

**IN THE UNITED STATES DISTRICT COURT
FOR THE DISTRICT OF DELAWARE**

SOFTVIEW LLC,)
)
Plaintiff,)
)
v.)
)
APPLE INC.; AT&T MOBILITY LLC;)
)
DELL INC.; HTC CORP.; HTC)
)
AMERICA, INC.; HUAWEI)
)
TECHNOLOGIES CO., LTD.;)
)
FUTUREWEI TECHNOLOGIES, INC.;)
)
KYOCERA CORP.; KYOCERA)
)
WIRELESS CORP.; LG)
)
ELECTRONICS, INC.; LG)
)
ELECTRONICS USA, INC.; LG)
)
ELECTRONICS MOBILECOMM)
)
U.S.A., INC.; MOTOROLA MOBILITY)
)
INC.; SAMSUNG ELECTRONICS CO.,)
)
LTD.; SAMSUNG ELECTRONICS)
)
AMERICA, INC.; SAMSUNG)
)
TELECOMMUNICATIONS AMERICA,)
)
LLC; and SONY ERICSSON MOBILE)
)
COMMUNICATIONS (USA) INC.,)
)
Defendants.)
)
)

Civil Action Nos. **10-389-LPS**
12-984-LPS
12-985-LPS
12-986-LPS
12-987-LPS
12-988-LPS
12-989-LPS
12-990-LPS
12-991-LPS

JOINT CLAIM CONSTRUCTION CHART

VOLUME 1 OF 2

EXHIBIT Petitioner - Motorola PX 1019a
--

America, Inc.; Samsung Telecommunications America, LLC; and Sony Ericsson Mobile Communications (USA) Inc. (collectively “Defendants”) hereby submit this Joint Claim Construction Chart.

I. UNDISPUTED CLAIM TERMS

The parties do not dispute the constructions of the claim terms included in the following chart:

Term/Phrase in Issue and Asserted Claim(s) in Which Term/Phrase Appears	Agreed Construction
object[s] ‘353 Patent: 58, 59, 139 ‘926 Patent: 72	a grouping of webpage content, such as a paragraph of text or a graphic image
bounding box ‘353 Patent: 59, 139	an outlined shape within which an object’s content (<i>e.g.</i> , text or graphic image) will appear

II. PROPOSED CONSTRUCTIONS OF DISPUTED CLAIM TERMS

The following chart identifies the disputed terms/phrases of the claims in issue and each side’s proposed construction of the disputed claim language. A chart including SoftView’s citations to the intrinsic evidence in support of its respective proposed constructions is attached hereto as Appendix A. A chart including Defendants’ citations to the intrinsic evidence in support of their respective proposed constructions is attached hereto as Appendix B. A copy of U.S. Patent No. 7,461,353 (“the ‘353 patent”) is attached hereto as Exhibit 1. A copy of U.S. Patent No. 7,831,926 (“the ‘926 patent”) is attached hereto as Exhibit 2. A copy of those portions of the intrinsic record relied upon by SoftView is attached hereto as Exhibit 3. A copy of those portions of the intrinsic record relied upon by Defendants is attached hereto as Exhibit 4.

	Term/Phrase in Issue and Asserted Claim(s) in Which Term/Phrase Appears	Plaintiff's Proposed Construction	Defendants' Proposed Construction
1	<p>scalable content</p> <p>'353 Patent: 33, 43, 48 by reference to 36, 52 by reference to 36, 58 by reference to 36, 59 by reference to 36, 66 by reference to 36, 139 by reference to 118, 183, 283</p> <p>'926 Patent: 55 by reference to 52, 59, 72 by reference to 52, 75 by reference to 52</p>	<p>graphic content capable of being rendered at multiple zoom levels</p>	<p>data in a format generated after pre-rendering that provides the layout, functionality, and design of the webpage at multiple user-selectable scaled resolutions</p> <p>“pre-rendering”: the process of blocks 150-154 of Figure 5 (<i>see</i> '353 patent, col. 15, l. 43 to col. 17, l. 41)</p>
2	<p>scalable / scaling / scaled</p> <p>'353 Patent: 33, 33 by reference to 1, 43, 48 by reference to 36, 52 by reference to 36, 58 by reference to 36, 59 by reference to 36, 66 by reference to 36, 139 by reference to 118, 183, 283</p> <p>'926 Patent: 55 by reference to 52, 59, 72 by reference to 52, 75 by reference to 52</p>	<p>capable of being rendered at multiple zoom levels / rendering at multiple zoom levels / rendered at multiple zoom levels</p>	<p>“scaling” / “scaled”: These terms have a plain and ordinary meaning and do not need to be construed. However, if the Court chooses to construe these terms, the following constructions should be used:</p> <p>“scaling”: setting to a user-selected resolution</p> <p>“scaled”: set to a user-selected resolution</p> <p>“scalable”: defined in the context of “scalable content” and “scalable vector-based content”</p>

	Term/Phrase in Issue and Asserted Claim(s) in Which Term/Phrase Appears	Plaintiff's Proposed Construction	Defendants' Proposed Construction
3	<p>translating</p> <p>'353 Patent: 33 by reference to 1, 139 by reference to 118</p> <p>'926 Patent: 30, 31 by reference to 30, 40 by reference to 30, 41 by reference to 30, 43 by reference to 30</p>	<p>This term has a plain and ordinary meaning and does not need to be construed. However, if the Court chooses to construe this term, the following construction should be used:</p> <p>converting</p>	<p>converting the format of</p>
4	<p>processing [the] HTML-based Web content to produce scalable content</p> <p>'353 Patent: 43 by reference to 36, 48 by reference to 36, 52 by reference to 36, 58 by reference to 36, 59 by reference to 36, 66 by reference to 36, 183 by reference to 149, 283 by reference to 252</p> <p>'926 Patent: 55 by reference to 52, 59 by reference to 52, 72 by reference to 52, 75 by reference to 52</p>	<p>processing [the] HTML-based Web content to produce content capable of being zoomed in or out</p>	<p>converting [the] Web content from its HTML format to a scalable content format</p>
5	<p>format</p> <p>'353 Patent: 33 by reference to 1, 139 by reference to 118</p> <p>'926 Patent: 30, 31 by reference to 30, 40 by reference to 30, 41 by reference to 30, 43 by reference to 30, 75</p>	<p>This term has a plain and ordinary meaning and does not need to be construed. However, if the Court chooses to construe this term, the following construction should be used:</p> <p>a particular way that information is encoded</p>	<p>structure of data in a file (e.g., .htm, .css, .svf, .bmp, .jpg, .gif)</p>

	Term/Phrase in Issue and Asserted Claim(s) in Which Term/Phrase Appears	Plaintiff's Proposed Construction	Defendants' Proposed Construction
6	vector-based content '353 Patent: 33, 183, 283	graphic content that includes one or more vectors	scalable content that includes, for each object, a directional data structure storing X & Y values from a single point for the page or frame to the object
7	scalable vector-based content '353 Patent: 43 '926 Patent: 30, 31 by reference to 30, 40 by reference to 30, 41 by reference to 30, 43, 59	"scalable vector-based content": graphic content that (1) is capable of being rendered at multiple zoom levels and (2) includes one or more vectors	See Defendants' proposed construction for "vector-based content."
8	vector '353 Patent: 33, 43, 58, 139 by reference to 138, 183, 283 '926 Patent: 30, 31 by reference to 30, 40 by reference to 30, 41 by reference to 30, 43, 59, 72	A mathematical expression representing a length and a direction in a two-dimensional space. In an X, Y coordinate system, a vector is represented by a value X2, Y2 relative to an origin point, represented by X1, Y1.	for each object, a directional data structure that stores X & Y values from a single point for the page or frame to the object
9	primary datum '353 Patent: 58, 59 by reference to 58, 139 by reference to 138 '926 Patent: 72	an origin point defined at an X,Y coordinate	single point defined at a fixed X,Y coordinate on the full-size web page
10	object datum '353 Patent: 58, 59 by reference to 58, 139 by reference to 138 '926 Patent: 72	reference point for an object	X,Y coordinate for an object that is defined across objects at a consistent location on the objects

	Term/Phrase in Issue and Asserted Claim(s) in Which Term/Phrase Appears	Plaintiff's Proposed Construction	Defendants' Proposed Construction
11	<p>layout location datum</p> <p>'353 Patent: 58, 59 by reference to 58, 139 by reference to 138</p> <p>'926 Patent: 72</p>	<p>one or more points corresponding to the location of the object</p>	<p>point defined at a fixed X,Y coordinate on the full-size web page that corresponds to an object</p>
12	<p>enabling the user to zoom and pan a view of the Web page</p> <p>'353 Patent: 317</p>	<p>enabling the user to zoom and move around the web page</p>	<p>using the scalable content to allow the user to resize and move around the web page</p>
13	<p>original</p> <p>original width and height of the Web page</p> <p>original page layout, functionality, and design</p>	<p>This term has a plain and ordinary meaning and does not need to be construed.</p>	<p>as viewed on a conventional desktop browser, such as Internet Explorer or Netscape Navigator, when the browser window is set to the full width resolution for which the page was designed</p> <p>the width and height of the Web page as viewed on a conventional desktop browser, such as Internet Explorer or Netscape Navigator, when the browser window is set to the full width resolution for which the page was designed</p> <p>the page layout, functionality, and design of the Web page as viewed on a conventional desktop browser, such as Internet Explorer or Netscape Navigator, when the browser window is set to the full width resolution</p>

Term/Phrase in Issue and Asserted Claim(s) in Which Term/Phrase Appears	Plaintiff's Proposed Construction	Defendants' Proposed Construction
<p style="text-align: center;">original page layout</p> <p>'353 Patent: 33 by reference to 1, 43 by reference to 36, 48 by reference to 36, 52 by reference to 36, 58, 59 by reference to 36 and 58, 66 by reference to 36, 139 by reference to 118, 183 by reference to 149, 283 by reference to 252</p> <p>'926 Patent: 30, 31 by reference to 30, 40 by reference to 30, 41 by reference to 30, 43 by reference to 30, 55 by reference to 52, 59 by reference to 52, 72, 75 by reference to 52</p>		<p>for which the page was designed</p> <p>the page layout of the Web page as viewed on a conventional desktop browser, such as Internet Explorer or Netscape Navigator, when the browser window is set to the full width resolution for which the page was designed</p>

	Term/Phrase in Issue and Asserted Claim(s) in Which Term/Phrase Appears	Plaintiff's Proposed Construction	Defendants' Proposed Construction
14	<p>fit across</p> <p>fit across the display</p> <p>rendered to fit across the display</p> <p>displayed to fit across the touch-sensitive display</p> <p>displayed to fit across at least one of a width and height of a display area of the touch-sensitive display</p> <p>'353 Patent: 33 by reference to 1, 48, 52 by reference to 51, 139 by reference to 118</p> <p>'926 Patent: 40, 41</p>	<p>This term has a plain and ordinary meaning and does not need to be construed.</p>	<p>fully fill</p> <p>fully fill the display</p> <p>rendered to fully fill the display</p> <p>displayed to fully fill the touch-sensitive display</p> <p>displayed to fully fill at least one of a width and height of a display area of the touch-sensitive display</p>
15	<p>tapping</p> <p>'353 Patent: 52, 317</p> <p>'926 Patent: 40, 41</p>	<p>This term has a plain and ordinary meaning and does not need to be construed.</p>	<p>making contact with the display using a stylus</p>

	Term/Phrase in Issue and Asserted Claim(s) in Which Term/Phrase Appears	Plaintiff's Proposed Construction	Defendants' Proposed Construction
16	<p>preserve(s) / preserved / preserving / preservation</p> <p>'353 Patent: 33 by reference to 1, 43 by reference to 36, 48 by reference to 36, 52 by reference to 36, 58 by reference to 36, 59 by reference to 36, 66, 139 by reference to 118, 183 by reference to 149, 283 by reference to 252</p> <p>'926 Patent: 30, 31 by reference to 30, 40 by reference to 30, 41 by reference to 30, 43 by reference to 30, 55 by reference to 52, 59 by reference to 52, 72 by reference to 52, 75 by reference to 52</p>	<p>This term has a plain and ordinary meaning and does not need to be construed.</p>	<p>These terms are indefinite and therefore cannot be construed.</p>
17	<p>machine-readable medium</p> <p>'353 Patent: 283</p>	<p>The machine-readable medium may include, but is not limited to, floppy diskettes, optical disks, ROMs, RAMs, EPROMs, EEPROMs, magnetic or optical cards, flash memory, or other type of media/machine-readable medium suitable for storing electronic instructions.</p>	<p>a medium for electronic instructions, such as a floppy disk, optical disk, CD-ROM, magneto-optical disk, ROM, RAM, EPROM, EEPROM, magnetic or optical card, flash memory, or carrier wave</p>

	Term/Phrase in Issue and Asserted Claim(s) in Which Term/Phrase Appears	Plaintiff's Proposed Construction	Defendants' Proposed Construction
18	<p>storage means</p> <p>'353 Patent: 33 by reference to 1</p> <p>'926 Patent: 30, 31 by reference to 30, 40 by reference to 30, 41 by reference to 30, 43 by reference to 30</p>	<p><u>Function</u>: storing a plurality of instructions</p> <p><u>Corresponding structure</u>: memory</p>	<p><u>Function</u>: storing a plurality of instructions</p> <p><u>Corresponding structure</u>: a floppy disk, optical disk, CD-ROM, magneto-optical disk, ROM, RAM, EPROM, EEPROM, magnetic or optical card, flash memory or carrier wave</p>
19	<p>processing means</p> <p>'353 Patent: 33 by reference to 1</p>	<p><u>Function</u>: processing</p> <p><u>Corresponding structure</u>: a processor, microcontroller, or logic circuitry</p>	<p><u>Function #1</u>: rendering a browser interface via which a user is enabled to request access to an original Web page, the Web page comprising HTML-based Web content having an original format defining an original width and height of the Web page and an original page layout, functionality, and design of content on the Web page</p> <p><u>Structure corresponding to Function #1</u>: Indefinite, as the specification does not disclose any algorithm for rendering a browser interface.</p> <p><u>Function #2</u>: retrieving the Web page via the wireless communication means</p> <p><u>Structure corresponding to Function #2</u>: Indefinite, as specification does not disclose any structure corresponding to "wireless communication means."</p>

Term/Phrase in Issue and Asserted Claim(s) in Which Term/Phrase Appears	Plaintiff's Proposed Construction	Defendants' Proposed Construction
		<p><u>Function #3</u>: translating at least a portion of the HTML-based Web content from its original format into scalable content that supports a scalable resolution independent representation of the Web page that preserves the original page layout, functionality and design of the content defined by its original format when scaled and rendered</p> <p><u>Structure corresponding to Function #3</u>: The algorithm depicted at boxes 150-160 of Fig. 5 (also depicted at box 114 of Fig. 2C), and '353 patent col. 3, ll. 50-52, col. 10, ll. 31-55, and col. 15, l. 43 to col. 18, l. 39.</p> <p><u>Function #4</u>: scaling the scalable content to render the Web page on the display such that a width of the Web page is rendered to fit across the display</p> <p><u>Structure corresponding to Function #4</u>: The algorithm depicted at boxes 160-172 of Fig. 6, and corresponding text at '353 patent col. 3, ll. 53-55 and col. 19, l. 14 to col. 20, l. 47, wherein the scale factor is chosen such that the width of the Web page fits across the display.</p>

	Term/Phrase in Issue and Asserted Claim(s) in Which Term/Phrase Appears	Plaintiff's Proposed Construction	Defendants' Proposed Construction
20	<p>wireless communication[s] means</p> <p>'353 Patent: 33 by reference to 1</p> <p>'926 Patent: 30, 31 by reference to 30, 40 by reference to 30, 41 by reference to 30, 43 by reference to 30, 55 by reference to 52, 59 by reference to 52, 72 by reference to 52, 75 by reference to 52</p>	<p><u>Functions:</u></p> <p>facilitate wireless communication with a network that supports access to the Internet</p> <p>facilitate communication with a mobile service provider network via which Web content may be accessed</p> <p>facilitate wireless communication with a network via which Web content may be accessed</p> <p><u>Corresponding structure:</u> an antenna</p>	<p><u>Functions (the appropriate function depends upon the specific claim):</u></p> <p>facilitating wireless communication with a network that supports access to the Internet ('353 patent claim 33 by reference to claim 1)</p> <p>facilitating communication with a mobile service provider network via which Web content may be accessed ('926 patent claim 30 and claims 31, 40, 41, and 43 by reference to claim 30)</p> <p>facilitating wireless communication with a network via which Web content may be accessed ('926 patent claims 55, 59, 72, and 75 by reference to claim 52)</p> <p><u>Corresponding structure:</u> Indefinite, as the specification does not disclose any structures corresponding to this limitation.</p>

