., violaTOGO\apps\violaWWWAbout.hmml.)</p> <p>Viola 5/27/93 discloses a file containing enabling information. <i>See, e.g., :</i></p> <p>ViolaWWW running on the client workstation can receive hypermedia document files (i.e., HTML and HMML documents) that contain enabling information from a network server (e.g., a file server or HTTP server) over a distributed hypermedia network environment. Examples of such documents include those in violaTOGO\docs.</p> <p>The hypermedia document files received from the network server contain predetermined text formats which enable a browser application to display at least a portion of a distributed hypermedia document within a browser-controlled window. For example, the hypermedia documents downloaded by ViolaWWW may contain HMML tags or HTML tags. In particular, the testPlot.hmml hypermedia document contains the HMML tags (i.e., text formats) TITLE, H1 and ITALIC. (See, e.g., testPlot.hmml.) As another example, the testAll.html file contains HTML tags, such as TITLE and H1.</p> <p>The hypermedia document downloaded from the remote network server is parsed by ViolaWWW to identify the tags. ViolaWWW then initiates processing specified by the tags. For example, ViolaWWW displays the text marked by the H1 tag in large, bold, header text and the text marked by the ITALIC tag in italics. (See, e.g., apps\VWHandler_hmml.v, src\sgmlsA2B.c and src\sgml.c, violaTOGO\libWWW.)</p> <p>Viola 5/27/93 discloses that the file is received at the client workstation from the network server. <i>See, e.g., :</i></p> <p>A client workstation running the ViolaWWW browser receives hypermedia documents from a server. Examples of servers from which ViolaWWW receives hypermedia documents include file servers or HTTP servers. Examples of files that ViolaWWW receives can be found in</p> |

| Claim Text from '985 Patent | Viola 5/27/93  |
|-----------------------------|--|
|                             | <p data-bbox="951 235 1860 375">\violaTOGO\docs, including testPlot.hmml and testAll.html, and \violaTOGO\apps, such as plot.v. Code files evidencing ViolaWWW's ability to retrieve documents from servers include apps\VWHandler_hmml.v and src\cl_generic.c.</p> <p data-bbox="873 418 1772 488">Viola 5/27/93 discloses that the browser displays at least a portion of a distributed hypermedia document. <i>See, e.g., :</i></p> <p data-bbox="951 527 1892 959">ViolaWWW displays hypermedia documents, including HMML and HTML hypermedia documents. Examples of hypermedia HTML and HMML documents that ViolaWWW displayed — either by retrieving them from a local directory or retrieving them from a server location — are stored in violaTOGO\docs, such as testPlot.hmml and testAll.html. ViolaWWW could display other hypermedia documents as well. ViolaWWW displays HTML and HMML documents by parsing them to identify HMML or HTML tags, and then initiates processing specified by the tags and displays the hypermedia document. For example, ViolaWWW will display the text marked by the H1 tag in large, bold, header text and the text marked by the ITALIC tag in italics. Other tags are discussed in more detail elsewhere in this chart.</p> <p data-bbox="873 1003 1822 1073">Viola 5/27/93 discloses that at least a portion of a hypermedia document is displayed in a browser-controlled window. <i>See, e.g., :</i></p> <p data-bbox="951 1112 1898 1399">ViolaWWW displays hypermedia documents, including HMML and HTML hypermedia documents, in the ViolaWWW browser window. Examples of hypermedia HTML and HMML documents that ViolaWWW displayed — either by retrieving them from a local directory or retrieving them from a server location — are stored in violaTOGO\docs, such as testPlot.hmml and testAll.html. ViolaWWW could display other hypermedia documents as well. ViolaWWW displays HTML and HMML documents by parsing them to</p> |



| Claim Text from '985 Patent   | Viola 5/27/93  |
|---|--|
|   | <p>identify HMML or HTML tags, and then initiates processing specified by the tags. The documents are displayed in the ViolaWWW browser window.</p>  |
| <p><b>985-1.c:</b><br/>executing the browser application on the client workstation, with the browser application:</p> | <p>Viola 5/27/93 discloses a browser application executing on the client workstation. <i>See, e.g., :</i></p> <p>ViolaWWW is a web browser computer program that was implemented using the Viola toolkit/language system. (See, e.g., violaTOGO\apps\www.v.)</p> <p>By way of example, Viola's documentation states that "ViolaWWW implements an X-Windows browser client to the World-Wide Web system" (See, e.g., violaTOGO\apps\violaWWWAbout.hmml.)</p>  |
| <p><b>985-1.d:</b><br/>responding to text formats to initiate processing specified by the text formats;</p>           | <p>Viola 5/27/93 discloses responding to text formats to initiate processing specified by the text formats, i.e., parsing text formats. <i>See, e.g., :</i></p> <p>ViolaWWW running on the client workstation can receive hypermedia document files (i.e., HTML and HMML documents) that contain text formats from a network server (e.g., a file server or HTTP server) over a distributed hypermedia network environment. Examples of such documents include those in violaTOGO\docs.</p> <p>The hypermedia document files received from the network server contain predetermined text formats which enable a browser application to display at least a portion of a distributed hypermedia document within a browser-controlled window. For example, the hypermedia documents downloaded by ViolaWWW may contain HMML tags or HTML tags. In particular, the testPlot.hmml hypermedia document contains the HMML tags (i.e., text formats) TITLE, H1 and ITALIC. (See, e.g., testPlot.hmml.) As another example, the testAll.html file contains HTML tags, such as TITLE and H1.</p> <p>The hypermedia document downloaded from the remote network server is parsed by ViolaWWW to identify the text formats. ViolaWWW then initiates processing specified by the text formats. For example,</p> |

| Claim Text from '985 Patent  | Viola 5/27/93   |
|--|---|
|  | <p>ViolaWWW displays the text marked by the H1 tag in large, bold, header text and the text marked by the ITALIC tag in italics. (See, e.g., apps\VWHandler_hmml.v, src\sgmlsA2B.c and src\sgml.c, violaTOGO\libWWW.)</p>   |
| <p><b>985-1.e:</b><br/>displaying at least a portion of the document within the browser-controlled window;</p> | <p>Viola 5/27/93 discloses that the browser displays a hypermedia document. <i>See, e.g., :</i></p> <p>ViolaWWW displays hypermedia documents, including HMML and HTML hypermedia documents. Examples of hypermedia HTML and HMML documents that ViolaWWW displayed — either by retrieving them from a local directory or retrieving them from a server location — are stored in violaTOGO\docs, such as testPlot.hmml and testAll.html. ViolaWWW could display other hypermedia documents as well. ViolaWWW displays HTML and HMML documents by parsing them to identify HMML or HTML tags, and then initiates processing specified by the tags and displays the hypermedia document. For example, ViolaWWW will display the text marked by the H1 tag in large, bold, header text and the text marked by the ITALIC tag in italics. Other tags are discussed in more detail elsewhere in this chart.</p> <p>Viola 5/27/93 discloses that a hypermedia document is displayed in a browser window. <i>See, e.g., :</i></p> <p>ViolaWWW displays hypermedia documents, including HMML and HTML hypermedia documents, in the ViolaWWW browser window. Examples of hypermedia HTML and HMML documents that ViolaWWW displayed — either by retrieving them from a local directory or retrieving them from a server location — are stored in violaTOGO\docs, such as testPlot.hmml and testAll.html. ViolaWWW could display other hypermedia documents as well. ViolaWWW displays HTML and HMML documents by parsing them to</p> |

| Claim Text from '985 Patent   | Viola 5/27/93   |
|---|---|
|   | <p>identify HMML or HTML tags, and then initiates processing specified by the tags. The documents are displayed in the ViolaWWW browser window.</p>   |
| <p><b>985-1.f:</b><br/> identifying an embed text format which corresponds to a first location in the document, where the embed text format specifies the location of at least a portion of an object external to the file, where the object has type information associated with it;</p> | <p>Viola 5/27/93 discloses identifying an embed text format. <i>See, e.g., :</i></p> <p>The file containing enabling information is downloaded from the remote network server and parsed by ViolaWWW to identify the text formats. ViolaWWW then initiates processing specified by the text formats. For example, ViolaWWW displays the text marked by the H1 tag in large, bold, header text and the text marked by the ITALIC tag in italics. (See, e.g., apps\VWHandler_hmml.v, src\sgmlsA2B.c and src\sgml.c, violaTOGO\libWWW.)</p> <p>For hypermedia documents containing embed text formats, these embed text formats are likewise identified by parsing the file. For example, when parsing testPlot.hmml or violaChier.hmml, violaWWW identifies the VOBJF tag while parsing. (See, e.g., src\sgml.c, apps\HMML_vobjf.v.)</p> <p>Viola 5/27/93 discloses that the embed text format corresponds to a first location in the hypermedia document. <i>See, e.g., :</i></p> <p>ViolaWWW running on the client workstation can receive hypermedia documents (e.g., HTML and HMML documents) from a network server (e.g., a file server or HTTP server) over the distributed hypermedia network environment. The hypermedia document received from the network server contains text formats which enable a browser application to display at least a portion of a distributed hypermedia document within a browser-controlled window. These text formats include embed text formats. For example, an HMML file can include an embed text format called VOBJF. (See, e.g., violaTOGO\docs\testPlot.hmml; docs\violaChier.hmml.) The VOBJF text format corresponds to a first location in the hypermedia document.</p> |

| Claim Text from '985 Patent | Viola 5/27/93   |
|-----------------------------|---|
|                             | <p>Viola 5/27/93 discloses that the embed text format specifies the location of an object. <i>See, e.g., :</i></p> <p style="padding-left: 40px;">The VOBJF embed text format specifies the location of an object. For example, testPlot.hmml includes a VOBJF tag that shows the tag's syntax, including that it specifies the location of an object based on a filepath location in which the object can be found:<br/> &lt;VOBJF&gt;/usr/work/viola/apps/plot.v&lt;\VOBJF&gt;<br/> (See violaTOGO\docs\testPlot.hmml.)</p> <p>Viola 5/27/93 discloses that the object is external to the file containing enabling information. <i>See, e.g., :</i></p> <p style="padding-left: 40px;">In one example, when a Viola applet is embedded in a ViolaWWW web page, using the VOBJF tag, at least a portion of an object external to a file containing enabling information, the default grid, appears in the ViolaWWW window upon browser launch (i.e., violaTOGO\docs\testPlot.hmml).<br/> The data for the default grid is specified in the file plot.v by the command:<br/> output("equation 0");<br/> (See apps\plot.v.)</p> <p>Viola 5/27/93 discloses that the object has associated type information. <i>See, e.g., :</i></p> <p style="padding-left: 40px;">For example, the file plot.v contains type information associated with the object.<br/> /path{/usr/work/vplot/vplot}<br/> (See violaTOGO\apps\plot.v.)<br/> The type information is used by the ViolaWWW to identify and locate the vplot executable application.<br/> switch (pid = vfork()) {</p> |

| Claim Text from '985 Patent  | Viola 5/27/93   |
|--|---|
|  | <pre> ... case 0: \* Child *\ ... execv(GET_path(self), args); (See src\cl_TTY.c.)  As another example, the file doodle.v also contains type information associated with the object.  \script      {     switch (arg[0]) {         ...     }     usual(); }  (See violaTOGO\apps\doodle.v) </pre>   |
| <p><b>985-1.g:</b><br/>utilizing the type information to identify and locate an executable application external to the file; and</p> | <p>Viola 5/27/93 discloses that the browser uses type information to identify and locate an executable application. <i>See, e.g.,</i> :</p> <p>For example, the file plot.v contains type information associated with the object.<br/>/path{/usr/work/vplot/vplot}<br/>(See violaTOGO\apps\plot.v.)<br/>The type information is used by the ViolaWWW to identify and locate the vplot executable application. ViolaWWW then invokes the executable application.</p> <pre> switch (pid = vfork()) { ... case 0: \* Child *\ ... </pre> |

| Claim Text from '985 Patent  | Viola 5/27/93   |
|--|---|
|  | <p>execv(GET_path(self), args);<br/>(See violaTOGO\src\cl_TTY.c.)</p> <p>Viola 5/27/93 discloses that the executable application is external to the file containing enabling information. <i>See, e.g.,</i> :</p> <p>For example, the vplot executable application is external to the testPlot.hmml file containing enabling information. (See violaTOGO\docs\testPlot.hmml), (<i>see, e.g.,</i> PA-NAT-00000078\vplot\).</p> <p>Viola 5/27/93 also discloses executable applications in the form of scripts, should the Court consider scripts to be covered by its claim construction. <i>See, e.g.,</i></p> <p>For example, the doodle.v script exists as part of the codebase for Viola 5/27/93 (See violaTOGO\apps\doodle.v) (<i>see e.g.,</i> PA-NAT-00000078\viola\)</p>   |
| <p><b>985-1.h:</b><br/>automatically invoking the executable application, in response to the identifying of the embed text format, to execute on the client workstation in order to display the object and enable an end-user to directly interact with the object while the object is being displayed within a display area created at the first location within the portion of the hypermedia document being displayed in the browser-controlled window.</p> | <p>Viola 5/27/93 discloses that the browser parses the embed text format. <i>See, e.g.,</i> :</p> <p>The file containing enabling information is downloaded from the remote network server and parsed by ViolaWWW to identify the text formats. ViolaWWW then initiates processing specified by the text formats. For example, ViolaWWW displays the text marked by the H1 tag in large, bold, header text and the text marked by the ITALIC tag in italics. (See, e.g., apps\VWHandler_hmml.v, src\sgmlsA2B.c and src\sgml.c, violaTOGO\libWWW.)</p> <p>For hypermedia documents containing embed text formats, these embed text formats are likewise identified by parsing the file. For example, when parsing testPlot.hmml or violaChier.hmml, violaWWW identifies the VOBJF tag while parsing. (See, e.g., src\sgml.c, apps\HMML_vobjf.v.)</p> |

| Claim Text from '985 Patent | Viola 5/27/93   |
|-----------------------------|---|
|                             | <p>Viola 5/27/93 discloses automatic invocation of the executable application. <i>See, e.g., :</i></p> <p>When ViolaWWW parses the VOBJF tag, it automatically invokes the vplot executable application. The automatic invocation does not require action by the user.</p> <p>For example, when ViolaWWW parses the VOBJF tag in testPlot.hmm1, the vplot application is automatically invoked as follows:</p> <pre>switch (pid = vfork()) { ... case 0: \* Child *\ ... execv(GET_path(self), args); (See src\cl_TTY.c.)</pre> <p>As another example, the doodle application is automatically invoked by the ViolaWWW browser upon parsing of the VOBJF tag as follows:</p> <pre>&lt;H2&gt;Noodle Doodles&lt;/H2&gt; &lt;VOBJF&gt;../apps/doodle.v&lt;/VOBJF&gt;</pre> <p>(See violaTOGO\docs\violaApps.hmm1)</p> <p>Viola 5/27/93 discloses that the executable application displays the object. <i>See, e.g., :</i></p> <p>For example, the vplot application displays the object as a grid inside the ViolaWWW window that is displaying testPlot.hmm1. The object is displayed at the location in the window corresponding to the VOBJF tag. (See violaTOGO\docs\testplot.hmm1.</p> <p>As another example, the doodle.v script displays the freehand drawing</p> |