Dated: August 31, 2012

/s/ Steven L. Caponi

Steven L. Caponi (#3484)
Blank Rome LLP
1201 Market Street, Suite 800
Wilmington, DE 19801
(302) 425-6400
caponi@blankrome.com
Attorney for Plaintiff SoftView LLC

/s/ Gregory Erich Stuhlman

Gregory Erich Stuhlman (#4765)
Greenberg Traurig, LLP
The Nemours Building
1007 North Orange Street, Suite 1200
Wilmington, DE 19801
(302) 661-7381
stuhlmang@gtlaw.com
*Attorneys for Defendants
LG Electronics, Inc., LG Electronics USA,
Inc. and LG Electronics Mobilecomm USA
Inc.*

/s/ Jack B. Blumenfeld

Jack B. Blumenfeld (#1014)
Morris, Nichols, Arshat & Tunnell LLP
1201 North Market Street
P.O. Box 1347
Wilmington, DE 19899
(302) 658-9200
jblumenfeld@mnat.com
*Attorneys for Defendant Motorola Mobility,
Inc. and AT&T Mobility LLC*

/s/ Richard L. Horwitz

Richard L. Horwitz (#2246)
David Ellis Moore (#3983)
Potter Anderson & Corroon, LLP
1313 N. Market St., Hercules Plaza, 6th Floor
P.O. Box 951
Wilmington, DE 19899-0951
(302) 984-6000
rhorwitzgpotteranderson.com
dmoore@potteranderson.com
*Attorneys for Defendants Apple Inc. and
AT&T Mobility LLC*

/s/ Anne Shea Gaza

Frederick L. Cottrell, III (#2555)
Anne Shea Gaza (#4093)
Jason J. Rawnsley (#5379)
Richards, Layton & Finger, P.A.
920 North King Street
Wilmington, DE 19801
302-651-7700
cottrell@rlf.com
gaza@rlf.com
rawnsley@rlf.com
*Attorneys for Defendants Huawei
Technologies Co., Ltd. and Futurewei
Technologies, Inc.*

/s/ Jaclyn Levy

Steven J. Fineman (#4025)
Jaclyn Levy (#5631)
Richards, Layton & Finger, PA
One Rodney Square
920 N. King Street
Wilmington, DE 19801
(302) 651-7700
fineman@rlf.com
levy@rlf.com
Attorneys for Defendant Dell Inc.

/s/ John C. Phillips, Jr.

John C. Phillips, Jr. (#110)
Megan C. Haney (#5016)
Phillips, Goldman & Spence, P.A.
1200 N. Broom Street
Wilmington, DE 19806
(302) 65504200
jcp@pgslaw.com
mch@pgslaw.com
*Attorneys for Defendant
Kyocera Corp. and Kyocera Wireless Corp*

/s/ Rodger D. Smith, II

Rodger D. Smith, II (#3778)
Morris, Nichols, Arshnt & Tunnell LLP
1201 North Market Street
P.O. Box 1347
Wilmington, DE 19899
(302) 351-9205
rsmith@mnat.com
*Attorney for Defendant Sony Ericsson Mobile
Communications (USA) Inc.*

/s/ Adam Wyatt Poff

Adam Wyatt Poff (#3990)
Monte Terrell Squire (#4764)
Young, Conaway, Stargatt & Taylor LLP
Rodney Square
1000 North King Street
Wilmington, DE 19801
(302) 571-6600
apoff@ycst.com
msquire@ycst.com
*Attorneys for Defendants Samsung
Electronics Co. Ltd, Samsung Electronics
America, Inc., and Samsung
Telecommunications America LLC*

/s/ John W. Shaw

John W. Shaw (#3362)
Karen E. Keller (#4489)
Andrew E. Russell (#5382)
Shaw Keller LLP
300 Delaware Avenue, Suite 1120
Wilmington, DE. 19801
(302) 298-0700
jshaw@shawkeller.com
kkeller@shawkeller.com
arussell@shawkeller.com
*Attorney for Defendants HTC Corp., HTC
America Inc.*

APPENDIX A—PLAINTIFF’S CITATIONS TO INTRINSIC EVIDENCE

Term/Phrase in Issue and Asserted Claim(s) in Which Term/Phrase Appears	Plaintiff’s Proposed Construction	Plaintiff’s Citations to Intrinsic Evidence
<p>1</p> <p>scalable content</p> <p>‘353 Patent: 33, 43, 48 by reference to 36, 52 by reference to 36, 58 by reference to 36, 59 by reference to 36, 66 by reference to 36, 139 by reference to 118, 183, 283</p> <p>‘926 Patent: 55 by reference to 52, 59, 72 by reference to 52, 75 by reference to 52</p>	<p>graphic content capable of being rendered at multiple zoom levels</p>	<p>‘353 Patent (Ex. 1) Figures: 2A, 2B, 2C, 4A, 4B, 4C, 4D, 4E, 4F, 4G, 6, 7A, 7B, 8A, 8B, 9B Specification: 1:55-60; 2:24-27; 2:37-44; 3:33-36; 3:37-40; 3:62-66; 6:50-56; 8:47-53; 8:61-9:3; 10:9-16; 10:18-20; 10:44-45; 12:7-16; 15:41-50; 18:45-46; 18:66-19:3; 19:27-19:31; 19:39-56; 20:11-17; 20:17-27; 20:29-44; 20:41-45; 21:10-13 Claims: 1, 4, 6, 10, 23, 28, 33, 36, 42, 43, 47, 57, 62, 70, 80, 84, 112, 118, 121, 123, 137, 142, 147, 149, 155, 156, 160, 173, 178, 183, 209, 216, 219, 221, 225, 238, 243, 248, 252, 258, 259, 263, 273, 278, 283, 311</p> <p>‘926 Patent (Ex. 2) Figures: 2A, 2B, 2C, 4A, 4B, 4C, 4D, 4E, 4F, 4G, 6, 7A, 7B, 8A, 8B, 9B Specification: 1:55-60; 2:24-27; 2:37-44; 3:33-36; 3:37-40; 3:62-66; 6:50-56; 8:47-53; 8:61-9:3; 10:9-16; 10:18-20; 10:44-45; 12:7-16; 15:41-50; 18:45-46; 18:66-19:3; 19:27-19:31; 19:39-56; 20:11-17; 20:17-27; 20:29-44; 20:41-45; 21:10-13 Claims: 51, 52, 58, 59, 63, 71, 74, 79, 88</p> <p>File History for U.S. Patent No. 7,461,353 (Ex. 3E) FH000752-53; FH000920; FH000928; FH000933; FH001042-49; FH001057; FH001105; FH001107-09; FH001110; FH001111; FH001115; FH001208; FH001349; FH001389</p> <p>File History For U.S. Patent No. 7,831,926 (Ex. 3F) FH003045; FH003523; FH003538; FH003540-54; FH003577</p>
<p>2</p> <p>scalable / scaling / scaled</p> <p>‘353 Patent: 33, 33 by reference to 1, 43, 48 by reference to 36, 52 by reference to 36, 58 by reference to 36, 59 by reference to 36, 139</p>	<p>capable of being rendered at multiple zoom levels / rendering at multiple zoom levels / rendered at multiple zoom levels</p>	<p>‘353 Patent (Ex. 1) Figures: 1C, 2A, 2B, 2C, 4A, 4E, 6, 7A, 7B, 8A, 8B, 9A, 9B Specification: Abstract; 1:55-60; 2:24-27; 2:37-44; 3:14-65; 6:50-56; 8:47-53; 8:61-9:3; 10:9-16; 10:18-20; 10:44-45; 12:7-16; 15:41-50; 18:45-46; 18:66-19:3; 19:14-18; 19:27-19:31; 19:39-56; 20:11-17; 20:17-27; 20:29-44; 20:41-45; 21:10-13 Claims: 1, 4, 6, 10, 23, 27, 28, 33, 36, 42, 43, 47, 57, 61, 62, 63, 70, 76, 80, 84, 112, 118, 121, 123, 127, 137, 141, 142, 147, 160, 177, 178, 179, 183, 209, 216, 219,</p>

Term/Phrase in Issue and Asserted Claim(s) in Which Term/Phrase Appears	Plaintiff's Proposed Construction	Plaintiff's Citations to Intrinsic Evidence
<p>by reference to 118, 183, 283</p> <p>'926 Patent: 55 by reference to 52, 59, 72 by reference to 52, 75 by reference to 52</p>		<p>221, 225, 238, 242, 243, 248, 252, 258, 259, 263, 273, 277, 278, 279, 282, 311</p> <p>'926 Patent (Ex. 2) Figures: 1C, 2A, 2B, 2C, 4A, 4E, 6, 7A, 7B, 8A, 8B, 9A, 9B Specification: Abstract; 1:55-60; 2:24-27; 2:37-44; 3:14-65; 6:50-56; 8:47-53; 8:61-9:3; 10:9-16; 10:18-20; 10:44-45; 12:7-16; 15:41-50; 18:45-46; 18:66-19:3; 19:14-18; 19:27-19:31; 19:39-56; 20:11-17; 20:17-27; 20:29-44; 20:41-45; 21:10-13 Claims: 1, 6, 7, 11, 19, 21, 23, 24, 25, 26, 30, 33, 35, 39, 42, 43, 46, 48, 49, 51, 52, 58, 59, 63, 70, 71, 74, 75, 79, 88</p> <p>File History for U.S. Patent No. 7,461,353 (Ex. 3E) FH000643-44 ¶ 0003; FH000917-20; FH00919-20; FH000923; FH000924; FH001046-49; FH001101-06;</p> <p>File History For U.S. Patent No. 7,831,926 (Ex. 3F) FH003045; FH003523; FH003538; FH003540-54; FH003577</p>
<p>3</p> <p>translating</p> <p>'353 Patent: 33 by reference to 1, 139 by reference to 118</p> <p>'926 Patent: 30, 31 by reference to 30, 40 by reference to 30, 41 by reference to 30, 43 by reference to 30</p>	<p>This term has a plain and ordinary meaning and does not need to be construed. However, if the Court chooses to construe this term, the following construction should be used:</p> <p>converting</p>	<p>'353 Patent (Ex. 1) Figures: 1A, 1B, 1C, 2A, 2B, 2C, 3 Specification: 1:43-50; 3:4-10; 3:11-14; 3:15-18; 3:19-21; 3:22-24; 3:25-27; 3:28-30; 7:1-4; 8:63-65; 10:21-31; 10:50-56; 11:46-53; 12:12-17 Claims 1; 4; 112; 118; 121; 209, 216; 219; 311</p> <p>'926 Patent (Ex. 2) Figures: 1A, 1B, 1C, 2A, 2B, 2C, 3; Specification: 2:61-67; 3:1-4; 3:5-8; 3:9-11; 3:12-14; 3:15-17; 3:18-20; 6:57-60; 8:50-55; 10:11-20; 10:39-45; 10:59-65; 11:35-42; 12:7-12; 15:43-45 Claims 1, 6, 21, 30, 33</p> <p>Provisional App. No. 60/211,019 (Ex. 3A) FH000006, lines 12-15; FH000009, lines 6-8; FH00010, Figure 1</p> <p>Provisional App. No. 60/217,345 (Ex. 3B)</p>

Term/Phrase in Issue and Asserted Claim(s) in Which Term/Phrase Appears	Plaintiff's Proposed Construction	Plaintiff's Citations to Intrinsic Evidence
		<p>FH000017, lines 18-22; 35-38; 45-48; FH000018, lines 32-34; FH000021, Figure 1</p> <p>File History for U.S. Patent No. 7,461,353 (Ex. 3E) FH000653, at ¶ [0049]; FH000656, at ¶ [0056]; FH000682, at lines 4-6; FH001389</p> <p>File History for U.S. Patent No. 7,831,926 (Ex. 3F) FH003000, at ¶ [0049]; FH003003, at ¶ [0056]; FH002993, at ¶ [0015];</p>
<p>4 processing [the] HTML-based Web content to produce scalable content</p> <p>'353 Patent: 43 by reference to 36, 48 by reference to 36, 52 by reference to 36, 58 by reference to 36, 66 by reference to 36, 183 by reference to 149, 283 by reference to 252</p> <p>'926 Patent: 55 by reference to 52, 59 by reference to 52, 72 by reference to 52, 75 by reference to 52</p>	<p>processing [the] HTML-based Web content to produce content capable of being zoomed in or out</p>	<p>'353 Patent (Ex. 1) Figures: 2A, 2B, 2C, 4A, 4B, 4C, 4D, 4E, 4F, 4G, 6, 7A, 7B, 8A, 8B, 9B Specification: Abstract: 1:55-60; 2:24-29; 2:37-44; 3:33-36; 3:37-40; 3:62-66; 6:50-56; 8:33-44; 8:47-53; 8:61-9:3; 10:9-16; 10:18-20; 10:11-20; 10:44-45; 12:7-16; 15:41-50; 15:55-64; 18:40-46; 18:47-19:3; 19:14-31; 19:39-56; 20:11-17; 20:17-27; 20:29-44; 20:41-45; 21:10-13 Claims: 1, 4, 6, 10, 23, 28, 33, 36, 42, 43, 47, 57, 62, 70, 80, 84, 112, 118, 121, 123, 137, 142, 147, 149, 155, 156, 160, 173, 178, 183, 209, 216, 219, 221, 225, 238, 243, 248, 252, 258, 259, 263, 273, 278, 283, 311</p> <p>'926 Patent (Ex. 2) Figures: 2A, 2B, 2C, 4A, 4B, 4C, 4D, 4E, 4F, 4G, 6, 7A, 7B, 8A, 8B, 9B Specification: Abstract: 1:55-60; 2:24-29; 2:37-44; 3:33-36; 3:37-40; 3:62-66; 6:50-56; 8:33-44; 8:47-53; 8:61-9:3; 10:9-16; 10:18-20; 10:11-20; 10:44-45; 12:7-16; 15:41-50; 15:55-64; 18:40-46; 18:47-19:3; 19:14-31; 19:39-56; 20:11-17; 20:17-27; 20:29-44; 20:41-45; 21:10-13 Claims: 51, 52, 58, 59, 63, 71, 74, 79, 88</p> <p>File History for U.S. Patent No. 7,461,353 (Ex. 3E) FH000752-53; FH000920; FH000928; FH000933; FH001042-49; FH001057; FH001105; FH001107-09; FH001110; FH001111; FH001115; FH001208; FH001349; FH001389</p> <p>File History For U.S. Patent No. 7,831,926 (Ex. 3F) FH003045; FH003523; FH003538; FH003540-54; FH003577</p>

Term/Phrase in Issue and Asserted Claim(s) in Which Term/Phrase Appears	Plaintiff's Proposed Construction	Plaintiff's Citations to Intrinsic Evidence
<p>5 format</p> <p>'353 Patent: 33 by reference to 1, 139 by reference to 118</p> <p>'926 Patent: 30, 31 by reference to 30, 40 by reference to 30, 41 by reference to 30, 43 by reference to 30, 75</p>	<p>This term has a plain and ordinary meaning and does not need to be construed. However, if the Court chooses to construe this term, the following construction should be used:</p> <p>a particular way that information is encoded</p>	<p>'353 Patent (Ex. 1) Figures: 2A, 2B, 2C Specification: 1:61-2:3; 2:57-61; 3:19-27; 4:50-53; 6:6-9; 8:14-18; 8:64-65; 9:32-43; 9:52-56; 10:2-8; 10:21-26; 10:62-65; 11:3-9; 11:41-58</p> <p>'926 Patent (Ex. 2) Figures: 2A, 2B, 2C Specification: 2:4-9; 3:9-17; 4:42-45; 5:63-66; 8:1-7; 8:54-56; 9:22-31; 9:42-46; 9:59-65; 10:11-16; 10:51-54; 10:59-65; 11:30-42</p> <p>Provisional App. No. 60/211,019 (Ex. 3A) FH000006 at lines 12-16; FH000009 at lines 6-8</p> <p>Provisional App. No. 60/217,345 (Ex. 3B) FH000022 (Figure 2); FH000024 (Figure 4); FH000016 at lines 3-5, 45-46; FH000017 at lines 18-22, 35-48; FH000018 at lines 51-53</p> <p>File History for U.S. Patent App. No. 09/828,511 (Ex. 3C) FH000042; FH000045 ¶ [0008]; FH000046 ¶ [0016], [0017]; FH000047 ¶ [0018]; FH000061-063 at claim 6, 7, 9-12, 16; FH000065-066 at claims 27-29; FH000117</p> <p>File History for U.S. Patent No. 7,210,099 (Ex. 3D) FH000185-194 at claims 1, 3, 4, 5, 6, 9, 11, 12, 15-17, 19, 21-23, 25, 26, 27, 29-32, 34-36, 38</p> <p>File History for U.S. Patent No. 7,461,353 (Ex. 3E) FH000677-681 at claims 1, 3-7, 9-11, 13-15; FH000682; FH000729 at claim 20; FH000754 at claim 40; FH000760 at claim 66; FH000761 at claim 71; FH000766 at claim 93; FH000772 at claim 123</p> <p>File History for U.S. Patent No. 7,831,926 (Ex. 3F) FH003025 at claim 1; FH003030-031 at claims 25, 27; FH003036 at claim 49;</p>