| Claim Text from '985 Patent | Viola 5/27/93  |
|-----------------------------|--|
|                             | <p>inside the ViolaWWW window. The drawing is displayed at the location in the window corresponding to the VOBJF tag. (See violaTOGO\apps\doodle.v).</p> <p>Viola 5/27/93 discloses that the executable application enables direct interaction with the object. <i>See, e.g., :</i></p> <p>For example, the vplot executable application enables the user to directly interact with the object using the slider bars to rotate the object around the X, Y and Z axes. (See, e.g., violaTOGO\apps\plot.v, src\cl_slider.c, src\cl_client.c.)</p> <p>As another example, the doodle.v script enables the user to directly interact with the object using an input device to create a freeform drawing in the display area. (See violaTOGO\apps\doodle.v)</p> <p>Viola 5/27/93 discloses that interaction with the object is at a first location in the hypermedia document. <i>See, e.g., :</i></p> <p>For example, the vplot executable application enables the user to directly interact with the object using the slider bars to rotate the object around the X, Y and Z axes. (See, e.g., apps\plot.v, src\cl_slider.c, src\cl_client.c.) This interaction with the object occurs at the first location within the portion of the HMML document displayed in the ViolaWWW window. (See, e.g., docs\testPlot.hmml, apps\plot.v)</p> <p>Similarly, the doodle.v script enables the freeform drawing interaction described above which occurs at the first location within the portion of the HMML document displayed in the ViolaWWW window. (See violaTOGO\apps\doodle.v)</p> |



| Claim Text from '985 Patent   | Viola 5/27/93   |
|---|---|
| <p><b>985-2.a:</b><br/>The method of claim 1 where: the information to enable comprises text formats.</p> | <p>Viola 5/27/93 discloses that the enabling information in the file is text formats. <i>See, e.g., :</i></p> <p>ViolaWWW running on the client workstation can receive hypermedia document files (i.e., HTML and HMML documents) that contain enabling information from a network server (e.g., a file server or HTTP server) over a distributed hypermedia network environment. Examples of such documents include those in violaTOGO\docs.</p> <p>The hypermedia document files received from the network server contain enabling information in the form of text formats which enable a browser application to display at least a portion of a distributed hypermedia document within a browser-controlled window. For example, the hypermedia documents downloaded by ViolaWWW may contain HMML tags or HTML tags. For example, the testPlot.hmml hypermedia document contains the HMML tags (i.e., text formats) TITLE, H1 and ITALIC. (See, e.g., testPlot.hmml.) As another example, the testAll.html file contains HTML tags, such as TITLE and H1.</p> <p>The hypermedia document downloaded from the remote network server is parsed by ViolaWWW to identify the text formats. ViolaWWW then initiates processing specified by the tags. For example, ViolaWWW displays the text marked by the H1 tag in large, bold, header text and the text marked by the ITALIC tag in italics. (See, e.g., apps\VWHandler_hmml.v, src\sgmlsA2B.c and src\sgml.c, violaTOGO\libWWW.)</p> |
| <p><b>985-3.a:</b><br/>The method of claim 2 where the text formats are HTML tags.</p>                    | <p>Viola 5/27/93 discloses that the text formats are HTML tags. <i>See, e.g., :</i></p> <p>ViolaWWW running on the client workstation can receive hypermedia document files that contain text formats in the form of HTML tags from a network server (e.g., a file server or HTTP server) over a distributed hypermedia network environment. Examples of such documents include those in violaTOGO\docs.</p>  |

| Claim Text from '985 Patent  | Viola 5/27/93  |
|--|--|
|  | For example, the testAll.html file contains HTML tags, such as TITLE and H1  |
| <p><b>985-4.a:</b><br/>The method of claim 1 where the information contained in the file received comprises at least one embed text format.</p>  | <p>Viola 5/27/93 discloses that the enabling information in the file includes an embed text format. <i>See, e.g., :</i></p> <p>ViolaWWW running on the client workstation can receive hypermedia documents (e.g., HTML and HMML documents) from a network server (e.g., a file server or HTTP server) over the distributed hypermedia network environment. The hypermedia document received from the network server contains text formats which enable a browser application to display at least a portion of a distributed hypermedia document within a browser-controlled window. These text formats include embed text formats.</p> <p>For example, an HMML file can include an embed text format called VOBJF. (See, e.g., violaTOGO\docs\testPlot.hmml; docs\violaChier.hmml.)</p>  |
| <p><b>985-5.a:</b><br/>The method of claim 1 where the step of identifying an embed text format comprises: parsing the received file to identify text formats included in the received file.</p> | <p>Viola 5/27/93 discloses that the embed text format is identified by parsing the file containing enabling information. <i>See, e.g., :</i></p> <p>The file containing enabling information is downloaded from the remote network server and parsed by ViolaWWW to identify the text formats. ViolaWWW then initiates processing specified by the text formats. For example, ViolaWWW displays the text marked by the H1 tag in large, bold, header text and the text marked by the ITALIC tag in italics. (See, e.g., apps\VWHandler_hmml.v, src\sgmlsA2B.c and src\sgml.c, violaTOGO\libWWW.)</p> <p>For hypermedia documents containing embed text formats, these embed text formats are likewise identified by parsing the file. For example, when parsing testPlot.hmml or violaChier.hmml, violaWWW identifies the VOBJF tag while parsing. (See, e.g., src\sgml.c, apps\HMML_vobjf.v.)</p> |

| Claim Text from '985 Patent  | Viola 5/27/93   |
|--|---|
| <p><b>985-6.a:</b><br/>The method of claim 5 where the parsing is by a parser in the browser.</p>                          | <p>Viola 5/27/93 discloses that the parser is in the browser <i>See, e.g.,</i> :</p> <p>Viola includes files for parsing. (See, e.g., apps\VWHandler_hmml.v, src\sgmlsA2B.c and src\sgml.c, violaTOGO\libwww.)<br/>Furthermore, as discussed within my report, HTML and/or HMML parsers were readily available and inherent to the operation of world wide web browser applications.</p>  |
| <p><b>985-7.a:</b><br/>The method of claim 1 where the processing specified by the text formats is specified directly.</p> | <p>Viola 5/27/93 discloses that the text formats directly specify the processing. <i>See, e.g.,</i> :</p> <p>ViolaWWW running on the client workstation can receive hypermedia document files (i.e., HTML and HMML documents) that contain text formats from a network server (e.g., a file server or HTTP server) over a distributed hypermedia network environment. Examples of such documents include those in violaTOGO\docs.<br/>The hypermedia document files received from the network server contain predetermined text formats that directly specify processing. For example, the hypermedia documents downloaded by ViolaWWW contain HMML tags or HTML tags. In particular, the testPlot.hmml hypermedia document contains the HMML tags such as TITLE, H1 and ITALIC. (See, e.g., testPlot.hmml.) As another example, the testAll.html file contains HTML tags, such as TITLE and H1.<br/>The hypermedia document downloaded from the remote network server is parsed by ViolaWWW to identify the tags. ViolaWWW then initiates processing directly specified by the tags. For example, ViolaWWW displays the text marked by the H1 tag in large, bold, header text and the text marked by the ITALIC tag in italics. (See, e.g., apps\VWHandler_hmml.v, src\sgmlsA2B.c and src\sgml.c, violaTOGO\libWWW.)</p> |