Term/Phrase in Issue and Asserted Claim(s) in Which Term/Phrase Appears	Plaintiff's Proposed Construction	Plaintiff's Citations to Intrinsic Evidence
		FH003042 at claim 75
6 vector-based content '353 Patent: 33, 183, 283	content that includes one or more vectors	<p>'353 Patent (Ex. 1) Figures: 1C, 2A, 2B, 2C, 4A, 4B, 4C, 4D, 4F, 4G, 5, 6 Specification: 1:43-45; 3:14-55; 4:2-8; 5:65-5:3; 6:7-10; 8:57-60; 10:22-31; 10:34-56; 12:12-20; 15:42-55; 17:65-18:16; 18:18-32; 19:14-18; 19:20-31; 19:57-63; 20:8-17; 21:1-12 Claims: 33, 43, 84, 137, 147, 183, 248, 283</p> <p>'926 Patent (Ex. 2) Figures: 1C, 2A, 2B, 2C, 4A, 4B, 4C, 4D, 4F, 4G, 5, 6 Specification: 1:43-45; 3:14-55; 4:2-8; 5:65-5:3; 6:7-10; 8:57-60; 10:22-31; 10:34-56; 12:12-20; 15:42-55; 17:65-18:16; 18:18-32; 19:14-18; 19:20-31; 19:57-63; 20:8-17; 21:1-12</p> <p>File History for U.S. Patent No. 7,461,353 (Ex. 3E) FH000643-45; FH000722-23; FH00917-28; FH001045-46; FH001058; FH001111</p> <p>File History For U.S. Patent No. 7,831,926 (Ex. 3F) FH003170; FH003382; FH003578-79; FH003589-97</p>
7 scalable vector-based content '353 Patent: 43 '926 Patent: 30, 31 by reference to 30, 40 by reference to 30, 41 by reference to 30, 43, 59	graphic content that (1) is capable of being rendered at multiple zoom levels and (2) includes one or more vectors	<p>'353 Patent (Ex. 1) Figures: 1C, 2A, 2B, 2C, 4A, 4B, 4C, 4D, 4E, 4F, 4G, 5, 6, 7A, 7B, 8A, 8B, 9B Specification: 1:43-45; 1:55-60; 2:24-27; 2:37-44; 3:33-36; 3:37-40; 3:62-66; 4:2-8; 5:65-5:3; 6:7-10; 6:50-56; 8:47-53; 8:57-60; 8:61-9:3; 10:9-16; 10:18-20; 10:22-31; 10:34-56; 12:7-20; 15:41-55; 18:18-32; 18:45-46; 18:66-19:3; 19:27-19:31; 19:39-56; 19:57-63; 20:11-17; 20:17-27; 20:29-44; 20:41-45; 21:1-13 Claims: 43, 156, 259</p> <p>'926 Patent (Ex. 2) Figures: 1C, 2A, 2B, 2C, 4A, 4B, 4C, 4D, 4E, 4F, 4G, 5, 6, 7A, 7B, 8A, 8B, 9B Specification: 1:43-45; 1:55-60; 2:24-27; 2:37-44; 3:33-36; 3:37-40; 3:62-66; 4:2-8; 5:65-5:3; 6:7-10; 6:50-56; 8:47-53; 8:57-60; 8:61-9:3; 10:9-16; 10:18-20; 10:22-31;</p>

Term/Phrase in Issue and Asserted Claim(s) in Which Term/Phrase Appears	Plaintiff's Proposed Construction	Plaintiff's Citations to Intrinsic Evidence
		<p>10:34-56; 12:7-20; 15:41-55; 18:18-32; 18:45-46; 18:66-19:3; 19:27-19:31; 19:39-56; 19:57-63; 20:11-17; 20:17-27; 20:29-44; 20:41-45; 21:1-13</p> <p>Claims: 1, 6, 7, 11, 19, 21, 26, 30, 33, 35, 39, 43, 48, 51, 59</p> <p>File History for U.S. Patent No. 7,461,353 (Ex. 3E) FH000643-45; FH000722-23; FH000752-53; FH000917-28; FH000933; FH001042-49; FH001057; FH001058; FH001105; FH001107-09; FH001110; FH001111; FH001115; FH001208; FH001349; FH001389</p> <p>File History For U.S. Patent No. 7,831,926 (Ex. 3F) FH003170; FH003382; FH003578-79; FH003589-97</p>
<p>8</p> <p>vector</p> <p>'353 Patent: 33, 43, 58, 139 by reference to 138, 183, 283</p> <p>'926 Patent: 30, 31 by reference to 30, 40 by reference to 30, 41 by reference to 30, 43, 59, 72</p>	<p>A mathematical expression representing a length and a direction in a two-dimensional space. In an X, Y coordinate system, a vector is represented by a value X2, Y2 relative to an origin point, represented by X1, Y1.</p>	<p>'353 Patent (Ex. 1) Figures: 1C, 2A, 2B, 2C, 4A, 4B, 4C, 4D, 4F, 4G, 5, 6 Specification: 1:43-45; 3:15-55; 4:2-8; 5:65-5:3; 6:7-10; 8:57-60; 10:22-31; 10:34-56; 12:12-20; 15:42-55; 17:65-18:16; 18:18-32; 19:14-18; 19:57-63; 20:8-17; 21:1-12 Claims: 5, 23, 24, 26, 33, 43, 58, 60, 84, 122, 137, 138, 140, 147, 156, 174, 176, 183, 219, 220, 238, 241, 248, 259, 274, 276, 283.</p> <p>'926 Patent (Ex. 2) Figures: 1C, 2A, 2B, 2C, 4A, 4B, 4C, 4D, 4F, 4G, 5, 6 Specification: 1:43-45; 3:15-55; 4:2-8; 5:65-5:3; 6:7-10; 8:57-60; 10:22-31; 10:34-56; 12:12-20; 15:42-55; 17:65-18:16; 18:18-32; 19:14-18; 19:57-63; 20:8-17; 21:1-12 Claims: 1, 6, 7, 11, 15, 16, 17, 19, 21, 22, 26, 30, 33, 34, 35, 39, 43, 44, 45, 48, 51, 59, 69, 72</p> <p>File History for U.S. Patent No. 7,461,353 (Ex. 3E) FH000643-45; FH000722-23; FH00917-28; FH001045-46; FH001058; FH001111</p> <p>File History For U.S. Patent No. 7,831,926 (Ex. 3F) FH003170; FH003382; FH003578-79; FH003589-97</p>

Term/Phrase in Issue and Asserted Claim(s) in Which Term/Phrase Appears	Plaintiff's Proposed Construction	Plaintiff's Citations to Intrinsic Evidence
<p>9</p> <p>primary datum</p> <p>'353 Patent: 58, 59 by reference to 58, 139 by reference to 138</p> <p>'926 Patent: 72</p>	<p>an origin point defined at an X,Y coordinate</p>	<p>'353 Patent (Ex. 1) Figures: 4A, 4B, 4C, 4D, 4E, 4F, 4G, 5, 6 Specification: 3:31-55; 17:45-18:16; 19:23-31; 19:63-20:1; 20:2-8; Claims: 5, 24, 58, 122, 138, 174, 220, 239, 274</p> <p>'926 Patent (Ex. 2) Figures: 4A, 4B, 4C, 4D, 4E, 4F, 4G, 5, 6 Specification: 3:31-55; 17:45-18:16; 19:23-31; 19:63-20:1; 20:2-8; Claims: 16, 22, 34, 44, 72.</p> <p>File History for U.S. Patent No. 7,461,353 (Ex. 3E) FH000927 Claim 75; FH001297 Claim 184</p> <p>File History For U.S. Patent No. 7,831,926 (Ex. 3F) FH003016</p>
<p>10</p> <p>object datum</p> <p>'353 Patent: 58, 59 by reference to 58, 139 by reference to 138</p> <p>'926 Patent: 72</p>	<p>reference point for an object</p>	<p>'353 Patent (Ex. 1) Figures: 4A, 4B, 4C, 4D, 4E, 4F, 4G Specification: 3:31-48; 17:46-53; 17:57-59; 17:65-18:10; 20:8-17. Claims: 5, 24, 58, 122, 138, 174, 220, 239, 274</p> <p>'926 Patent (Ex. 2) Figures: 4B, 4C, 4D, 4F and 4G Specification: 3:31-48; 17:46-53; 17:57-59; 17:65-18:10; 20:8-17. Claims: 16, 22, 34, 44, 72</p> <p>File History for U.S. Patent No. 7,461,353 (Ex. 3E) FH000668 ¶ 0084; FH000668-69 ¶ 0085; FH000680 ¶ 8; FH000927 at claim 75</p> <p>File History For U.S. Patent No. 7,831,926 (Ex. 3F) FH003016; FH003382; FH003587; FH003593</p>

Term/Phrase in Issue and Asserted Claim(s) in Which Term/Phrase Appears	Proposed Construction	Plaintiff's Citations to Intrinsic Evidence
<p>11</p> <p>layout location datum</p> <p>'353 Patent: 58, 59 by reference to 58, 139 by reference to 138</p> <p>'926 Patent: 72</p>	<p>one or more points corresponding to the location of the object</p>	<p>'353 Patent (Ex. 1) Figures: 4A, 4B, 4C, 4D, 4E, 4F, 4G Specification: 3:29-50; 17:46-53; 17:57-59; 17:65-18:10; 20:8-17. Claims: 5, 24, 58, 138, 174, 239, 274</p> <p>'926 Patent (Ex. 2) Figures: 4A, 4B, 4C, 4D, 4E, 4F, 4G Specification: 3:29-50; 17:46-53; 17:57-59; 17:65-18:10; 20:8-17. Claims: 16, 22, 26, 38, 44, 72</p> <p>File History for U.S. Patent No. 7,461,353 (Ex. 3E) FH000668 ¶ 0084; FH000668-69 ¶ 0085; FH000680 ¶ 8; FH000927 at claim 75</p> <p>File History For U.S. Patent No. 7,831,926 (Ex. 3F) FH003016; FH003382; FH003587; FH003593</p>
<p>12</p> <p>enabling the user to zoom and pan a view of the Web page</p> <p>'353 Patent: 317</p>	<p>enabling the user to zoom and move around the web page</p>	<p>'353 patent (Ex. 1) Figures: 4A, 4E, 6, 7A, 7B, 8A, 8B, 9A, 9B Specification: Abstract; 2:22-26; 2:40-46; 3:56-67; 4:3-8; 4:67-5:9; 9:4-12; 19:14-31; 20:56-67 Claims 2, 15, 18-21, 27, 39, 51, 53-55, 61, 73, 78, 79, 92, 95-98, 119, 131, 133, 141, 149, 152, 165, 168, 177, 185, 193, 217, 233, 242, 255, 269, 277, 293, 317, 318</p> <p>'926 patent (Ex. 2) Figures 4A, 4E, 6, 7A, 7B, 8A, 8B, 9A, 9B Specification: 2:31-36; 2:44-49; 3:62-65; 4:59-5:1; 9:59-9:2; 19:20-31 Claims: 4, 14, 23, 31, 41, 42, 46, 55, 65, 70, 82</p> <p>U.S. Provisional Patent App. No. 60/211,019 (Ex. 3A) FH000007 at 14-18, 29-33</p> <p>U.S. Provisional Patent App. No. 60/217,345 (Ex. 3B) FH000018 at 3:15-17, 3:34-38</p>

Term/Phrase in Issue and Asserted Claim(s) in Which Term/Phrase Appears	Plaintiff's Proposed Construction	Plaintiff's Citations to Intrinsic Evidence
		<p>File History for U.S. Patent App. No. 09/828,511 (Ex. 3C) FH000053 ¶ [0035]; FH000054 ¶ [0036]</p> <p>File History for U.S. Patent No. 7,210,099 (Ex. 3D) FH000195; FH000616 ¶ [0007]</p> <p>File History for U.S. Patent No. 7,461,353 (Ex. 3E) FH000682</p>
<p>13 original</p> <p>original width and height of the Web page</p> <p>original page layout, functionality, and design</p> <p>original page layout</p> <p>'353 Patent: 33 by reference to 1, 43 by reference to 36, 48 by reference to 36, 52 by reference to 36, 58, 59 by reference to 36, and 58, 66 by reference to 36, 139 by reference to 118, 183 by reference to 149, 283 by reference to 252</p> <p>'926 Patent: 30, 31 by reference to 30, 40 by reference to 30, 41 by reference to 30, 43 by reference to 30, 55 by reference</p>	<p>This term has a plain and ordinary meaning and does not need to be construed.</p>	<p>'353 Patent (Ex. 1) Figures 4A-9B Specification: 1:61-18; 2:23-35; 2:47-59; 3:31-64; 4:64-53; 5:10-11; 5:16-18; 5:67-63; 6:23-31; 7:28-32; 7:38-48; 7:56-8:44; 9:4-12; 10:22-31; 10:39-50; 11:3-9; 12:12-15:42; 15:50-16:67; 17:8-30; 17:42-45; 18:33-39; 18:47-19:3; 19:28-31; 20:11-32; 20:54-55; 20:64-67; 21:1-10</p> <p>'926 Patent (Ex. 2) Figures 4A-9B Specification: 3:21-55; 4:46-62; 5:2-3; 5:7-9; 5:57-60; 6:12-14; 7:17-21; 7:27-37; 7:45-8:34; 8:61-9:3; 10:11-20; 10:29-39; 10:59-65; 12:7-15:42; 15:50-16:67; 17:8-30; 17:42-45; 18:33-39; 18:47-19:3; 19:28-31; 20:11-32; 20:54-55; 20:64-67; 21:1-10</p> <p>Provisional App. No. 60/211,019 (Ex. 3A) FH000005 at 22-30; FH000008 at 7-15</p> <p>File History for U.S. Patent App. No. 09/828,511 (Ex. 3C) FH000042; FH000045</p> <p>File History for U.S. Patent No. 7,210,099 (Ex. 3D) FH000149 ¶ [0007], [0008]; FH000195</p>

Term/Phrase in Issue and Asserted Claim(s) in Which Term/Phrase Appears	Proposed Construction	Plaintiff's Citations to Intrinsic Evidence
to 52, 59 by reference to 52, 72, 75 by reference to 52		<p>File History for U.S. Patent No. 7,461,353 (Ex. 3E) FH000645; FH000682; FH000983; FH001042; FH001046-048; FH001060-061; FH001065-081; FH001100-111; FH001211-212; FH001348-366</p> <p>File History for U.S. Patent No. 7,831,926 (Ex. 3F) FH002992; FH003381-382; FH003450-451; FH003530; FH003538-540; FH003542-543; FH003553-554; FH003562-582</p>
14 fit across fit across the display rendered to fit across the display displayed to fit across the touch-sensitive display displayed to fit across at least one of a width and height of a display area of the touch-sensitive display '353 Patent: 33 by reference to 1, 48, 52 by reference to 51, 139 by reference to 118 '926 Patent: 40, 41	This term has a plain and ordinary meaning and does not need to be construed.	<p>'353 Patent (Ex. 1) Figures: 6, 7A, 7B, 8A, 8B, 9A, 9B Specification: 3:53-55; 3:56-64; 18:33-39; 19:63-20:6; 20:46-48; 21:1-13 Claims: 1, 14, 17, 21, 48, 53, 63, 78, 87, 91, 94, 98, 118, 128, 131, 133, 164, 167, 170, 189, 192, 196, 216, 228, 232, 236, 264, 267, 269, 285, 289, 292, 296</p> <p>'926 Patent (Ex. 2) Figures: 6, 7A, 7B, 8A, 8B, 9A, 9B Specification: 3:44-55; 18:33-39; 19:64-20:7; 20:46-48; 21:1-14 Claims: 12, 13, 14, 40, 41, 87</p> <p>File History for U.S. Patent No. 7,461,353 (Ex. 3E) FH001052; FH001339-1346</p>
15 tapping '353 Patent: 52, 317	This term has a plain and ordinary meaning and does not need to be construed.	<p>'353 Patent (Ex. 1) Figures: 7A, 7B, 8A, 8B, 9A, 9B Specification: 2:39-46; 3:56-64; 20:56-67 Claims: 13, 16, 20, 50, 52, 55, 79, 90, 93, 97, 129, 132, 134, 162, 166, 171, 187,</p>