| Claim Text from '985 Patent   | Viola 5/27/93   |
|---|---|
| <p><b>985-8.a:</b><br/>The method of claim 1 where the correspondence is implied by the order of the text format in a set of all of the text formats.</p> | <p>Viola 5/27/93 discloses that the correspondence is implied by the order of text formats. <i>See, e.g., :</i></p> <p>In ViolaWWW, the correspondence between the location in the document and the text formats is implied by the order of the text formats. For example, testPlot.hmml has several HMML tags (text formats). In testPlot.hmml, TITLE tag appears before H1 tag. H1 tag is followed by P tag. VOBJF tag appears later. When ViolaWWW displays the document, the title (associated with TITLE tag) is displayed first. The title is followed by heading (associated with H1 tag). A paragraph (associated with P tag) is displayed after the heading. An object is embedded later in the document where VOBJF tag is specified. (See, e.g., docs\testPlot.hmml.)</p> <p>Similarly, for testAll.html, a title (associated with a TITLE tag) is displayed ahead of Header 1 (associated with a subsequent header tag), which is displayed ahead of Header 2 (associated with a still subsequent header tag). (See docs\testAll.html.)</p> |
| <p><b>985-9.a:</b><br/>The method of claim 1 where the embed text format specifies the location of at least a portion of an object directly.</p>          | <p>Viola 5/27/93 discloses that the embed text format specifies the location of the object directly. <i>See, e.g., :</i></p> <p>The VOBJF embed text format specifies the location of an object directly. For example, testPlot.hmml includes a VOBJF tag specifying the location of an object based on a filepath location in which the object can be found:<br/>&lt;VOBJF&gt;/usr/work/viola/apps/plot.v&lt;\VOBJF&gt;<br/>(See violaTOGO/docs/testPlot.hmml.)</p>  |
| <p><b>985-10.a:</b><br/>The method of claim 1 where having type information associated is by including type information in the embed text format.</p>     | <p>Viola 5/27/93 discloses that the type information is in the embed text format. <i>See, e.g., :</i></p> <p>For example, the file plot.v (which contains type information as described above) is in the VOBJF embed text format.</p>   |

| Claim Text from '985 Patent  | Viola 5/27/93   |
|--|---|
|  | <p>&lt;VOBJF&gt;/usr/work/viola/apps/plot.v&lt;\VOBJF&gt;<br/>(See violaTOGO\docs\testPlot.hmm1.)</p>   |
| <p><b>985-11.a:</b><br/>The method of claim 1 where automatically invoking does not require interactive action by the user.</p>  | <p>Viola 5/27/93 discloses that automatic invocation does not require interactive action by the user. <i>See, e.g., :</i></p> <p>When ViolaWWW parses the VOBJF tag, it automatically invokes the vplot executable application. The automatic invocation does not require action by the user.</p> <p>For example, when ViolaWWW parses the VOBJF tag in testPlot.hmm1, the vplot application is automatically invoked without any interactive action by the user, as follows:</p> <pre>switch (pid = vfork()) { ... case 0: \* Child *\ ... execv(GET_path(self), args); (See violaTOGO\src\cl_TTY.c.)</pre> <p>As another example, the doodle application is automatically invoked by the ViolaWWW browser upon parsing of the VOBJF tag as follows:</p> <pre>&lt;H2&gt;Noodle Doodles&lt;/H2&gt; &lt;VOBJF&gt;../apps/doodle.v&lt;/VOBJF&gt;</pre> <p>(See violaTOGO\docs\violaApps.hmm1)</p> |
| <p><b>985-16.a:</b><br/>One or more computer readable media encoded with software comprising computer executable instructions, for use in a distributed hypermedia</p> | <p>Viola 5/27/93 discloses computer code physically embodied on a medium. <i>See, e.g., :</i></p> <p>The computer on which ViolaWWW executes includes computer usable</p>   |

| Claim Text from '985 Patent  | Viola 5/27/93  |
|--|--|
| network environment, wherein the network environment comprises at least one client workstation and one network server coupled to the network environment, and when the software is executed operable to:   | <p>media having computer readable program code physically embodied therein. By way of example, ViolaWWW executed on Sun SPARCstations running a SunOS operating system. ViolaWWW could be executed on the Sun SPARCstation with the command “viola -o www.” (See, e.g., violaTOGO\RUN.WWW.) Viola, including the www.v file that implements the ViolaWWW browser, contains computer readable code that is stored on the SPARCstation's computer usable media.</p> <p>Viola 5/27/93 discloses a client workstation and a network server in a distributed hypermedia environment. <i>See</i> evidence recited for 985-1.a.</p> |
| <b>985-16.b:</b><br>receive, at the client workstation from the network server over the network environment, at least one file containing information to enable a browser application to display at least a portion of a distributed hypermedia document within a browser-controlled window; | Viola 5/27/93 discloses a browser application; a file containing enabling information received from a server; that the browser displays at least a portion of a distributed hypermedia document; and that the display is in a browser-controlled window. <i>See</i> evidence recited for 985-1.b.  |
| <b>985-16.c:</b><br>cause the client workstation to utilize the browser to:  | Viola 5/27/93 discloses a browser application executing on the client workstation. <i>See</i> evidence recited for 985-1.c.  |
| <b>985-16.d:</b><br>respond to text formats to initiate processing specified by the text formats;  | Viola 5/27/93 discloses parsing text formats. <i>See</i> evidence recited for 985-1.d.   |
| <b>985-16.e:</b><br>display at least a portion of the document within the browser-controlled window;   | Viola 5/27/93 discloses displaying at least a portion of the document within the browser-controlled window. <i>See</i> evidence recited for 985-1.e.   |
| <b>985-16.f:</b><br>identify an embed text format corresponding to a first location in the document, the embed text format specifying the location of at least a portion of an object external to the file, with the object having type information associated with it;                      | Viola 5/27/93 discloses identifying an embed text format; that the embed text format corresponds to a first location in a hypermedia document; that the embed text format specifies the location of at least a portion of an object external to the file containing enabling information; and that the object has associated type information. <i>See</i> evidence recited for 985-1.f.  |

| Claim Text from '985 Patent   | Viola 5/27/93   |
|---|---|
| <b>985-16.g:</b><br>utilize the type information to identify and locate an executable application external to the file; and   | Viola 5/27/93 discloses using type information to identify and locate an executable application external to the file. <i>See</i> evidence recited for 985-1.g.  |
| <b>985-16.h:</b><br>automatically invoke the executable application, in response to the identifying of the embed text format, to execute on the client workstation in order to display the object and enable an end-user to directly interact with the object while the object is being displayed within a display area created at the first location within the portion of the hypermedia document being displayed in the browser-controlled window. | Viola 5/27/93 discloses automatically invoking the executable application; that the executable application displays the object and enables an end-user to directly interact with it; and that the interaction with the object is at a first location in a hypermedia document. <i>See</i> evidence recited for 985-1.h. |
| <b>985-17.a:</b><br>The computer readable media of claim 16 where: the information to enable comprises text formats.  | Viola 5/27/93 discloses that the enabling information in the file is text formats. <i>See</i> evidence recited for 985-2.a.   |
| <b>985-18.a:</b><br>The computer readable media of claim 17 where: the text formats are HTML tags.  | Viola 5/27/93 discloses that the text formats are HTML tags. <i>See</i> evidence recited for 985-3.a.   |
| <b>985-19.a:</b><br>The computer readable media of claim 16 where: the information contained in the file received comprises at least one embed text format.   | Viola 5/27/93 discloses that the enabling information in the file includes an embed text format. <i>See</i> evidence recited for 985-4.a.   |
| <b>985-20.a:</b><br>A method of serving digital information in a computer network environment having a network server coupled the network environment, and where the network environment is a distributed   | Viola 5/27/93 discloses digital information. <i>See, e.g.,</i> :<br><br>The information that is exchanged between a client workstation running ViolaWWW and a network server (such as web server or file server) is digital information.  |