Term/Phrase in Issue and Asserted Claim(s) in Which Term/Phrase Appears	Plaintiff's Proposed Construction	Plaintiff's Citations to Intrinsic Evidence
<p>'926 Patent: 40, 41</p>		<p>191, 194, 227, 231, 234, 265, 268, 270, 287, 291, 294, 317</p> <p>'926 Patent (Ex. 2) Figures: 7A, 7B, 8A, 8B, 9A, 9B Specification: 2:44-49; 3:47-55; 20:58-67 Claims 12, 13, 14, 40, 41, 42, 64, 65, 66, 87</p>
<p>16 preserve(s) / preserved / preserving / preservation</p> <p>'353 Patent: 33 by reference to 1, 43 by reference to 36, 48 by reference to 36, 52 by reference to 36, 58 by reference to 36, 59 by reference to 36, 66, 139 by reference to 118, 183 by reference to 149, 283 by reference to 252</p> <p>'926 Patent: 30, 31 by reference to 30, 40 by reference to 30, 41 by reference to 30, 43 by reference to 30, 55 by reference to 52, 59 by reference to 52, 72 by reference to 52, 75 by reference to 52</p>	<p>This term has a plain and ordinary meaning and does not need to be construed.</p>	<p>'353 Patent (Ex 1) Figures: 4A-9B Specification: 2:4-18; 2:23-35; 2:47-59; 3:31-64; 4:64-5:3; 5:10-11; 5:16-18; 5:20-24; 5:67-6:3; 10:22-31 18:33-39; 18:47-19:3; 20:18-32; 20:54-55; 20:64-67; 21:1-13 Claims: 1, 6, 35, 36, 66, 69, 70, 87, 106, 111, 112, 118, 123, 149, 182, 184, 185, 204, 209, 215, 216, 221, 250, 252, 282, 285, 310, 311, 304, 318</p> <p>'926 Patent (Ex. 2) Figures 4A-9B; 3:21-55 Specification: 2:14-29; 2:33-41; 3:31-64; 4:56-60; 5:2-15; 5:56-60; 10:11-20 Claims: 1, 30, 52, 79</p> <p>Provisional App. No. 60/211,019 (Ex. 3A) FH000005 at lines 45-47; FH000008 at lines 17-24</p> <p>Provisional App. No. 60/217,345 (Ex. 3B) FH000019 at 18-25</p> <p>File History for U.S. Patent App. No. 09/828,511 (Ex. 3C) FH000042; FH000045</p> <p>File History for U.S. Patent No. 7,210,099 (Ex. 3D) FH000149 ¶¶ [0007], [0008]; FH000185 at claim 1; FH000186-187 at claim 9; FH000188 at claim 14; FH000189-190 at claim 21; FH000191 at claim 25; FH000192 at claim 30; FH000193-194 at claim 34; FH000195</p>

Term/Phrase in Issue and Asserted Claim(s) in Which Term/Phrase Appears	Plaintiff's Proposed Construction	Plaintiff's Citations to Intrinsic Evidence
		<p>File History for U.S. Patent No. 7,461,353 (Ex. 3E) FH000645; FH000677 at claim 1; FH000679 at claim 9; FH000680 at claim 13; FH000682; FH000755 at claim 44; FH000756 at claim 50; FH000757 at claims 52, 55; FH000758 at claim 57; FH000760 at claim 66; FH000761 at claims 71, 73; FH00763 at claims 79, 81; FH000764 at claims 83, 84, FH000766 at claim 93; FH000768 at claim 104; FH000769 at claims 109, 111; FH000770 at claims 113, 114; FH000772 at claim 123; FH000774 at claims 133, 135; FH000775 at claim 138; FH000983; FH001041-061; FH001068-081; FH001101-104; FH001149 at claim 71; FH001150 at claim 76; FH001155 at claim 99; FH001161 at claim 128; FH001163 at claim 143; FH001168 at claim 174; FH001169 at claim 180; FH001170 at claim 185; FH001175-176 at claim 211; FH001180 at claim 244; FH001183 at claim 265; FH001184 at claim 271; FH001185-86 at claim 276; FH001190-191 at claim 303; FH001197 at claim 337; FH001200 at claim 359; FH001203-204 at claims 383-390; FH001206 at claims 392, 393; FH001209-212; FH001258-159; FH001274-367</p> <p>File History for U.S. Patent No. 7,831,926 (Ex. 3F) FH003028 at claims 13, 15, 16; FH003030-032 at claims 25, 27, 29; FH003033-034 at claim 37, 39, 40, FH003036 at claim 49; FH003039 at claims 63, 65; FH003039-040 at claim 66; FH003042 at claim 75; FH003044 at claim 87; FH003424-426; FH003432; FH003438-439; FH003445; FH003451; FH003530; FH003537-596</p> <p>U.S. Patent No. 7,584,423 (Ex. 3G) Claims: 1, 8, 9, 20, 21, 24, 25, 27, 36, 37, 45, 46, 47</p> <p>U.S. Patent No. 7,844,889 (Ex. 3H) Claims 1, 42, 77, 108</p> <p>U.S. Patent No. 7,823,083 (Ex. 3I) Claims 1, 25, 26, 52, 53, 79, 80</p>

Term/Phrase in Issue and Asserted Claim(s) in Which Term/Phrase Appears	Plaintiff's Proposed Construction	Plaintiff's Citations to Intrinsic Evidence
		<p>U.S. Patent No. 8,145,995 (Ex. 3J) Claims 1, 13, 14</p>
<p>17 machine-readable medium '353 Patent: 283</p>	<p>The machine-readable medium may include, but is not limited to, floppy diskettes, optical disks, ROMs, RAMs, EPROMs, EEPROMs, magnetic or optical cards, flash memory, or other type of media/machine-readable medium suitable for storing electronic instructions.</p>	<p>'353 Patent (Ex. 1) 4:29-34</p> <p>'926 Patent (Ex. 2) 4:21-26</p>
<p>18 storage means '353 Patent: 33 by reference to 1 '926 Patent: 30, 31 by reference to 30, 40 by reference to 30, 41 by reference to 30, 43 by reference to 30</p>	<p><u>Function</u>: storing a plurality of instructions</p> <p><u>Corresponding structure</u>: memory</p>	<p>'353 Patent (Ex. 1) Figures: 1A, 1C, 2A, 2C, 3, 5, 6, 10 Specification: 3:4-10; 3:15-18; 3:19-21; 3:25-27; 3:28-30; 3:50-52; 3:53-55; 3:65-67; 4:29-34; 5:29-32; 19:14-20; 21:19-32; 21:33-35; 22:1-6 Claims: 1, 36, 70, 71, 72, 87, 112, 317</p> <p>'926 Patent (Ex. 2) Figures: 1A, 1C, 2A, 2C, 3, 5, 6, 10 Specification: 2:61-67; 3:5-8; 3:9-11; 3:15-17; 3:18-20; 3:41-43; 3:44-46; 3:56-58; 4:21-26; 5:20-23; 19:14-20; 21: 20-33; 21:34-36; 22:3-8; Claims: 1, 30, 52, 79</p>
<p>19 processing means '353 Patent: 33 by reference to 1</p>	<p><u>Function</u>: processing</p> <p><u>Corresponding structure</u>: a processor, microcontroller, or logic circuitry</p>	<p>'353 Patent (Ex. 1) Figures: 1C, 2C, 3, 4A, 4B, 4C, 4D, 4E, 4F, 5, 6, 10 Specification: 3:15-18; 3:25-27; 3:28-55; 3:65-67; 4:17-22; 21:18-32; 22:1-6; Claims: 1, 36, 70, 71, 72, 85, 87, 112, 317</p>

Term/Phrase in Issue and Asserted Claim(s) in Which Term/Phrase Appears	Plaintiff's Proposed Construction	Plaintiff's Citations to Intrinsic Evidence
<p>20 wireless communication[s] means</p> <p>'353 Patent: 33 by reference to 1</p> <p>'926 Patent: 30, 31 by reference to 30, 40 by reference to 30, 41 by reference to 30, 43 by reference to 30, 55 by reference to 52, 59 by reference to 52, 72 by reference to 52, 75 by reference to 52</p>	<p><u>Functions:</u></p> <p>facilitate wireless communication with a network that supports access to the Internet</p> <p>facilitate communication with a mobile service provider network via which Web content may be accessed</p> <p>facilitate wireless communication with a network via which Web content may be accessed</p> <p><u>Corresponding structure:</u> an antenna</p>	<p>'926 Patent (Ex. 2) Figures 1C, 2C, 3, 4A, 4B, 4C, 4D, 4E, 4F, 5, 6, 10; Specification: 3:5-8; 3:15-17; 3:18-46; 3:56-58; 4:9-14; 21: 20-33; 22:3-8; Claims: 1, 30, 52, 79, 80, 81</p> <p>'353 Patent (Ex. 1) Figures: 1A, 1B Specification: 3:4-10; 3:11-14; 6:53-64; 9:14-31; 9:57-58; Claims: 1, 33, 36, 70, 87, 112, 317</p> <p>'926 Patent (Ex. 2) Figure: 1A, 1B Specification: 2:61-67; 3:1-4; 9:4-21; 9:47-53; Claims: 1, 30, 52</p> <p>File History for U.S. Patent No. 7,461,353 (Ex. 3E) FH001072</p>

APPENDIX B—DEFENDANTS’ CITATIONS TO INTRINSIC EVIDENCE¹

Term/Phrase in Issue and Asserted Claim(s) in Which Term/Phrase Appears	Defendants’ Proposed Construction	Defendants’ Citations to Intrinsic Evidence
<p>1 scalable content</p> <p>‘353 Patent: 33, 43, 48 by reference to 36, 52 by reference to 36, 58 by reference to 36, 66 by reference to 36, 139 by reference to 118, 183, 283</p> <p>‘926 Patent: 55 by reference to 52, 59, 72 by reference to 52, 75 by reference to 52</p>	<p>data in a format generated after pre-rendering that provides the layout, functionality, and design of the webpage at multiple user-selectable scaled resolutions</p> <p>“pre-rendering”: the process of blocks 150-154 of Figure 5 (see ‘353 patent, col. 15, l. 43 to col. 17, l. 41)</p>	<p>“1. Field of the Invention. The invention relates generally to translation of Internet and World Wide Web content to scalable vector representation. More particularly, the invention relates to apparatus and methods for zoom enabling the display of content in an Internet information browser by retrieving and translating HyperText Markup Language (HTML), extensible Markup Language (XML), and other Internet content to vector representations of that content.” ‘353 patent, col. 1, ll. 42-50.</p> <p>“In accordance with aspects of the invention, mobile devices enabled to support resolution-independent scalable display of Internet (Web) content to allow Web pages to be scaled (zoomed) and panned for better viewing on smaller screen sizes are disclosed. The mobile devices employ novel processing of original Web content, including HTML-based content, XML, cascade style sheets, etc. to generate scalable content. The scalable content and/or data derived therefrom are then employed to enable the Web content to be rapidly rendered, zoomed, and panned.” ‘353 patent, col. 2, ll. 23-31.</p> <p>“According to additional aspects of the invention, methods and software for enabling support for resolution-independent scalable display of Web content are provided. The methods and software enable users of various devices, from handheld devices with small screens, to desktop PC’s and laptops, to very large screen devices, to view and interact with Web pages in a manner independent of the screen resolution of such device’s built-in or associated display, while maintaining the look and feel of browsing such pages with a conventional desktop browser. Thus, users of various devices having different screen resolutions are enabled to browse Web pages from among literally billions of Web pages while providing a full Web browsing experience.” ‘353 patent, col. 2, ll. 47-59.</p>

¹ Citations to the ‘353 patent specification also refer to the identical portion of the ‘926 patent specification. Any Figures referred to or discussed in the cited written portions of the specification are incorporated by reference herein. Defendants also rely on the language of the asserted and unasserted claims in which the disputed terms appear.

Term/Phrase in Issue and Asserted Claim(s) in Which Term/Phrase Appears	Defendants' Proposed Construction	Defendants' Citations to Intrinsic Evidence
		<p>“Apparatus and methods are described for creating resolution independent vector display of Internet content to allow it to be scaled (zoomed) larger and smaller for better viewing or to fit any resolution or screen size. In addition, infrastructure and methods are provided for delivering such content to clients.” ‘353 patent, col. 4, ll. 3-8.</p> <p>“According to one embodiment, an ultra-thin client-side viewer provides the graphics, linking, caching, and function handling capabilities necessary for extending the web to almost any platform. It is designed as a lightweight browser (micro-browser) running directly on device operating systems. In alternative embodiments, the client-side viewer may be deployed as a standard browser plugin, or Java applet for extending browser functionality. In one embodiment, the client-side viewer attains its small size and efficiency by taking advantage of the power of SVF (Simple Vector Format) to describe almost any current web content. SVF files can be handled with a tiny fraction of the client code required by normal web browsers because current browsers must interpret a large and growing number of file types and their idiosyncrasies. SVF was originally designed to handle a superset of the most commonly used file formats in the complex world of CAD.” ‘353 patent, col. 4, ll. 43-59.</p> <p>“By working tightly with a server-side content translator, web content and functionality can be passed seamlessly to the end user platform without any degradation in the look or feel of the output. In addition, because the resulting file graphics are handled as vectors, the end user can control real time changes in the size of text and graphics as well as what portion of the file is viewable in the display. This ‘zoom and pan’ capability, familiar to CAD and other vector content software users, adds dramatically to the usability of non-standard display sizes. For very small displays, real time zooming and panning allows the user to see graphics and text at sizes that make them easily readable, and then ‘back up’ to view an entire page for context or pan in any direction for navigation. Because the client-side viewer manipulates vectors, there is no loss in quality as the display is zoomed. The graphics rendering engine within the client is so efficient that file manipulation</p>

Term/Phrase in Issue and Asserted Claim(s) in Which Term/Phrase Appears	Defendants' Proposed Construction	Defendants' Citations to Intrinsic Evidence
		<p>happens in a fraction of a second. There is no perceptible wait for the user as the file is resized, or the window is repositioned. Content created for one display resolution now can be sized, real time, for any other display without degradation. Besides making small displays eminently usable, this technology extends web content into some surprising new arenas. For example, it enables normal desktop displays to be effective for individuals with visual impairment, or for content designed for 640.times.480 standard PC monitors to be shown without degradation on web billboards now appearing in cities like Seattle and San Francisco.</p> <p>With a client of such extraordinary power packed in a tiny footprint, end user device manufacturers can free up valuable memory space for pre-fetching, caching and pre-loading content, dramatically improving performance for use in low bandwidth and portable applications. In the example of a wireless handheld device where expensive flash memory must be used instead of more cost effective bulk storage technology, the difference between consuming 10's of megabytes of flash memory with a standard browser versus running the client-side viewer described herein is dramatic.” ‘353 patent, col. 4, l. 64 to col. 5, l. 34; see also <i>id.</i> Fig. 5, Fig. 6; <i>id.</i> col. 5, ll. 36-59.</p> <p>“When the HTML documents and graphic content are received by proxy server 32, a scalable vector representation of the web page is generated in a block 114 by an HTML translator 58. In brief, HTML translator 58 translates HTML, XML, and cascaded style sheet (CSS) layout content into a scalable vector representation, such as SVF. Details of the HTML translation process are contained below. In addition, the graphic images are converted into a compressed bitmap format in a block 116 by a graphics translator 60. The vectorized content 62 and compressed bitmaps 64 are then streamed back to the client (i.e., computer 18) in a block 118, as depicted by a transfer path 66. In one embodiment, the content portions are sent in separate streams using multiplexed connections. In another embodiment, the content portions are sent via a multiplexed stream using a single connection. As the vectorized content and compressed bitmap data are received by the client device, they are processed by a thin client 68 running on the client device, whereby a representation of the original web page content may be rendered on the client device's display screen at various user-selectable scaled resolutions and pan offsets in a block 120,</p>