| Claim Text from '985 Patent   | Viola 5/27/93  |
|---|--|
| hypermedia environment, the method comprising:  | <p>For example, ViolaWWW running on the client workstation can receive hypermedia documents (HTML and HMML documents) from a network server over the distributed hypermedia network environment. These documents are transmitted according to network protocols that transmit information in digital form. Examples of the types of documents that can be served in digital form can be found in violaTOGO\docs, including testPlot.hmml and testAll.html.</p> <p>ViolaWWW also could receive *.v files over networks in digital form. Examples of *.v files are in violaTOGO\apps, and include plot.v and doodle.v.</p> <p>Viola 5/27/93 discloses a network server in a distributed hypermedia environment. <i>See</i> evidence recited for 985-1.a.</p>   |
| <p><b>985-20.b:</b><br/>communicating via the network server with at least one client workstation over said network in order to cause said client workstation to:</p> | <p>Viola 5/27/93 discloses a client workstation. <i>See</i> evidence recited for 985-1.a.</p> <p>Viola 5/27/93 discloses communicating via network server in order to cause the client workstation to act. <i>See, e.g., :</i></p> <p>The ViolaWWW browser operating on a client workstation requests hypermedia documents from a server, such as a file server or an HTTP server. The server responds by communicating to the client workstation. By way of example, the server transmits hypermedia documents, examples of which can be found in violaTOGO\docs. The server can transmit *.v files, examples of which are found in violaTOGO\apps. In one example, the server transmits testPlot.hmml and plot.v to the client workstation. The client workstation, after receiving the hypermedia document and/or *.v file from the network server, parses the hypermedia document and responds to the text formats contained in the hypermedia document to initiate processing specified by the text formats and/or by initiating processing according to a *.v file. (See, e.g., src\sgmlsA2B.c and src\sgml.c.). Similar behavior can be exhibited with respect to doodle.v.</p> |
| <b>985-20.c:</b>  | Viola 5/27/93 discloses a browser application; a file containing enabling  |



| <b>Claim Text from '985 Patent</b>   | <b>Viola 5/27/93</b>  |
|--|---|
| receive, over said network environment from said server, at least one file containing information to enable a browser application to display at least a portion of a distributed hypermedia document within a browser-controlled window;   | information received from a server; that the browser displays at least a portion of a distributed hypermedia document; and that the display is in a browser-controlled window. <i>See</i> evidence recited for 985-1.b.   |
| <b>985-20.d:</b><br>execute, at said client workstation, a browser application, with the browser application:  | Viola 5/27/93 discloses a browser application executing on the client workstation. <i>See</i> evidence recited for 985-1.c.   |
| <b>985-20.e:</b><br>responding to text formats to initiate processing specified by the text formats;   | Viola 5/27/93 discloses parsing text formats. <i>See</i> evidence recited for 985-1.d.  |
| <b>985-20.f:</b><br>displaying, on said client workstation, at least a portion of the document within the browser-controlled window;   | Viola 5/27/93 discloses displaying at least a portion of the document within the browser-controlled window. <i>See</i> evidence recited for 985-1.e.  |
| <b>985-20.g:</b><br>identifying an embed text format which corresponds to a first location in the document, where the embed text format specifies the location of at least a portion of an object external to the file, where the object has type information associated with it;  | Viola 5/27/93 discloses identifying an embed text format; that the embed text format corresponds to a first location in a hypermedia document; that the embed text format specifies the location of at least a portion of an object external to the file containing enabling information; and that the object has associated type information. <i>See</i> evidence recited for 985-1.f. |
| <b>985-20.h:</b><br>utilizing the type information to identify and locate an executable application external to the file; and  | Viola 5/27/93 discloses using type information to identify and locate an executable application external to the file. <i>See</i> evidence recited for 985-1.g.  |
| <b>985-20.i:</b><br>automatically invoking the executable application, in response to the identifying of the embed text format, to execute on the client workstation in order to display the object and enable an end-user to directly interact with the object while the object is being displayed within a display area created at | Viola 5/27/93 discloses automatically invoking the executable application; that the executable application displays the object and enables an end-user to directly interact with it; and that the interaction with the object is at a first location in a hypermedia document. <i>See</i> evidence recited for 985-1.h.   |

| Claim Text from '985 Patent  | Viola 5/27/93   |
|--|---|
| the first location within the portion of the hypermedia document being displayed in the browser-controlled window.   |   |
| <b>985-21.a:</b><br>The method of claim 20 where: the information to enable comprises text formats.  | Viola 5/27/93 discloses that the enabling information in the file is text formats. <i>See</i> evidence recited for 985-2.a.   |
| <b>985-22.a:</b><br>The method of claim 21 where: the text formats are HTML tags.  | Viola 5/27/93 discloses that the text formats are HTML tags. <i>See</i> evidence recited for 985-3.a.   |
| <b>985-23.a:</b><br>The method of claim 20 where: the information contained in the file received comprises at least one embed text format.   | Viola 5/27/93 discloses that the enabling information in the file includes an embed text format. <i>See</i> evidence recited for 985-4.a.   |
| <b>985-24.a:</b><br>A method for running an executable application in a computer network environment, wherein said network environment has at least one client workstation and one network server coupled to a network environment, the method comprising:   | Viola 5/27/93 discloses a client workstation and a network server in a network environment. <i>See</i> evidence recited for 985-1.a.<br><br>Viola 5/27/93 discloses an executable application. <i>See</i> evidence recited for 985-1.g.   |
| <b>985-24.b:</b><br>enabling an end-user to directly interact with an object by utilizing said executable application to interactively process said object while the object is being displayed within a display area created at a first location within a portion of a hypermedia document being displayed in a browser-controlled window, | Viola 5/27/93 discloses displaying at least a portion of the document within the browser-controlled window. <i>See</i> evidence recited for 985-1.e.<br><br>Viola 5/27/93 discloses an object external to a file containing enabling information. <i>See</i> evidence recited for 985-1.f.<br><br>Viola 5/27/93 discloses that there is enabling of an end-user to directly interact with the object. <i>See, e.g., :</i> |

| Claim Text from '985 Patent  | Viola 5/27/93   |
|--|---|
|  | <p>For example, the vplot executable application enables the user to directly interact with the object using the slider bars to rotate the object around the X, Y and Z axes. (See, e.g., apps\plot.v, src\cl_slider.c, src\cl_client.c.)</p> <p>Viola 5/27/93 discloses that the interaction with the object is at a first location in a hypermedia document. <i>See</i> evidence recited for 985-1.h.</p> <p>Viola 5/27/93 discloses that the object is displayed at a first location within a portion of the hypermedia document being displayed. <i>See, e.g., :</i></p> <p>In one example, the object is displayed and interactive processing of the object occurs at the first location within the portion of the hypermedia document displayed in the ViolaWWW window.</p> <p>For example, the vplot application displays the object as a grid (the default grid) inside the ViolaWWW window. The object is displayed at the first location in the portion of the testPlot.hmml hypermedia document being displayed in the ViolaWWW window. (See, e.g., violaTOGO\docs\testPlot.hmml.)</p> <p>As another example, the doodle.v script displays the freehand drawing inside the ViolaWWW window. The drawing is displayed at the location in the window corresponding to the location specified in the script. (See violaTOGO\apps\doodle.v).</p> |
| <p><b>985-24.c:</b><br/>wherein said network environment is a distributed hypermedia environment,</p>  | <p>Viola 5/27/93 discloses a client workstation and a network server in a distributed hypermedia environment. <i>See</i> evidence recited for 985-1.a.</p>  |
| <p><b>985-24.d:</b><br/>wherein said client workstation receives, over said network environment from said server, at least one file containing information to enable said browser application to display, on said client workstation, at least said portion of said distributed hypermedia</p> | <p>Viola 5/27/93 discloses a browser application; a file containing enabling information received from a server; that the browser displays at least a portion of a distributed hypermedia document; and that the display is in a browser-controlled window. <i>See</i> evidence recited for 985-1.b.</p>  |