Term/Phrase in Issue and Asserted Claim(s) in Which Term/Phrase Appears	Defendants' Proposed Construction	Defendants' Citations to Intrinsic Evidence
		<p>thereby enabling a user to more clearly see an overview or details in the web page.” ‘353 patent, col. 8, l. 57 to col. 9, l. 12.</p> <p>“As before, the retrieved HTML documents are translated into scalable vector representations by HTML translator 58 in a block 114, while the graphic images are translated into a compressed bitmap format by image translator 60 in a block 116, as depicted by vectorized content 62 and bitmap content 64. The vectorized content and bitmap content are then streamed from the web server to the client in a block 119, as depicted by a transfer path 67. Upon arriving at the client, the vectorized content and bitmap content are processed, scaled, and rendered on the client in a block 120.” ‘353 patent, col. 10, ll. 22-31.</p> <p>“At this point, the client performs translations on the HTML content and the graphic image content that are substantially similar to that performed by the proxy server in FIG. 1A or at the web site in FIG. 1B, as provided by blocks 114 and 116. The vectorized and image content is then processed and scaled by thin client 68 in a block 120, as depicted by device output 71.” ‘353 patent, col. 10, ll. 50-56.</p> <p>“As discussed above, HTML translator 58 creates a scalable vector representation of the original HTML content of a requested web page.” ‘353 patent, col. 12, ll. 12-14.</p> <p>“Next, in a block 154, the page layout is defined based on the bounding boxes. In actuality, generation of the page layout information is performed in conjunction with defining the boundary boxes for the objects, wherein the location of a given object is based on the location of other related (e.g., if within a table) or non-related objects corresponding to HTML content that have been previously parsed. For example, the location of a given paragraph will depend on the other content for the page that are listed prior to the definition for the paragraph in the primary HTML document or referenced document, if applicable. As the HTML content of the primary and any referenced HTML documents are parsed, the page layout is generated based on the various HTML tags and the content embedded between tag pairs and/or referenced by a tag pair statement (e.g., graphic images).</p>

Term/Phrase in Issue and Asserted Claim(s) in Which Term/Phrase Appears	Defendants' Proposed Construction	Defendants' Citations to Intrinsic Evidence
		<p>As will be recognized by those skilled in the art, the functions performed in blocks 150, 152, and 154 are commonly performed by conventional browsers during a pre-rendering process</p> <p>At this point, the present invention deviates substantially from the prior art by using the various object layout data generated during the pre-rendering process to generate a scalable vector representation of the original page content.” ‘353 patent, col. 17, ll. 16-45; <i>see also id.</i> col. 15, l. 43 to col. 20, l. 55; <i>id.</i> Fig. 5, Fig. 6.</p> <p>“[T]he claimed invention takes <u>HTML</u>-based Web content <u>in its original format</u> (which defines the page layout, <u>functionality</u> and design of the web page) and <u>translates</u> the HTML-based Web content into “<u>scalable content</u>” that supports a scalable, <u>resolution-independent</u> representation of the HTML-based Web content. In other words, the claimed invention <u>converts</u> an <u>HTML</u> web page into a “<u>vector graphics</u>” web page and displays the web page on a PDA. When viewing the “vector graphics” web page on the PDA, the user may <u>zoom in and out</u> of the displayed web page, in order to increase/decrease the size of the web page components that are displayed on the PDA. Additionally, the claimed invention preserves the <u>functionality</u> of the <u>original HTML web page</u> <u>after</u> it has been <u>translated</u> into a “<u>vector graphics</u>” web page and displayed on the PDA.” ‘353 patent Notice of Allowability dated Aug. 8, 2008, at 3-4 (all emphasis in original).</p> <p>“During the translation process, and in the process of serving cached, pre-translated, or native SVF content, output files are ‘streamed’ to the client-side viewer. . . . Content can be selectively streamed, with text and links coming through first, followed by graphic images and other content, for example. Should the user be accessing a link, rather than having interest in the entire file served, links can be selected early in the transfer and the next file download started immediately. In addition to streaming, the server-side content converter may also layer the content by type. This means that text can be put in one layer, links in another, GIF images in another, Javascript in another and so on. Layers can be turned on or off depending upon client capabilities, making files for less capable clients, or for users interested in a reduced functionality, higher transfer performance mode to be handled automatically.” ‘353 patent, col. 6, ll. 17-34.</p>

Term/Phrase in Issue and Asserted Claim(s) in Which Term/Phrase Appears	Defendants' Proposed Construction	Defendants' Citations to Intrinsic Evidence
		<p>“HTML is a standardized language that describes the layout of content on a web page, and attributes of that content. This layout and attribute information is defined by sets of tags contained in HTML code corresponding to the page. The tags define various HTML layout and display information, including tables, paragraph boundaries, graphic image positions and bounding box sizes, typeface styles, sizes, and colors, borders, and other presentation attributes. A portion or all of a web page's text content may be contained in the parent HTML document corresponding to the URL. In addition to basic HTML, web page documents may contain XML (eXtensible markup language) code, as well as scripting language code, such as javascript. However, for simplicity, any documents containing web page content other than only graphic content that are discussed herein will be referred to as HTML documents.” ‘353 patent, col. 7, l. 56 to col. 8, l. 4.</p> <p>‘353 patent Supplemental Amendment dated May 20, 2008, deleting “attributes” and inserting “functionality, and design” in multiple claims, e.g., prosecution claim 71:</p> <p>. . . HTML-based Web content having an original format defining an original width and height of the Web page and an original page layout and attributes, functionality, and design of content on the Web page; <u>in response to a user request to access the Web page,</u></p> <p>. . . scalable content that supports a scalable resolution-independent display <u>representation</u> of the Web page that substantially retains <u>preserves</u> the original page layout, <u>functionality</u> and attributes <u>design</u> of the content</p> <p><i>See also id.</i>, Remarks at 77-95.</p> <p>“The scalable vector representation is completed in a block 160, wherein a reference is created for each object that includes or links an object's content and attributes, such as object type (e.g., text, image), object typeface, and boundary box parameters, to the object's vector.” ‘353 patent, col. 18, ll. 17-19.</p>

Term/Phrase in Issue and Asserted Claim(s) in Which Term/Phrase Appears	Defendants' Proposed Construction	Defendants' Citations to Intrinsic Evidence
		<p>“[C]ontent is then scaled in a block 170. For image content, this comprises decompressing and scaling the compressed bitmaps corresponding to those images. For text content, this comprises scaling the font (i.e., typeface) that the text content portions of the web page are written in the parent HTML document and any referenced documents. There are various techniques for typeface scaling that may be implemented here, depending on the available resources provided by the operating system of the client device. For example, for WINDOWS.TM. operating systems, many TRUETYPE.TM. fonts are available, which use a common scalable definition for each font, enabling those fonts to be scaled to just about any size. In other cases, such as current PDA (e.g., Palm Pilots) operating systems, there is no existing feature that supports scaling fonts. As a result, bitmapped fonts of different font sizes and styles may be used. In addition to scaling image and text content, other types of content, such as separator lines and borders may also be scaled by block 170.” ‘353 patent, col. 20, ll. 25-43.</p> <p>“However, for simplicity, any documents containing web page content other than only graphic content that are discussed herein will be referred to as HTML documents.” ‘353 patent, col. 8, ll. 1-4.</p> <p>“Furthermore, graphic content oftentimes comprises significantly larger file sizes than HTML content, leading to significant transfer times in some instances. For simplicity, the transfer of the various HTML documents and graphic files for the content request are depicted by HTML documents 52 and graphic documents 54, which are transferred over a transfer path 56.” ‘353 patent, col. 8, ll. 50-56.</p> <p>“When the HTML documents and graphic content are received by proxy server 32, a scalable vector representation of the web page is generated in a block 114 by an HTML 60 translator 58. In brief, HTML translator 58 translates HTML, XML, and cascaded style sheet (CSS) layout content into a scalable vector representation, such as SVF.” ‘353 patent, col. 8, ll. 57-62.</p> <p>“Mobile devices enabled to support resolution-independent scalable display of Internet (Web) content to allow Web pages to be scaled (zoomed) and panned for</p>

Term/Phrase in Issue and Asserted Claim(s) in Which Term/Phrase Appears	Defendants' Proposed Construction	Defendants' Citations to Intrinsic Evidence
		<p>better viewing on smaller screen sizes. The mobile devices employ software-based processing of original Web content, including HTML-based content, XML, cascade style sheets, etc. to generate scalable content. The scalable content and/or data derived therefrom are then employed to enable the Web content to be rapidly rendered, zoomed, and panned. Moreover, the rendered displays provide substantially the same or identical layout as the original Web page, enabling users to easily navigate to selected content and features on familiar Web pages.” ‘353 patent, Abstract.</p> <p>‘353 patent, Fig. 1A, Fig. 1B, Fig. 1C, Fig. 2A, Fig. 2B, Fig. 2C, Fig. 3, Fig. 4A, Fig. 4B, Fig. 4C, Fig. 4D, Fig. 4E, Fig. 4F, Fig. 4G, Fig. 7A, Fig. 7B, Fig. 8A, Fig. 8B, Fig. 9A, Fig. 9B.</p> <p>App. No. 11/868,124, Remarks dated Nov. 24, 2010, at 18-139.</p> <p>App. No. 11/868,124, Reply to Action of January 24, 2011, dated July 22, 2011, at 14-18.</p> <p>App. No. 11/868,124, Response to Office Action of October 13, 2011, dated Apr. 3, 2012, at 14-19.</p> <p>App. No. 11/868,124, Office Action mailed Apr. 18, 2012, at 2-12.</p> <p>“In accordance with teachings disclosed in the present application (and its related applications incorporated herein), users of various devices, from handheld devices with small screens, to desktop PC's and laptops, to very large screen devices, are enabled to view and interact with Web pages in a manner independent of the screen resolution of such devices' built-in or associated display, while preserving the look and feel (i.e., functionality) of browsing such pages with a conventional desktop browser. As a result, users are enabled to access millions of Web pages on various devices having different screen resolutions while providing a full Web browsing experience similar to that experienced when browsing the same Web pages using a desktop browser.</p>

Term/Phrase in Issue and Asserted Claim(s) in Which Term/Phrase Appears	Defendants' Proposed Construction	Defendants' Citations to Intrinsic Evidence
		<p>In order to clarify this result, Applicants have amended many of the claims to recite, in part, "preserves the original page layout, functionality, and design of the [HTML-based Web page] content." '353 patent Supplemental Amendment dated May 20, 2008, at 77.</p> <p>"As discussed above and in further detail in the present specification, Web pages comprise HTML-based content which may be stored in one or more documents commonly referred to as HTML documents. In addition, Web pages may include dynamically-generated content. Each Web page has a corresponding main or 'parent' HTML document that includes HTML code defining the Web page content layout, at least at some level. The parent HTML document may reference other HTML documents, as well as other content (such as image content) that further define the layout of content contained in the referenced documents. This may proceed in a hierarchical or nested fashion.</p> <p>To access the Web page, the browser initiates an HTTP connection with the Web server hosting the Web page, and begins downloading the parent HTML document. Depending on how the Web server and/or Web page is configured, additional content (i.e., beyond that included in the parent HTML document) referenced by the parent HTML document, may be retrieved by the Web page host server and then downloaded to the requesting client device, or a portion of this content may be downloaded by the client device via a separate connection. Generally, content that is hosted by a Web server or Web site is assembled by the Web server and downloaded to the client device. On the other hand, externally-referenced content (that is, content that is not stored on the Web server or Web site), is often left to the client device (i.e., the browser) to retrieve." '353 patent Supplemental Amendment dated May 20, 2008, at 82.</p> <p>"A point for discussing the foregoing is to make it clear that,</p> <ol style="list-style-type: none"> 1. Even when rendering the same Web page source content (i.e., the HTML code definition of the Web page), conventional Web browsers may not render the (non-scaled) Web page identically. Scaling Web pages may also result in alternation of the page layout. However, under

Term/Phrase in Issue and Asserted Claim(s) in Which Term/Phrase Appears	Defendants' Proposed Construction	Defendants' Citations to Intrinsic Evidence
		<p>aspects of embodiments of the invention (such as claimed in claim 71) the overall layout, functionality and appearance (design) of the scaled Web pages defined by the HTMLcode for the Web page are preserved.</p> <p>2. Plug-ins may be required to render non-HTML content that is embedded within some web pages or used in a separate window launched from a web page. Notably, the plug-in content is not a Web page, but rather a specific type of content requiring a corresponding plug-in application to render the content.</p> <p>The new claim language introduces the term 'functionality.' Preserving functionality generally pertains to preserving the interoperability of various HTML-based Web page content, such as hyperlinks and UI controls such as input forms defined via corresponding HTML-based code. It is noted that the HTML code defining a Web page's overall layout, functionality and design does not define how a user interaction with the Web content is to be supported, but rather defines the existence of a corresponding function within the Web content to support the interaction. For example, a hyperlink definition within a Web page merely defines a link (hyperlink reference of <i>href</i>) to corresponding content, it does not define how the hyperlink associated control is to appear on the screen nor how the hyperlink is to be activated. That is up to the browser's implementation, which varies by browser. For example, some browsers underline text content associated with a hyperlink, while others change the appearance of a pointer when over a control (e.g., text content) associated with a hyperlink (or otherwise change the appearance of such content). Moreover, how the hyperlink is activated is not defined by the corresponding HTML-based definition, but again is left to the browser implementation. Accordingly, preserving content functionality means that functionality defined by corresponding HTML code (e.g., activation of a hyperlink in the present example) is supported, without limiting the particular user interface for how that activation is facilitated." '353 patent Supplemental Amendment dated May 20, 2008, at 90-91.</p> <p>See "scalable content" above and "scalable vector-based content" below.</p>
2 scalable / scaling / scaled '353 Patent: 33, 33 by reference	"scaling" / "scaled": These terms have a plain and ordinary meaning and do not need to be	

Term/Phrase in Issue and Asserted Claim(s) in Which Term/Phrase Appears	Defendants' Proposed Construction	Defendants' Citations to Intrinsic Evidence
<p>to 1, 43, 48 by reference to 36, 52 by reference to 36, 58 by reference to 36, 59 by reference to 36, 66 by reference to 36, 139 by reference to 118, 183, 283</p> <p>'926 Patent: 55 by reference to 52, 59, 72 by reference to 52, 75 by reference to 52</p>	<p>construed. However, if the Court chooses to construe these terms, the following constructions should be used:</p> <p>“scaling”: setting to a user-selected resolution</p> <p>“scaled”: set to a user-selected resolution</p> <p>“scalable”: defined in the context of “scalable content” and “scalable vector-based content”</p> <p>converting the format of</p>	
<p>3</p> <p>translating</p> <p>'353 Patent: 33 by reference to 1, 139 by reference to 118</p> <p>'926 Patent: 30, 31 by reference to 30, 40 by reference to 30, 41 by reference to 30, 43 by reference to 30</p>		<p>“The invention relates generally to translation of Internet and World Wide Web content to scalable vector representation. More particularly, the invention relates to apparatus and methods for zoom enabling the display of content in an Internet information browser by retrieving and translating HyperText Markup Language (HTML), extensible Markup Language (XML), and other Internet content to vector representations of that content. '353 patent, col. 1, ll. 43-50.</p> <p>“The mobile devices employ novel processing of original Web content, including HTML-based content, XML, cascade style sheets, etc. to generate scalable content.” '353 patent, col. 2, ll. 26-29.</p> <p>“When the HTML documents and graphic content are received by proxy server 32, a scalable vector representation of the web page is generated in a block 114 by an HTML translator 58. In brief, HTML translator 58 translates HTML, XML, and cascaded style sheet (CSS) layout content into a scalable vector representation, such as SVF.” '353 patent, col. 8, ll. 57-62.</p> <p>“In a block 103, HTTP negotiations are performed to determine the format the content is to be delivered in. For example, the request may contain indicia</p>