| Claim Text from '985 Patent   | Viola 5/27/93   |
|---|---|
| document within said browser-controlled window,   |   |
| <b>985-24.e:</b><br>wherein said executable application is external to said file,   | Viola 5/27/93 discloses an executable application external to the file. <i>See</i> evidence recited for 985-1.g.  |
| <b>985-24.f:</b><br>wherein said client workstation executes the browser application, with the browser application responding to text formats to initiate processing specified by the text formats, | Viola 5/27/93 discloses a browser application executing on the client workstation. <i>See</i> evidence recited for 985-1.c.<br><br>Viola 5/27/93 discloses parsing text formats. <i>See</i> evidence recited for 985-1.d. |
| <b>985-24.g:</b><br>wherein at least said portion of the document is displayed within the browser-controlled window,  | Viola 5/27/93 discloses displaying at least a portion of the document within the browser-controlled window. <i>See</i> evidence recited for 985-1.e.  |
| <b>985-24.h:</b><br>wherein an embed text format which corresponds to said first location in the document is identified by the browser,   | Viola 5/27/93 discloses identifying an embed text format and that the embed text format corresponds to a first location in a hypermedia document. <i>See</i> evidence recited for 985-1.f.                                |
| <b>985-24.i:</b><br>wherein the embed text format specifies the location of at least a portion of said object external to the file,   | Viola 5/27/93 discloses that the embed text format specifies the location of at least a portion of an object external to the file containing enabling information. <i>See</i> evidence recited for 985-1.f.               |
| <b>985-24.j:</b><br>wherein the object has type information associated with it,   | Viola 5/27/93 discloses that the object has associated type information. <i>See</i> evidence recited for 985-1.f.   |
| <b>985-24.k:</b><br>wherein the type information is utilized by the browser to identify and locate said executable application, and   | Viola 5/27/93 discloses using type information to identify and locate an executable application external to the file. <i>See</i> evidence recited for 985-1.g.  |
| <b>985-24.l:</b><br>wherein the executable application is automatically invoked by the browser, in response to the identifying of the embed text format.  | Viola 5/27/93 discloses automatically invoking the executable application. <i>See</i> evidence recited for 985-1.h.   |
|   |   |

| Claim Text from '985 Patent  | Viola 5/27/93  |
|--|--|
| <b>985-25.a:</b><br>The method of claim 24 where: the information to enable comprises text formats.  | Viola 5/27/93 discloses that the enabling information in the file is text formats. <i>See evidence recited for 985-2.a.</i>  |
| <b>985-26.a:</b><br>The method of claim 25 where: the text formats are HTML tags.  | Viola 5/27/93 discloses that the text formats are HTML tags. <i>See evidence recited for 985-3.a.</i>  |
| <b>985-27.a:</b><br>The method of claim 24 where: the information contained in the file received comprises at least one embed text format.   | Viola 5/27/93 discloses that the enabling information in the file includes an embed text format. <i>See evidence recited for 985-4.a.</i>  |
| <b>985-28.a:</b><br>One or more computer readable media encoded with software comprising an executable application for use in a system having at least one client workstation and one network server coupled to a network environment, operable to:  | <p>Viola 5/27/93 discloses computer code physically embodied on a medium. <i>See evidence recited for 985-16.a.</i></p> <p>Viola 5/27/93 discloses a client workstation and a network server in a network environment. <i>See evidence recited for 985-1.a.</i></p> <p>Viola 5/27/93 discloses an executable application. <i>See evidence recited for 985-1.g.</i></p>   |
| <b>985-28.b:</b><br>cause the client workstation to display an object and enable an end-user to directly interact with said object while the object is being displayed within a display area created at a first location within a portion of a hypermedia document being displayed in a browser-controlled window, | <p>Viola 5/27/93 discloses displaying at least a portion of the document within the browser-controlled window. <i>See evidence recited for 985-1.e.</i></p> <p>Viola 5/27/93 discloses an object external to a file containing enabling information. <i>See evidence recited for 985-1.f.</i></p> <p>Viola 5/27/93 discloses that there is enabling of an end-user to directly interact with the object. <i>See evidence recited for 985-24.b.</i></p> <p>Viola 5/27/93 discloses that the interaction with the object is at a first location in a hypermedia document. <i>See evidence recited for 985-1.h.</i></p> |

| Claim Text from '985 Patent  | Viola 5/27/93   |
|--|---|
|  | <p>Viola 5/27/93 discloses that the object is displayed within a display area created at the first location. <i>See, e.g.,</i> :</p> <p>In one example, the object is displayed and interactive processing of the object occurs at the first location within the portion of the hypermedia document displayed in the ViolaWWW window.</p> <p>For example, the vplot application displays the object as a grid (the default grid) inside the ViolaWWW window. The object is displayed at the first location in the portion of the testPlot.hmml hypermedia document being displayed in the ViolaWWW window. (See, e.g., violaTOGO\docs\testPlot.hmml.)</p> <p>As another example, the doodle.v script displays the freehand drawing inside the ViolaWWW window. The drawing is displayed at the location in the window corresponding to the location specified in the script. (See violaTOGO\apps\doodle.v).</p> |
| <p><b>985-28.c:</b><br/>wherein said network environment is a distributed hypermedia environment,</p>  | <p>Viola 5/27/93 discloses a client workstation and a network server in a distributed hypermedia environment. <i>See</i> evidence recited for 985-1.a.</p>  |
| <p><b>985-28.d:</b><br/>wherein said client workstation receives, over said network environment from said server, at least one file containing information to enable said browser application to display, on said client workstation, at least said portion of said distributed hypermedia document within said browser-controlled window,</p> | <p>Viola 5/27/93 discloses a browser application; a file containing enabling information received from a server; that the browser displays at least a portion of a distributed hypermedia document; and that the display is in a browser-controlled window. <i>See</i> evidence recited for 985-1.b.</p>  |
| <p><b>985-28.e:</b><br/>wherein said executable application is external to said file,</p>  | <p>Viola 5/27/93 discloses an executable application external to the file. <i>See</i> evidence recited for 985-1.g.</p>   |
| <p><b>985-28.f:</b><br/>wherein said client workstation executes said</p>  | <p>Viola 5/27/93 discloses a browser application executing on the client workstation. <i>See</i> evidence recited for 985-1.c.</p>  |

| Claim Text from '985 Patent   | Viola 5/27/93   |
|---|---|
| browser application, with the browser application responding to text formats to initiate processing specified by the text formats,  | Viola 5/27/93 discloses parsing text formats. <i>See</i> evidence recited for 985-1.d.  |
| <b>985-28.g:</b><br>wherein at least said portion of the document is displayed within the browser-controlled window,  | Viola 5/27/93 discloses displaying at least a portion of the document within the browser-controlled window. <i>See</i> evidence recited for 985-1.e.  |
| <b>985-28.h:</b><br>wherein an embed text format which corresponds to said first location in the document is identified by the browser,   | Viola 5/27/93 discloses identifying an embed text format and that the embed text format corresponds to a first location in a hypermedia document. <i>See</i> evidence recited for 985-1.f.                  |
| <b>985-28.i:</b><br>wherein the embed text format specifies the location of at least a portion of said object external to the file,   | Viola 5/27/93 discloses that the embed text format specifies the location of at least a portion of an object external to the file containing enabling information. <i>See</i> evidence recited for 985-1.f. |
| <b>985-28.j:</b><br>wherein the object has type information associated with it,   | Viola 5/27/93 discloses that the object has associated type information. <i>See</i> evidence recited for 985-1.f.   |
| <b>985-28.k:</b><br>wherein the type information is utilized by the browser to identify and locate said executable application, and   | Viola 5/27/93 discloses using type information to identify and locate an executable application external to the file. <i>See</i> evidence recited for 985-1.g.  |
| <b>985-28.l:</b><br>wherein the executable application is automatically invoked by the browser, in response to the identifying of the embed text format.  | Viola 5/27/93 discloses automatically invoking the executable application. <i>See</i> evidence recited for 985-1.h.   |
|   |   |
| <b>985-36.a:</b><br>A method for running an application program in a distributed hypermedia network environment, wherein the distributed hypermedia network environment comprises at least one client workstation and one remote network server | Viola 5/27/93 discloses an application program in a distributed hypermedia environment comprising at least client workstation and network server. <i>See</i> evidence recited for 985-1.a.                  |