Term/Phrase in Issue and Asserted Claim(s) in Which Term/Phrase Appears	Defendants' Proposed Construction	Defendants' Citations to Intrinsic Evidence
		<p>identifying the type of content requested, such as an SVF MIME type (e.g., image/vnd.svf)." '353 patent, col. 10, ll. 2-6.</p> <p>"The proxy server responds to client content requests by delivering content in one of the requested formats, by retrieving the content in an appropriate format from its cache, or from an upstream content source (again using standard HTTP content negotiation features), or by translating upstream content from a supported original format to SVF or the client bitmap format.</p> <p>Requests from the server installation to its cache and from the cache to upstream content sources are made in HTTP carried over TCP using simple straightforward Web content requests. For example, requests from clients to the proxy server comprise HTTP proxy requests (e.g., "GET http://www/xyz.com/some_page.html HTTP/1.0 . . . ") carried over TCP or over a lightweight multiplexing protocol over TCP. The multiplexing protocol allows the server to push image thumbnails to the client before the SVF stream is available, as well as offering a channel for control and status information, more simultaneous channels than the client operating system may support, and a mechanism for prioritizing information flow from server to client under loose client control. In addition to HTTP requests, the proxy server architecture supports other user-level protocols, such as FTP and Gopher." '353 patent, col. 11, ll. 3-24.</p> <p>"The various content translators used by the proxy server accept (via HTTP PUT) or request (driven by HTTP proxy GET/POST) content in supported, but client-unsupported, formats; and return (via HTTP PUT or GET/POST response) one or more representations of that content in a client-supported format. In the embodiments illustrated in FIG. 1A-C, two translators are used: HTML translator 58 and image translator 60. Future content types may be accommodated by new translators, by extending existing translators to cover the new content types, or by extending the client's capabilities. Standard HTTP content negotiation mechanisms are used to inform the proxy server of the client's capabilities and expectations on each request." '353 patent, col. 11, ll. 41-53.</p> <p>"With reference to the flowchart of FIG. 5, the process for translating the HTML</p>

Term/Phrase in Issue and Asserted Claim(s) in Which Term/Phrase Appears	Defendants' Proposed Construction	Defendants' Citations to Intrinsic Evidence
		<p>content into a scalable vector representation proceeds as follows. The process is initiated when the proxy server receives the HTML corresponding to the parent document (and frame documents, if appropriate), whereupon a pre-rendering parsing of the HTML is performed to determine where to place the various objects on the display page in a block 150</p> <p>As will be recognized by those skilled in the art, the functions performed in blocks 150, 152, and 154 are commonly performed by conventional browsers during a pre-rendering process</p> <p>At this point, the present invention deviates substantially from the prior art by using the various object layout data generated during the pre-rendering process to generate a scalable vector representation of the original page content.” ‘353 patent, col. 15, l. 43 to col. 17, l. 45.</p> <p>‘353 patent, Fig. 1A, Fig. 1B, Fig. 1C, Fig. 2A, Fig. 2B, Fig. 2C.</p> <p>“[T]he claimed invention takes <u>HTML</u>-based Web content <u>in its original format</u> (which defines the page layout, <u>functionality</u> and design of the web page) and <u>translates</u> the HTML-based Web content into “<u>scalable content</u>” that supports a scalable, <u>resolution-independent</u> representation of the HTML-based Web content. In other words, the claimed invention <u>converts</u> an <u>HTML</u> web page into a “<u>vector graphics</u>” web page and displays the web page on a PDA. When viewing the “vector graphics” web page on the PDA, the user may <u>zoom in and out</u> of the displayed web page, in order to increase/decrease the size of the web page components that are displayed on the PDA. Additionally, the claimed invention preserves the <u>functionality</u> of the <u>original HTML web page after</u> it has been <u>translated</u> into a “<u>vector graphics</u>” <u>web page</u> and displayed on the PDA.” ‘353 patent Notice of Allowability dated August 8, 2008, at 3-4 (all emphasis in original).</p> <p>“When the HTML documents and graphic content are received by proxy server 32, a scalable vector representation of the web page is generated in a block 114 by an HTML translator 58. In brief, HTML translator 58 translates HTML, XML, and</p>

Term/Phrase in Issue and Asserted Claim(s) in Which Term/Phrase Appears	Defendants' Proposed Construction	Defendants' Citations to Intrinsic Evidence
		<p>cascaded style sheet (CSS) layout content into a scalable vector representation, such as SVF.” ‘353 patent, col. 8, ll. 57-62.</p> <p>“As before, the retrieved HTML documents are translated into scalable vector representations by HTMLtranslator 58 in a block 114, while the graphic images are translated into a compressed bitmap format by image translator 60 in a block 116, as depicted by vectorized content 62 and bitmap content 64.” ‘353 patent, col. 10, ll. 22-27.</p> <p>“This content is depicted by HTML documents 52 and image files 54, which are sent from the network site to the client via a transfer path 69. At this point, the client performs translations on the HTMLcontent and the graphic image content that are substantially similar to that performed by the proxy server in FIG. 1Aor at the web site inFIG.1B, as provided by blocks 114 and 116. The vectorized and image content is then processed and scaled by thin client 68 in a block 120, as depicted by device output 71.” ‘353 patent, col. 10, ll. 48-56.</p> <p>App. No. 11/868,124, Remarks dated Nov. 24, 2010, at 18-139.</p> <p>App. No. 11/868,124, Reply to Action of January 24, 2011, dated July 22, 2011, at 14-18.</p> <p>App. No. 11/868,124, Response to Office Action of October 13, 2011, dated Apr. 3, 2012, at 14-19.</p> <p>App. No. 11/868,124, Office Action mailed Apr. 18, 2012, at 2-12.</p> <p>See “translating” above.</p>
4 processing [the] HTML-based Web content to produce scalable content ‘353 Patent: 43 by reference to 36, 48 by reference to 36, 52 by reference to 36, 58 by reference	converting [the] Web content from its HTML format to a scalable content format	

Term/Phrase in Issue and Asserted Claim(s) in Which Term/Phrase Appears	Defendants' Proposed Construction	Defendants' Citations to Intrinsic Evidence
<p>to 36, 59 by reference to 36, 66 by reference to 36, 183 by reference to 149, 283 by reference to 252</p> <p>'926 Patent: 55 by reference to 52, 59 by reference to 52, 72 by reference to 52, 75 by reference to 52</p>		
<p>5</p> <p>format</p> <p>'353 Patent: 33 by reference to 1, 139 by reference to 118</p> <p>'926 Patent: 30, 31 by reference to 30, 40 by reference to 30, 41 by reference to 30, 43 by reference to 30, 75</p>	<p>structure of data in a file (e.g., .htm, .css, .svf, .bmp, .jpg, .gif)</p>	<p>'353 patent, Fig. 2A, Fig. 2B, Fig. 2C (at box 116: "IMAGE CONTENT TRANSLATED INTO COMPRESSED BITMAP FORMAT").</p> <p>"Internet content is stored in multiple file formats. These formats include HTML (Hyper Text Markup Language) and XML (extended Markup Language) as well as graphic file format GIF (Graphics Interchange Format) and JPEG (Joint Photographic Experts Group). These four file formats constitute the majority of Internet content" '353 patent, col. 1, ll. 61-65.</p> <p>"According to one embodiment, an ultra-thin client-side viewer provides the graphics, linking, caching, and function handling capabilities necessary for extending the web to almost any platform. It is designed as a lightweight browser (micro-browser) running directly on device operating systems. In alternative embodiments, the client-side viewer may be deployed as a standard browser plug-in, or Java applet for extending browser functionality. In one embodiment, the client-side viewer attains its small size and efficiency by taking advantage of the power of SVF (Simple Vector Format) to describe almost any current web content. SVF files can be handled with a tiny fraction of the client code required by normal web browsers because current browsers must interpret a large and growing number of file types and their idiosyncrasies. SVF was originally designed to handle a superset of the most commonly used file formats in the complex world of CAD. It can accommodate not only new graphical functions, but the storage and transfer of almost any foreseeable new functional capability. SVF has been under</p>

Term/Phrase in Issue and Asserted Claim(s) in Which Term/Phrase Appears	Defendants' Proposed Construction	Defendants' Citations to Intrinsic Evidence
		<p>consideration by the W3C (World Wide Web Consortium) for adoption as a standard for vector content on the World Wide Web.</p> <p>By working tightly with a server-side content translator, web content and functionality can be passed seamlessly to the end user platform without any degradation in the look or feel of the output. In addition, because the resulting file graphics are handled as vectors, the end user can control real time changes in the size of text and graphics as well as what portion of the file is viewable in the display. This 'zoom and pan' capability, familiar to CAD and other vector content software users, adds dramatically to the usability of non-standard display sizes. For very small displays, real time zooming and panning allows the user to see graphics and text at sizes that make them easily readable, and then 'back up' to view an entire page for context or pan in any direction for navigation. Because the client-side viewer manipulates vectors, there is no loss in quality as the display is zoomed. The graphics rendering engine within the client is so efficient that file manipulation happens in a fraction of a second. There is no perceptible wait for the user as the file is resized, or the window is repositioned. Content created for one display resolution now can be sized, real time, for any other display without degradation. Besides making small displays eminently usable, this technology extends web content into some surprising new arenas. For example, it enables normal desktop displays to be effective for individuals with visual impairment, or for content designed for 640.times.480 standard PC monitors to be shown without degradation on web billboards now appearing in cities like Seattle and San Francisco." '353 patent, col. 4, l. 42 to col. 5, l. 24.</p> <p>"The server-side content translator rapidly translates Web content to SVF, compresses and encrypts the SVF results if desired, and transfers the vector formatted results to the client-side viewer." '353 patent, col. 6, ll. 6-10.</p> <p>"These files will typically 15 comprise data stored in one of several well-known graphic formats, including bitmap files (BMP), GIF (Graphics Interchange Format) files, and JPEG (Joint Photographic Experts Group) files." '353 patent, col. 8, ll. 14-18.</p>

Term/Phrase in Issue and Asserted Claim(s) in Which Term/Phrase Appears	Defendants' Proposed Construction	Defendants' Citations to Intrinsic Evidence
6 vector-based content '353 Patent: 33, 183, 283	scalable content that includes, for each object, a directional data structure storing X & Y values from a single point for the page or frame to the object	<p>“Standard HTTP content negotiations features specify the formats in which content is to be delivered (SVF, bitmap, and possibly others, which can be handed off to cooperating client-side display software).” ‘353 patent, col. 10, ll. 62-65.</p> <p>“The proxy server responds to client content requests by delivering content in one of the requested formats, by retrieving the content in an appropriate format from its cache, or from an upstream content source (again using standard HTTP content negotiation features), or by translating upstream content from a supported original format to SVF or the client bitmap format.” ‘353 patent, col. 11, ll. 3-9.</p>
		<p>“By working tightly with a server-side content translator, web content and functionality can be passed seamlessly to the end user platform without any degradation in the look or feel of the output. In addition, because the resulting file graphics are handled as vectors, the end user can control real time changes in the size of text and graphics as well as what portion of the file is viewable in the display. This “zoom and pan” capability, familiar to CAD and other vector content software users, adds dramatically to the usability of non-standard display sizes. For very small displays, real time zooming and panning allows the user to see graphics and text at sizes that make them easily readable, and then “back up” to view an entire page for context or pan in any direction for navigation.” ‘353 patent, col. 4, l. 64 to col. 5, l. 9.</p> <p>“When the HTML documents and graphic content are received by proxy server 32, a scalable vector representation of the web page is generated in a block 114 by an HTML translator 58. In brief, HTML translator 58 translates HTML, XML, and cascaded style sheet (CSS) layout content into a scalable vector representation, such as SVF.” ‘353 patent, col. 8, ll. 57-62.</p> <p>‘353 patent, Fig. 5.</p> <p>“At this point, the present invention deviates substantially from the prior art by using the various object layout data generated during the pre-rendering process to generate a scalable vector representation of the original page content.” ‘353 patent, col. 17, ll. 42-45.</p>

Term/Phrase in Issue and Asserted Claim(s) in Which Term/Phrase Appears	Defendants' Proposed Construction	Defendants' Citations to Intrinsic Evidence
		<p>“First, in a block 156, a datum point is defined for the page and the bounding box for each object. For example, as shown in FIG. 4C, a rendered page datum 262 is defined to be coincident with the upper left hand corner of the display frame of the rendered page for the web page. Generally, any point on the page may be used as the page datum—the only requirement is that the page datum that is selected is used consistently throughout the process. The use of the upper left hand corner of the display frame is advantageous since the location of the first object encountered in the HTML code for a page is located relative to this corner.” ‘353 patent, col. 17, ll. 45-56.</p> <p>“Once the page’s datum point and an object’s datum point are known, a vector between these points is generated for each object in a block 158.” ‘353 patent, col. 17, ll. 65-67.</p> <p>“The scalable vector representation is completed in a block 160, wherein a reference is created for each object that includes or links an object’s content and attributes, such as object type (e.g., text, image), object typeface, and boundary box parameters, to the object’s vector. For example, object 250B is a graphic image having a vector 250D and a bounding box that is 180 pixels high and 350 pixels wide, while object 252B is a graphic image having a vector 252D and a bounding box that includes a height of 200 pixels and a width of 350 pixels. This enables client-side operations to be performed that only initially consider the vectors, wherein if it is determined that a vector’s endpoint (and/or the bounding box corresponding to the object the vector points to) would appear off of a display, there is no need to retrieve the content and attribute data linked to the vector. This concept is explained in further detail in the following section.” ‘353 patent, col. 18, ll. 17-32.</p> <p>‘926 patent Claim 21: “The mobile device of claim 1, wherein translating the HTML-based Web content to produce scalable vector-based page layout information comprises: processing the HTML-based Web content with a rendering engine to generate page layout information corresponding to the original page</p>

Term/Phrase in Issue and Asserted Claim(s) in Which Term/Phrase Appears	Defendants' Proposed Construction	Defendants' Citations to Intrinsic Evidence
		<p>layout as interpreted by the rendering engine; and employing the page layout information to produce scalable vector-based page layout information.”</p> <p>‘926 patent Claim 22: “The mobile device of claim 21, wherein the page layout information defines a layout location for a plurality of objects, including text objects, graphic layout objects, and/or image objects included on the Web page, and wherein producing vector-based page layout information comprises: defining a primary datum corresponding to a page layout; and, for each object, defining an object datum corresponding to the layout location for the object on the page layout; generating a vector from the primary datum to the object datum for the object; and creating a reference that links the object to its corresponding vector.”</p> <p>“vector (FIG. 4C) directional data structure stores X & Y values from known datum (primary datum) to object bounding box (object datum) vector between these points is generated for each object (17:65-67) Evidence supporting the definition of the term ‘vector’ is provided by Examiner from the Microsoft Computer Dictionary, Fifth Edition.” ‘353 patent Order Granting <i>Inter Partes</i> Reexamination dated Aug. 8, 2011, at 15.</p> <p>“[C]ontent is then scaled in a block 170. For image content, this comprises decompressing and scaling the compressed bitmaps corresponding to those images. For text content, this comprises scaling the font (i.e., typeface) that the text content portions of the web page are written in the parent HTML document and any referenced documents. There are various techniques for typeface scaling that may be implemented here, depending on the available resources provided by the operating system of the client device. For example, for WINDOWS.TM. operating systems, many TRUETYPE.TM. fonts are available, which use a common scalable definition for each font, enabling those fonts to be scaled to just about any size. In other cases, such as current PDA (e.g., Palm Pilots) operating systems, there is no existing feature that supports scaling fonts. As a result, bitmapped fonts of different font sizes and styles may be used. In addition to scaling image and text content, other types of content, such as separator lines and borders may also be scaled by block 170.” ‘353 patent, col. 20, ll. 25-43.</p>

Term/Phrase in Issue and Asserted Claim(s) in Which Term/Phrase Appears	Defendants' Proposed Construction	Defendants' Citations to Intrinsic Evidence
7 scalable vector-based content '353 Patent: 43 '926 Patent: 30, 31 by reference to 30, 40 by reference to 30, 41 by reference to 30, 43, 59	See Defendants' proposed construction for "vector-based content."	See <i>also</i> "scalable content" above. See "vector-based content" above.
8 vector '353 Patent: 33, 43, 58, 139 by reference to 138, 183, 283 '926 Patent: 30, 31 by reference to 30, 40 by reference to 30, 41 by reference to 30, 43, 59, 72	for each object, a directional data structure that stores X & Y values from a single point for the page or frame to the object	"vector (FIG. 4C) directional data structure stores X & Y values from known datum (primary datum) to object bounding box (object datum) vector between these points is generated for each object (17:65-67) Evidence supporting the definition of the term 'vector' is provided by Examiner from the Microsoft Computer Dictionary, Fifth Edition." '353 patent Order Granting <i>Inter Partes</i> Reexamination dated August 8, 2011, at 15. See <i>also</i> "scalable content," "scalable vector-based content," and "vector-based content" above.
9 primary datum '353 Patent: 58, 59 by reference to 58, 139 by reference to 138 '926 Patent: 72	single point defined at a fixed X,Y coordinate on the full-size web page	"FIG. 4A is a representation of an exemplary web page has displayed on a conventional browser; FIG. 4B is a schematic diagram illustrates various objects that are generated based on the HTML code of the web page of FIG. 4A; FIG. 4C is a schematic diagram illustrating a set of vectors and bounding boxes corresponding to the objects generated in FIG. 4B; FIG. 4D is a schematic diagram illustrating how various vectors and bounding boxes may be defined in accordance with the invention; FIG. 4E is a representation of the web page of FIG. 4A after it has been offset and scaled in accordance with the invention; FIG. 4F is a schematic diagram illustrating new datum points and bounding

Term/Phrase in Issue and Asserted Claim(s) in Which Term/Phrase Appears	Defendants' Proposed Construction	Defendants' Citations to Intrinsic Evidence
		<p>boxes corresponding to the scaled and offset web page;</p> <p>FIG. 4G is a schematic diagram illustrating new vectors and bounding box parameters for a pair of objects in the scaled and offset web page;</p> <p>FIG. 5 is a flowchart illustrating the logic used by the invention when translating content into a scalable vector representation of that content;</p> <p>FIG. 6 is a flowchart illustrating client-side operations that are performed to create a rendered display page based on the translated content the client receives and user-input;" '353 patent, col. 3, ll. 31-56; see also <i>id.</i> Fig. 4A, Fig. 4B, Fig. 4C, Fig. 4D, Fig. 4E, Fig. 4F, Fig. 4G, Fig. 5, Fig. 6.</p> <p>"At this point, the present invention deviates substantially from the prior art by using the various object layout data generated during the pre-rendering process to generate a scalable vector representation of the original page content." '353 patent, col. 17, ll. 42-45.</p> <p>"First, in a block 156, a datum point is defined for the page and the bounding box for each object. For example, as shown in FIG. 4C, a rendered page datum 262 is defined to be coincident with the upper left hand corner of the display frame of the rendered page for the web page. Generally, any point on the page may be used as the page datum-the only requirement is that the page datum that is selected is used consistently throughout the process. The use of the upper left hand corner of the display frame is advantageous since the location of the first object encountered in the HTML code for a page is located relative to this corner." '353 patent, col. 17, ll. 45-56.</p> <p>"Once the page's datum point and an object's datum point are known, a vector between these points is generated for each object in a block 158. With reference to FIG. 4D, in one embodiment, wherein the page datum point corresponds to the upper left and corner of the display frame and is assigned an XY value 266 of 0,0 the vector for a given object may be stored as the XY value of the datum point of that object relative to 0,0, such as a value of 150, 225 (ref. num. 268) for a vector 250D pointing to an object datum 250C, and a value of 150, 425 (ref. num. 270) for</p>

Term/Phrase in Issue and Asserted Claim(s) in Which Term/Phrase Appears	Defendants' Proposed Construction	Defendants' Citations to Intrinsic Evidence
		<p>a vector 252D pointing to an object datum 252C. In another embodiment, each vector may be stored as XY data relative to a 0, 0 datum point corresponding to the upper left hand corner of the frame the object belongs to. For example, a vector 250D' from a frame datum 214D to object datum 250C is stored as 20, 200 (ref. num. 268'), while a vector 252D from frame datum 214D to object datum 252C is stored as 20, 425. In this embodiment, offset information for each frame relative to a known datum will also be stored, as depicted by a vector 214D." '353 patent, col. 17, l. 65 to col. 18, l. 16.</p> <p>"As shown in FIG. 4G, respective offsets in X and Y, ($-\Delta X$ and $-\Delta Y$ in the Figure) are applied to the starting point of each of the vectors. The vectors are then scaled by a scale factor 'SF.' The results of the new vectors are depicted by vectors 250D" and 252D". This produces a new datum for each object's bounding box that is relative to rendered page datum 262, which remains fixed." '353 patent, col. 19, ll. 57-63.</p> <p>"Returning to the flowchart of FIG. 6, once the vectors and bounding boxes are offset and scaled, content corresponding to objects having at least a portion of their bounding boxes falling within the display limit bounding box is retrieved from the client device's display list in a block 168." '353 patent, col. 20, ll. 18-22.</p> <p>"Next, in a block 154, the page layout is defined based on the bounding boxes. In actuality, generation of the page layout information is performed in conjunction with defining the boundary boxes for the objects, wherein the location of a given object is based on the location of other related (e.g., if within a table) or non-related objects corresponding to HTML content that have been previously parsed . . . As the HTML content of the primary and any referenced HTML documents are parsed, the page layout is generated based on the various HTML tags and the content embedded between tag pairs and/or referenced by a tag pair statement (e.g., graphic images)." '353 patent, col. 17, ll. 16-30.</p> <p>"As will be recognized by those skilled in the art, the functions performed in blocks 150, 152, and 154 are commonly performed by conventional browsers during a pre-</p>

Term/Phrase in Issue and Asserted Claim(s) in Which Term/Phrase Appears	Defendants' Proposed Construction	Defendants' Citations to Intrinsic Evidence
		<p>rendering process.” ‘353 patent, col. 17, ll. 31-34.</p> <p>‘353 patent Claim 5: “The wireless device of claim 1, wherein execution of the instructions performs further operations comprising: parsing markup language code to determine the original page layout of display content within the Web page, wherein the original page layout defines a layout location for a plurality of objects, including at least one of text objects, graphic layout objects, or graphic image objects included in the Web page; defining a primary datum corresponding to the original page layout; and, for each object, defining an object datum corresponding to the layout location for the object; generating a vector from the primary datum to the object datum for the object” See <i>also</i> ‘353 patent Claims 24, 174, 220, 239 and 274.</p> <p>‘353 patent Claim 58: “The mobile hand-held device of claim 36, wherein execution of the instructions performs further operations comprising: parsing HTML-based code corresponding to the retrieved Web page to determine the original page layout of the content on the Web page; logically grouping selected content into objects; defining a primary datum corresponding to the original page layout; and, for each object, defining an object datum corresponding to a layout location datum for the object's associated display content; generating a vector from the primary datum to the object datum for the object”</p> <p>‘353 patent Claim 118: “A method, comprising: rendering a browser interface on a display of a device via which a user is enabled to request access to a Web page, the Web page comprising HTML-based Web content having an original format defining an original width and height of the Web page and an original page layout, functionality, and design of content on the Web page; in response to a user request to access the Web page, retrieving the Web page via the device, and translating at least a portion of the HTML-based Web content from its original format into scalable content that supports a scalable resolution-independent representation of the Web page”</p> <p>‘353 patent Claim 138: “The method of claim 118, further comprising: parsing</p>

Term/Phrase in Issue and Asserted Claim(s) in Which Term/Phrase Appears	Defendants' Proposed Construction	Defendants' Citations to Intrinsic Evidence
		<p>markup language code corresponding to the retrieved Web page to determine the original page layout of the content on the Web page; logically grouping selected content into objects; defining a primary datum corresponding to the original page layout; and, for each object, defining an object datum corresponding to a layout location datum for the object's associated display content; generating a vector from the primary datum to the object datum for the object”</p> <p>‘926 patent Claim 21: “The mobile device of claim 1, wherein translating the HTML-based Web content to produce scalable vector-based page layout information comprises: processing the HTML-based Web content with a rendering engine to generate page layout information corresponding to the original page layout as interpreted by the rendering engine; and employing the page layout information to produce scalable vector-based page layout information.” See <i>also</i> ‘926 patent Claim 51.</p> <p>‘926 patent Claim 34: “The mobile phone of claim 30, wherein execution of the instructions performs further operations comprising: parsing and processing markup language code associated with the Web page to determine the original page layout of display content within the Web page, wherein the original page layout defines a layout location for a plurality of objects, including text objects, graphic layout objects, and/or graphic image objects included in the Web page; defining a primary datum corresponding to the original page layout; and, for each object, defining an object datum corresponding to the layout location for the object; generating a vector from the primary datum to the object datum for the object”</p> <p>‘926 patent Claim 72: “The mobile device of claim 52, wherein execution of the instructions performs further operations comprising: parsing HTML-based code corresponding to the received Web content to identify content on the Web page; logically grouping selected content into objects; defining a primary datum corresponding to the original page layout; and, for each object, defining an object datum corresponding to a layout location datum for the object's associated display content; generating a vector from the primary datum to the object datum for the object”</p>

Term/Phrase in Issue and Asserted Claim(s) in Which Term/Phrase Appears	Defendants' Proposed Construction	Defendants' Citations to Intrinsic Evidence
		<p>“It is common practice to lay out web page content using a page origin of 0,0 at the upper left hand corner of the page. Accordingly, web page rendering engines such as Gecko typically employ a pixel-based coordinate system having an origin at 0,0 with X values increasing to the right, and Y values increasing downward. The coordinates of the pixel-based coordinate system are integer values corresponding to the location of the pixels in a rendered page.” ‘926 patent Comments on Statement of Reasons for Allowance dated Sept. 9, 2010, at 47; <i>see also id.</i> at 42-60 (entire section designated as “Scalable Vector-based Page Layout Information”).</p> <p>“Optionally, the page layout datum may be considered at l, l, representing the X, Y location of the pixel at the upper left hand corner of a content drawing area used to render page content in a browser.” ‘926 patent Comments on Statement of Reasons for Allowance dated Sept. 9, 2010, at 47 n.11; <i>see also id.</i> at 42-60 (entire section designated as “Scalable Vector-based Page Layout Information”).</p> <p>“Since the content in this example is laid out relative to the body block, the upper left hand corner of the body block container box is defined as the 0, 0 datum for the examples herein rather than the upper left hand corner of the page.” ‘926 patent Comments on Statement of Reasons for Allowance dated Sept. 9, 2010, at 49.</p> <p>“For example, in one embodiment the page layout information including object datums (i.e., XY points) determined by the operations in the flowchart of FIG. 5 (and as illustrated in the XY page layout example of FIGs. 4A-D) are mapped as vectors to a virtual display area.” ‘926 patent Amendment and Response to Office Action dated Apr. 21, 2010, at 31.</p> <p>“This process begins by employing a rendering engine to determine page layout information that includes information from which the location for each display object (i.e., HTML element or block having content that is to be displayed) can be determined. In general, the page layout information includes information from which at least one of a datum or bounding box for each display object can be determined. In one embodiment, the page layout is interpreted using the rendering</p>

Term/Phrase in Issue and Asserted Claim(s) in Which Term/Phrase Appears	Defendants' Proposed Construction	Defendants' Citations to Intrinsic Evidence
		<p>engine based on a default page width in pixels, and the location of the page layout objects is interpreted by the rendering engine are defined by corresponding pixel locations on the page. Accordingly, at this point the page layout information, such as a datum, is defined by corresponding pixel locations in a two-dimensional (XY) pixel-based coordinate space used by the rendering engine.” ‘926 patent Comments on Statement of Reasons for Allowance dated Sept. 9, 2010, at 43; <i>see also id.</i> at 42-60 (entire section designated as “Scalable Vector-based Page Layout Information”).</p> <p>“Browser rendering engines typically determine page layout based on the size of the browser window content area (i.e., the portion of the browser application window in which the page content is rendered). In particular, the width of the content area is employed.” ‘926 patent Comments on Statement of Reasons for Allowance dated Sept. 9, 2010, at 47; <i>see also id.</i> at 42-60 (entire section designated as “Scalable Vector-based Page Layout Information”).</p> <p>“In summary, in one non-limiting embodiment the element of ‘translating the HTML-based Web content to produce scalable vector-based page layout information’ includes employing a rendering engine to interpret the page layout and mapping selected page layout coordinates from a pixel-based coordinate system to a vector based virtual coordinate space. Once the layout information is defined by vector-based points in the virtual coordinate space, the coordinates of the layout information can be scaled to produce scaled page layout information. Scaled views of Web page content can then be generated by rendering display objects using applicable scale factors at locations defined the scaled coordinates corresponding to each display object.” ‘926 patent Comments on Statement of Reasons for Allowance dated Sept. 9, 2010, at 59-60; <i>see also id.</i> at 42-60 (entire section designated as “Scalable Vector-based Page Layout Information”).</p> <p>“datum point defined for page / “page datum” and bounding box for each object (17:45-64), relative to a consistent point of reference” ‘353 patent Order Granting <i>Inter Partes</i> Reexamination dated Aug. 8, 2011, at 15.</p>
10	object datum X,Y coordinate for an object that	“FIG. 4A is a representation of an exemplary web page has displayed on a

Term/Phrase in Issue and Asserted Claim(s) in Which Term/Phrase Appears	Defendants' Proposed Construction	Defendants' Citations to Intrinsic Evidence
<p>'353 Patent: 58, 59 by reference to 58, 139 by reference to 138</p> <p>'926 Patent: 72</p>	<p>is defined across objects at a consistent location on the objects</p>	<p>conventional browser;</p> <p>FIG. 4B is a schematic diagram illustrating various objects that are generated based on the HTML code of the web page of FIG. 4A;</p> <p>FIG. 4C is a schematic diagram illustrating a set of vectors and bounding boxes corresponding to the objects generated in FIG. 4B;</p> <p>FIG. 4D is a schematic diagram illustrating how various vectors and bounding boxes may be defined in accordance with the invention;</p> <p>FIG. 4E is a representation of the web page of FIG. 4A after it has been offset and scaled in accordance with the invention;</p> <p>FIG. 4F is a schematic diagram illustrating new datum points and bounding boxes corresponding to the scaled and offset web page;</p> <p>FIG. 4G is a schematic diagram illustrating new vectors and bounding box parameters for a pair of objects in the scaled and offset web page;</p> <p>FIG. 5 is a flowchart illustrating the logic used by the invention when translating content into a scalable vector representation of that content;</p> <p>FIG. 6 is a flowchart illustrating client-side operations that are performed to create a rendered display page based on the translated content the client receives and user-input;" '353 patent, col. 3, ll. 31-56; <i>see also id.</i> Fig. 4A, Fig. 4B, Fig. 4C, Fig. 4D, Fig. 4E, Fig. 4F, Fig. 4G, Fig. 5, Fig. 6.</p> <p>"First, in a block 156, a datum point is defined for the page and the bounding box for each object." '353 patent, col. 17, ll. 45-47.</p> <p>"In general, the datum points for each object may also be located any place on the object, as long as the object datum points are used in a predictable manner. For example, as depicted in FIG. 4C, various datum points for corresponding objects are defined to be coincident with the upper left hand corner of the bounding box for that object, wherein the object's datum point shares the root reference number of the object with an appended 'C.'" '353 patent, col. 17, ll. 57-64.</p>

Term/Phrase in Issue and Asserted Claim(s) in Which Term/Phrase Appears	Defendants' Proposed Construction	Defendants' Citations to Intrinsic Evidence
		<p>“Once the page’s datum point and an object’s datum point are known, a vector between these points is generated for each object in a block 158. With reference to FIG. 4D, in one embodiment, wherein the page datum point corresponds to the upper left and corner of the display frame and is assigned an XY value 266 of 0,0 the vector for a given object may be stored as the XY value of the datum point of that object relative to 0,0, such as a value of 150, 225 (ref. num. 268) for a vector 250D pointing to an object datum 250C, and a value of 150, 425 (ref. num. 270) for a vector 252D pointing to an object datum 252C. In another embodiment, each vector may be stored as XY data relative to a 0, 0 datum point corresponding to the upper left hand corner of the frame the object belongs to. For example, a vector 250D' from a frame datum 214D to object datum 250C is stored as 20, 200 (ref. num. 268'), while a vector 252D from frame datum 214D to object datum 252C is stored as 20, 425. In this embodiment, offset information for each frame relative to a known datum will also be stored, as depicted by a vector 214D.” ‘353 patent, col. 17, l. 65 to col. 18, l. 16.</p> <p>“As shown in FIG. 4G, respective offsets in X and Y, ($-\Delta X$ and $-\Delta Y$ in the Figure) are applied to the starting point of each of the vectors. The vectors are then scaled by a scale factor ‘SF.’ The results of the new vectors are depicted by vectors 250D and 252D”. This produces a new datum for each object’s bounding box that is relative to rendered page datum 262, which remains fixed.” ‘353 patent, col. 19, ll. 57-63.</p> <p>“This foregoing process establishes a starting point (the new datum) for where the content in each object’s bounding box will be rendered. At this point, each object’s bounding box is then drawn from its new datum using the scaling factor. For example, in the original web page 210D (FIG. 4D), bounding box 250B had an X - axes datum of 150 pixels, a Y-axis datum of 225 pixels, and a height and width of 180x350 pixels. In contrast, after being offset and scaled, bounding box 250B' has an X-axis datum of 150*SF-ΔX, a Y-axis datum of 225*SF-ΔY, and a height and width of 180*SFx350*SF.” ‘353 patent, col. 20, ll. 7-17.</p>