| Claim Text from '985 Patent   | Viola 5/27/93   |
|---|---|
| coupled to the distributed hypermedia network environment, the method comprising:   |   |
| <b>985-36.b:</b><br>receiving, at the client workstation from the network server over the distributed hypermedia network environment, at least one file containing information to enable a browser application to display at least a portion of a distributed hypermedia document within a browser-controlled window; | Viola 5/27/93 discloses a browser application; a file containing enabling information; that the file is received at the client workstation from the network server; that the browser displays at least a portion of a distributed hypermedia document; and that at least a portion of a hypermedia document is displayed in a browser-controlled window. <i>See</i> evidence recited for 985-1.b.   |
| <b>985-36.c:</b><br>executing the browser application on the client workstation, with the browser application:  | Viola 5/27/93 discloses a browser application executing on the client workstation. <i>See</i> evidence recited for 985-1.c.   |
| <b>985-36.d:</b><br>responding to text formats to initiate processing specified by the text formats;  | Viola 5/27/93 discloses parsing text formats. <i>See</i> evidence recited for 985-1.d.  |
| <b>985-36.e:</b><br>displaying at least a portion of the document within the browser-controlled window;   | Viola 5/27/93 discloses displaying at least a portion of the document within the browser-controlled window. <i>See</i> evidence recited for 985-1.e.  |
| <b>985-36.f:</b><br>identifying an embed text format which corresponds to a first location in the document, where the embed text format specifies the location of at least a portion of an object;  | <p>Viola 5/27/93 discloses an object. <i>See, e.g., :</i></p> <p style="padding-left: 40px;">In one example, when a Viola applet is embedded in a ViolaWWW web page, using the VOBJF tag, at least a portion of an object, the default grid, appears in the ViolaWWW window upon browser launch.</p> <p style="padding-left: 40px;">The data for the default grid is specified in the file plot.v by the command:<br/>output("equation 0");<br/>(See apps\plot.v.)</p> <p>Viola 5/27/93 discloses identifying an embed text format; that the embed text format corresponds to a first location in the hypermedia document; and that the embed text format specifies the location of an object. <i>See</i> evidence recited for 985-1.f.</p> |

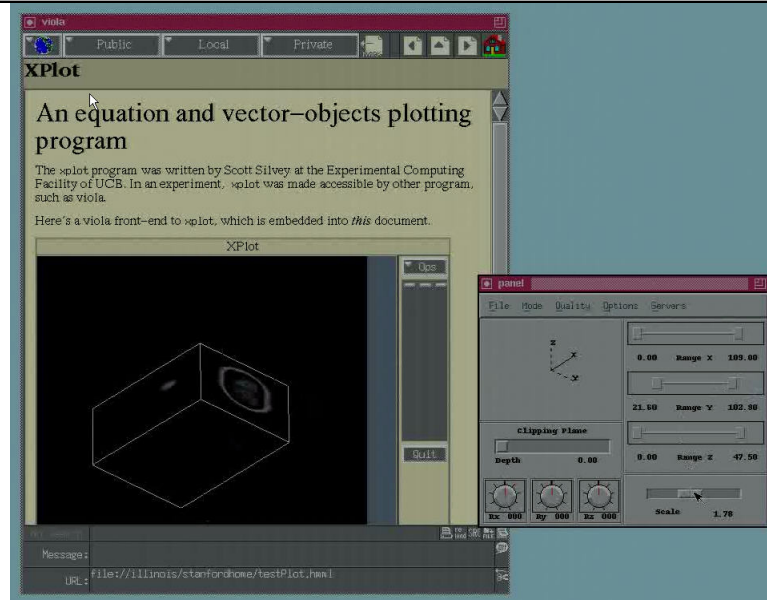


| Claim Text from '985 Patent  | Viola 5/27/93   |
|--|---|
| <p><b>985-36.g:</b><br/>identifying and locating an executable application associated with the object; and</p>   | <p>Viola 5/27/93 discloses that the browser identifies and locates an executable application associated with the object. <i>See, e.g.,</i></p> <p>For example, the file plot.v contains type information associated with the object.</p> <pre>/path{/usr/work/vplot/vplot} (See apps\plot.v.)</pre> <p>The type information is used by the ViolaWWW to identify and locate the vplot executable application. ViolaWWW then invokes the vplot executable application.</p> <pre>switch (pid = vfork()) { ... case 0: \* Child *\ ... execv(GET_path(self), args); (See src\cl_TTY.c.)</pre> <p>As another example, the file doodle.v also contains type information associated with the object.</p> <pre>\script      { switch (arg[0]) { ... } usual(); }</pre> <p>(See violaTOGO\apps\doodle.v)</p> |
| <p><b>985-36.h:</b><br/>automatically invoking the executable application, in response to the identifying of the embed text format, in order to enable an end-user to directly interact with the object, while the object is being</p> | <p>Viola 5/27/93 discloses identifying an embed text format. <i>See</i> evidence recited in 985-1.f.</p> <p>Viola 5/27/93 discloses automatic invocation of the executable application; that the executable application displays the object; that the executable application</p>  |

| Claim Text from '985 Patent  | Viola 5/27/93  |
|--|--|
| <p>displayed within a display area created at the first location within the portion of the hypermedia document being displayed in the browser-controlled window,</p> | <p>enables direct interaction with the object; and that interaction with the object is at a first location in the hypermedia document. <i>See</i> evidence recited in 985-1.h.</p> <p>Viola 5/27/93 discloses that the object is displayed at a first location within a portion of the hypermedia document being displayed. <i>See</i> evidence recited at 985-24.b.</p> <p>Viola 5/27/93 discloses that a hypermedia document is displayed in a browser window. <i>See, e.g.,</i> evidence recited for 985-1.e.</p>   |
| <p><b>985-36.i:</b><br/>wherein the executable application is part of a distributed application, and</p>   | <p>Viola 5/27/93 discloses a distributed application. <i>See, e.g.,</i> :</p> <p>Viola's documentation states that "Through a communication facility such as the socket, a message may also be generated from another process on the network." (See, e.g., violaTOGO\docs\violaBrief.hmm1.) By way of example, and as discussed elsewhere in this chart, ViolaWWW provides for automatically invoking the vplot executable application using the execv system call. (See violaTOGO\src\cl_TTY.c.) Vplot can be implemented as a distributed application wherein a portion of the computations — for example, graphics transformations — can be executed on a remote server by using the rexec socket-based system call or other Unix socket-based system calls in lieu of the execv system call. In this case, a portion of the computations would be performed on a process on a remote server.</p> <p>Furthermore, as discussed in my report and in the videos Viola video 14.avi, and Viola video 15.avi, vplot can be interchanged with any application, and this would include distributed applications such as VIS. See example below. ViolaWWW works with a distributed application in the same way that it works with an executable application on the client workstation.</p> |

## Claim Text from '985 Patent

Viola 5/27/93



VIS

Viola 5/27/93 discloses that the executable application is part of a distributed application. *See, e.g.,* :

Viola's documentation states that "Through a communication facility such as the socket, a message may also be generated from another process on the network." (See, e.g., `violaTOGO/docs/violaBrief.html`.) By way of example, and as discussed elsewhere in this chart, ViolaWWW provides for automatically invoking the `vplot` executable application using the `execv` system call. (See `violaTOGO/src/cl_TTY.c`.) `Vplot` can be implemented as a distributed application wherein a portion of the computations — for example, graphics transformations — can be executed on a remote server by using the `rexec` socket-based system call or other Unix socket-based system calls in lieu of the `execv` system call. In this

| Claim Text from '985 Patent  | Viola 5/27/93   |
|--|---|
|  | <p>case, the executable application is part of a distributed application. Furthermore, as discussed in my report and in Viola video 14.avi, and Viola video 15.avi, vplot can be interchanged with any application including distributed applications such as VIS. ViolaWWW works with a distributed application in the same way that it works with an executable application on the client workstation.</p>  |
| <p><b>985-36.j:</b><br/>wherein at least a portion of the distributed application is for execution on a remote network server coupled to the distributed hypermedia network environment.</p> | <p>Viola 5/27/93 discloses that the distributed application executes at least partially on a network server. <i>See, e.g.,</i> :</p> <p>Viola's documentation states that "Through a communication facility such as the socket, a message may also be generated from another process on the network." (See, e.g., violaTOGO\docs\violaBrief.hmm.) By way of example, and as discussed elsewhere in this chart, ViolaWWW provides for automatically invoking the vplot executable application using the <code>execv</code> system call. (See violaTOGO\src\cl_TTY.c.) Vplot can be implemented as a distributed application wherein a portion of the computations — for example, graphics transformations — can be executed on a remote server by using the <code>rexec</code> socket-based system call or other Unix socket-based system calls in lieu of the <code>execv</code> system call. In this case, a portion of the computations would be performed on a process on a remote server.</p> <p>Furthermore, as discussed in my report and in Viola video 14.avi, and Viola video 15.avi, vplot can be interchanged with other applications, including distributed applications such as VIS. ViolaWWW works with a distributed application in the same way that it works with an executable application on the client workstation.</p> |
|  |   |
| <p><b>985-37.a:</b><br/>The method of claim 36 where: the information to enable comprises text formats.</p>  | <p>Viola 5/27/93 discloses that the enabling information in the file is text formats. <i>See</i> evidence recited for 985-2.a.</p>  |
|  |   |
| <p><b>985-38.a:</b></p>  | <p>Viola 5/27/93 discloses that the text formats are HTML tags. <i>See</i> evidence</p>   |

| Claim Text from '985 Patent   | Viola 5/27/93   |
|---|---|
| The method of claim 37 where: the text formats are HTML tags.   | recited for 985-3.a.  |
| <b>985-39.a:</b><br>The method of claim 36 where: the information contained in the file received comprises at least one embed text format.  | Viola 5/27/93 discloses that the enabling information in the file includes an embed text format. <i>See</i> evidence recited for 985-4.a.   |
| <b>985-40.a:</b><br>A method of serving digital information in a computer network environment having a network server coupled to said computer network environment, and where the network environment is a distributed hypermedia network environment, the method comprising: | Viola 5/27/93 discloses digital information. <i>See</i> evidence recited for 985-20.a.<br><br>Viola 5/27/93 discloses a network server in a distributed hypermedia environment. <i>See</i> evidence recited for 985-1.a.  |
| <b>985-40.b:</b><br>communicating via the network server with at least one remote client workstation over said computer network environment in order to cause said client workstation to:   | Viola 5/27/93 discloses a client workstation. <i>See</i> evidence recited for 985-1.a.<br><br>Viola 5/27/93 discloses communicating via network server in order to cause the client workstation to act. <i>See</i> evidence recited for 985-20.b.   |
| <b>985-40.c:</b><br>receive, over said computer network environment from the network server, at least one file containing information to enable a browser application to display at least a portion of a distributed hypermedia document within a browser-controlled window;  | Viola 5/27/93 discloses a browser application; a file containing enabling information received from a server; that the browser displays at least a portion of a distributed hypermedia document; and that the display is in a browser-controlled window. <i>See</i> evidence recited for 985-1.b. |
| <b>985-40.d:</b><br>execute, at said client workstation, a browser application, with the browser application:   | Viola 5/27/93 discloses a browser application executing on the client workstation. <i>See</i> evidence recited for 985-1.c.   |
| <b>985-40.e:</b><br>responding to text formats to initiate processing   | Viola 5/27/93 discloses parsing text formats. <i>See</i> evidence recited for 985-1.d.  |

| Claim Text from '985 Patent   | Viola 5/27/93  |
|---|--|
| specified by the text formats;  |  |
| <b>985-40.f:</b><br>displaying, on said client workstation, at least a portion of the document within the browser-controlled window;  | Viola 5/27/93 discloses displaying at least a portion of the document within the browser-controlled window. <i>See</i> evidence recited for 985-1.e.   |
| <b>985-40.g:</b><br>identifying an embed text format which corresponds to a first location in the document, where the embed text format specifies the location of at least a portion of an object;  | Viola 5/27/93 discloses an object. <i>See</i> evidence recited for 985-36.f.<br><br>Viola 5/27/93 discloses identifying an embed text format; that the embed text format corresponds to a first location in the hypermedia document; and that the embed text format specifies the location of an object. <i>See</i> evidence recited for 985-1.f.  |
| <b>985-40.h:</b><br>identifying and locating an executable application associated with the object; and  | Viola 5/27/93 discloses that the browser identifies and locates an executable application associated with the object. <i>See</i> evidence recited for 985-36.g.  |
| <b>985-40.i:</b><br>automatically invoking the executable application, in response to the identifying of the embed text format, in order to enable an end-user to directly interact with the object while the object is being displayed within a display area created at the first location within the portion of the hypermedia document being displayed in the browser-controlled window, | Viola 5/27/93 discloses identifying an embed text format. <i>See</i> evidence recited in 985-1.f.<br><br>Viola 5/27/93 discloses automatic invocation of the executable application; that the executable application displays the object; that the executable application enables direct interaction with the object; and that interaction with the object is at a first location in the hypermedia document. <i>See</i> evidence recited in 985-1.h.<br><br>Viola 5/27/93 discloses that the object is displayed at a first location within a portion of the hypermedia document being displayed. <i>See</i> evidence recited for 985-24.b.<br><br>Viola 5/27/93 discloses that a hypermedia document is displayed in a browser window. <i>See, e.g.,</i> evidence recited for 985-1.e. |
| <b>985-40.j:</b>  | Viola 5/27/93 discloses that the executable application is part of a distributed   |

| Claim Text from '985 Patent  | Viola 5/27/93   |
|--|---|
| wherein the executable application is part of a distributed application, and   | application. <i>See</i> evidence recited in 985-36.i.   |
| <b>985-40.k:</b><br>wherein at least a portion of the distributed application is for execution on the network server.                      | Viola 5/27/93 discloses that the distributed application executes at least partially on a network server. <i>See</i> evidence recited for 985-36.j. |
| <b>985-41.a:</b><br>The method of claim 40 where: the information to enable comprises text formats.  | Viola 5/27/93 discloses that the enabling information in the file is text formats. <i>See</i> evidence recited for 985-2.a.                         |
| <b>985-42.a:</b><br>The method of claim 41 where: the text formats are HTML tags.  | Viola 5/27/93 discloses that the text formats are HTML tags. <i>See</i> evidence recited for 985-3.a.   |
| <b>985-43.a:</b><br>The method of claim 40 where: the information contained in the file received comprises at least one embed text format. | Viola 5/27/93 discloses that the enabling information in the file includes an embed text format. <i>See</i> evidence recited for 985-4.a.           |