Term/Phrase in Issue and Asserted Claim(s) in Which Term/Phrase Appears	Defendants' Proposed Construction	Defendants' Citations to Intrinsic Evidence
		<p>“Returning to the flowchart of FIG. 6, once the vectors and bounding boxes are offset and scaled, content corresponding to objects having at least a portion of their bounding boxes falling within the display limit bounding box is retrieved from the client device's display list in a block 168.” ‘353 patent, col. 20, ll. 18-22.</p> <p>‘353 patent Claim 5: “The wireless device of claim 1, wherein execution of the instructions performs further operations comprising: parsing markup language code to determine the original page layout of display content within the Web page, wherein the original page layout defines a layout location for a plurality of objects, including at least one of text objects, graphic layout objects, or graphic image objects included in the Web page; defining a primary datum corresponding to the original page layout; and, for each object, defining an object datum corresponding to the layout location for the object; generating a vector from the primary datum to the object datum for the object” See <i>also</i> ‘353 patent Claims 24, 174, 220, 239 and 274.</p> <p>‘353 patent Claim 58: “The mobile hand-held device of claim 36, wherein execution of the instructions performs further operations comprising: parsing HTML-based code corresponding to the retrieved Web page to determine the original page layout of the content on the Web page; logically grouping selected content into objects; defining a primary datum corresponding to the original page layout; and, for each object, defining an object datum corresponding to a layout location datum for the object's associated display content; generating a vector from the primary datum to the object datum for the object”</p> <p>‘353 patent Claim 118: “A method, comprising: rendering a browser interface on a display of a device via which a user is enabled to request access to a Web page, the Web page comprising HTML-based Web content having an original format defining an original width and height of the Web page and an original page layout, functionality, and design of content on the Web page; in response to a user request to access the Web page, retrieving the Web page via the device, and translating at least a portion of the HTML-based Web content from its original format into scalable content that supports a scalable resolution-independent representation of</p>

Term/Phrase in Issue and Asserted Claim(s) in Which Term/Phrase Appears	Defendants' Proposed Construction	Defendants' Citations to Intrinsic Evidence
		<p>the Web page”</p> <p>‘353 patent Claim 138: “The method of claim 118, further comprising: parsing markup language code corresponding to the retrieved Web page to determine the original page layout of the content on the Web page; logically grouping selected content into objects; defining a primary datum corresponding to the original page layout; and, for each object, defining an object datum corresponding to a layout location datum for the object's associated display content; generating a vector from the primary datum to the object datum for the object”</p> <p>‘926 patent Claim 21: “The mobile device of claim 1, wherein translating the HTML-based Web content to produce scalable vector-based page layout information comprises: processing the HTML-based Web content with a rendering engine to generate page layout information corresponding to the original page layout as interpreted by the rendering engine; and employing the page layout information to produce scalable vector-based page layout information.” <i>See also</i>, ‘926 patent Claim 51.</p> <p>‘926 patent Claim 34: “The mobile phone of claim 30, wherein execution of the instructions performs further operations comprising: parsing and processing markup language code associated with the Web page to determine the original page layout of display content within the Web page, wherein the original page layout defines a layout location for a plurality of objects, including text objects, graphic layout objects, and/or graphic image objects included in the Web page; defining a primary datum corresponding to the original page layout; and, for each object, defining an object datum corresponding to the layout location for the object; generating a vector from the primary datum to the object datum for the object”</p> <p>‘926 patent Claim 72: “The mobile device of claim 52, wherein execution of the instructions performs further operations comprising: parsing HTML-based code corresponding to the received Web content to identify content on the Web page; logically grouping selected content into objects; defining a primary datum corresponding to the original page layout; and, for each object, defining an object</p>

Term/Phrase in Issue and Asserted Claim(s) in Which Term/Phrase Appears	Defendants' Proposed Construction	Defendants' Citations to Intrinsic Evidence
		<p>datum corresponding to a layout location datum for the object's associated display content; generating a vector from the primary datum to the object datum for the object”</p> <p>“For example, in one embodiment the page layout information including object datums (i.e., XY points) determined by the operations in the flowchart of FIG. 5 (and as illustrated in the XY page layout example of FIGs. 4A-D) are mapped as vectors to a virtual display area. The virtual display area comprises a virtual (XY) coordinate system that is resolution independent, and the datum points are mapped to the virtual coordinate system as point vectors (i.e., from XY pixel locations in the original page layout (as interpreted by the rendering engine) to XY coordinate points in the virtual coordinate system).” ‘926 patent Amendment and Response to Office Action dated Apr. 21, 2010, at 31.</p> <p>“This process begins by employing a rendering engine to determine page layout information that includes information from which the location for each display object (i.e., HTML element or block having content that is to be displayed) can be determined. In general, the page layout information includes information from which at least one of a datum or bounding box for each display object can be determined. In one embodiment, the page layout is interpreted using the rendering engine based on a default page width in pixels, and the location of the page layout objects is interpreted by the rendering engine are defined by corresponding pixel locations on the page. Accordingly, at this point the page layout information, such as a datum, is defined by corresponding pixel locations in a two-dimensional (XY) pixel-based coordinate space used by the rendering engine.” ‘926 patent Comments on Statement of Reasons for Allowance dated Sept. 9, 2010, at 43; <i>see also id.</i> at 42-60 (entire section designated as “Scalable Vector-based Page Layout Information”).</p> <p>“In summary, in one non-limiting embodiment the element of ‘translating the HTML-based Web content to produce scalable vector-based page layout information’ includes employing a rendering engine to interpret the page layout and mapping selected page layout coordinates from a pixel-based coordinate system to a</p>

Term/Phrase in Issue and Asserted Claim(s) in Which Term/Phrase Appears	Defendants' Proposed Construction	Defendants' Citations to Intrinsic Evidence
<p>11 layout location datum</p> <p>'353 Patent: 58, 59 by reference to 58, 139 by reference to 138</p> <p>'926 Patent: 72</p>	<p>point defined at a fixed X,Y coordinate on the full-size web page that corresponds to an object</p>	<p>vector based virtual coordinate space. Once the layout information is defined by vector-based points in the virtual coordinate space, the coordinates of the layout information can be scaled to produce scaled page layout information. Scaled views of Web page content can then be generated by rendering display objects using applicable scale factors at locations defined the scaled coordinates corresponding to each display object.” ‘926 patent Comments on Statement of Reasons for Allowance dated Sept. 9, 2010, at 59-60; <i>see also id.</i> at 42-60 (entire section designated as “Scalable Vector-based Page Layout Information”).</p> <p>“datum point defined for page / “page datum” and bounding box for each object (17:45-64), relative to a consistent point of reference” ‘353 patent Order Granting <i>Inter Partes</i> Reexamination dated Aug. 8, 2011, at 15.</p> <p>“FIG. 4A is a representation of an exemplary web page has displayed on a conventional browser;</p> <p>FIG. 4B is a schematic diagram illustrates various objects that are generated based on the HTML code of the web page of FIG. 4A;</p> <p>FIG. 4C is a schematic diagram illustrating a set of vectors and bounding boxes corresponding to the objects generated in FIG. 4B;</p> <p>FIG. 4D is a schematic diagram illustrating how various vectors and bounding boxes may be defined in accordance with the invention;</p> <p>FIG. 4E is a representation of the web page of FIG. 4A after it has been offset and scaled in accordance with the invention;</p> <p>FIG. 4F is a schematic diagram illustrating new datum points and bounding boxes corresponding to the scaled and offset web page;</p> <p>FIG. 4G is a schematic diagram illustrating new vectors and bounding box parameters for a pair of objects in the scaled and offset web page;</p> <p>FIG. 5 is a flowchart illustrating the logic used by the invention when translating content into a scalable vector representation of that content;</p> <p>FIG. 6 is a flowchart illustrating client-side operations that are performed to</p>

Term/Phrase in Issue and Asserted Claim(s) in Which Term/Phrase Appears	Defendants' Proposed Construction	Defendants' Citations to Intrinsic Evidence
		<p>create a rendered display page based on the translated content the client receives and user-input;" '353 patent, col. 3, ll. 31-56; <i>see also id.</i> Fig. 4A, Fig. 4B, Fig. 4C, Fig. 4D, Fig. 4E, Fig. 4F, Fig. 4G, Fig. 5, Fig. 6.</p> <p>"Web page documents comprise HTML code that is parsed, interpreted, and rendered by a browser. An HTML document comprises a plurality of HTML "markup" elements (tags) with corresponding attributes, that are used to describe the layout and formatting of various objects, including plain text and graphic objects, embedded between tag pairs." '353 patent, col. 15, ll. 19-24.</p> <p>"Next, in a block 154, the page layout is defined based on the bounding boxes. In actuality, generation of the page layout information is performed in conjunction with defining the boundary boxes for the objects, wherein the location of a given object is based on the location of other related (e.g., if within a table) or non-related objects corresponding to HTML content that have been previously parsed." '353 patent, col. 17, ll.</p> <p>"At this point, the present invention deviates substantially from the prior art by using the various object layout data generated during the pre-rendering process to generate a scalable vector representation of the original page content." '353 patent, col. 17, ll. 42-45.</p> <p>"In general, the datum points for each object may also be located any place on the object, as long as the object datum points are used in a predictable manner. For example, as depicted in FIG. 4C, various datum points for corresponding objects are defined to be coincident with the upper left hand corner of the bounding box for that object, wherein the object's datum point shares the root reference number of the object with an appended 'C.'" '353 patent, col. 17, ll. 57-64.</p> <p>"Once the page's datum point and an object's datum point are known, a vector between these points is generated for each object in a block 158. With reference to FIG. 4D, in one embodiment, wherein the page datum point corresponds to the upper left and corner of the display frame and is assigned an XY value 266 of 0,0</p>

Term/Phrase in Issue and Asserted Claim(s) in Which Term/Phrase Appears	Defendants' Proposed Construction	Defendants' Citations to Intrinsic Evidence
		<p>the vector for a given object may be stored as the XY value of the datum point of that object relative to 0,0, such as a value of 150, 225 (ref. num. 268) for a vector 250D pointing to an object datum 250C, and a value of 150, 425 (ref. num. 270) for a vector 252D pointing to an object datum 252C. In another embodiment, each vector maybe stored as XY data relative to a 0, 0 datum point corresponding to the upper left hand corner of the frame the object belongs to. For example, a vector 250D' from a frame datum 214D to object datum 250C is stored as 20, 200 (ref. num. 268'), while a vector 252D from frame datum 214D to object datum 252C is stored as 20, 425. In this embodiment, offset information for each frame relative to a known datum will also be stored, as depicted by a vector 214D." '353 patent, col. 17, l. 65 to col. 18, l. 16.</p> <p>"This foregoing process establishes a starting point (the new datum) for where the content in each object's bounding box will be rendered. At this point, each object's bounding box is then drawn from its new datum using the scaling factor. For example, in the original web page 210D (FIG. 4D), bounding box 250B had an X - axes datum of 150 pixels, a Y-axis datum of 225 pixels, and a height and width of 180x350 pixels. In contrast, after being offset and scaled, bounding box 250B' has an X-axis datum of 150*SF-ΔX, a Y-axis datum of 225*SF-ΔY, and a height and width of 180*SFx350*SF." '353 patent, col. 20, ll. 7-17.</p> <p>"Returning to the flowchart of FIG. 6, once the vectors and bounding boxes are offset and scaled, content corresponding to objects having at least a portion of their bounding boxes falling within the display limit bounding box is retrieved from the client device's display list in a block 168." '353 patent, col. 20, ll. 18-22.</p> <p>'353 patent Claim 5: "The wireless device of claim 1, wherein execution of the instructions performs further operations comprising: parsing markup language code to determine the original page layout of display content within the Web page, wherein the original page layout defines a layout location for a plurality of objects, including at least one of text objects, graphic layout objects, or graphic image objects included in the Web page; defining a primary datum corresponding to the original page layout; and, for each object, defining an object datum corresponding</p>

Term/Phrase in Issue and Asserted Claim(s) in Which Term/Phrase Appears	Defendants' Proposed Construction	Defendants' Citations to Intrinsic Evidence
		<p>to the layout location for the object; generating a vector from the primary datum to the object datum for the object” See <i>also</i> ‘353 patent Claims 24, 174, 220, 239 and 274.</p> <p>‘353 patent Claim 58: “The mobile hand-held device of claim 36, wherein execution of the instructions performs further operations comprising: parsing HTML-based code corresponding to the retrieved Web page to determine the original page layout of the content on the Web page; logically grouping selected content into objects; defining a primary datum corresponding to the original page layout; and, for each object, defining an object datum corresponding to a layout location datum for the object's associated display content; generating a vector from the primary datum to the object datum for the object”</p> <p>‘353 patent Claim 118: “A method, comprising: rendering a browser interface on a display of a device via which a user is enabled to request access to a Web page, the Web page comprising HTML-based Web content having an original format defining an original width and height of the Web page and an original page layout, functionality, and design of content on the Web page; in response to a user request to access the Web page, retrieving the Web page via the device, and translating at least a portion of the HTML-based Web content from its original format into scalable content that supports a scalable resolution-independent representation of the Web page”</p> <p>‘353 patent Claim 138: “The method of claim 118, further comprising: parsing markup language code corresponding to the retrieved Web page to determine the original page layout of the content on the Web page; logically grouping selected content into objects; defining a primary datum corresponding to the original page layout; and, for each object, defining an object datum corresponding to a layout location datum for the object's associated display content; generating a vector from the primary datum to the object datum for the object”</p> <p>‘926 patent Claim 21: “The mobile device of claim 1, wherein translating the HTML-based Web content to produce scalable vector-based page layout</p>

Term/Phrase in Issue and Asserted Claim(s) in Which Term/Phrase Appears	Defendants' Proposed Construction	Defendants' Citations to Intrinsic Evidence
		<p>information comprises: processing the HTML-based Web content with a rendering engine to generate page layout information corresponding to the original page layout as interpreted by the rendering engine; and employing the page layout information to produce scalable vector-based page layout information.” See <i>also</i>, ‘926 patent Claim 51.</p> <p>‘926 patent Claim 34: “The mobile phone of claim 30, wherein execution of the instructions performs further operations comprising: parsing and processing markup language code associated with the Web page to determine the original page layout of display content within the Web page, wherein the original page layout defines a layout location for a plurality of objects, including text objects, graphic layout objects, and/or graphic image objects included in the Web page; defining a primary datum corresponding to the original page layout; and, for each object, defining an object datum corresponding to the layout location for the object; generating a vector from the primary datum to the object datum for the object”</p> <p>‘926 patent Claim 72: “The mobile device of claim 52, wherein execution of the instructions performs further operations comprising: parsing HTML-based code corresponding to the received Web content to identify content on the Web page; logically grouping selected content into objects; defining a primary datum corresponding to the original page layout; and, for each object, defining an object datum corresponding to a layout location datum for the object's associated display content; generating a vector from the primary datum to the object datum for the object”</p> <p>“It is common practice to lay out web page content using a page origin of 0,011 at the upper left hand corner of the page. Accordingly, web page rendering engines such as Gecko typically employ a pixel-based coordinate system having an origin at 0,0 with X values increasing to the right, and Y values increasing downward. The coordinates of the pixel-based coordinate system are integer values corresponding to the location of the pixels in a rendered page.” ‘926 patent Comments on Statement of Reasons for Allowance dated Sept. 9, 2010, at 47; see <i>also id.</i> at 42-60 (entire section designated as “Scalable Vector-based Page Layout Information”).</p>

Term/Phrase in Issue and Asserted Claim(s) in Which Term/Phrase Appears	Defendants' Proposed Construction	Defendants' Citations to Intrinsic Evidence
		<p>“Optionally, the page layout datum may be considered at l, l, representing the X, Y location of the pixel at the upper left hand corner of a content drawing area used to render page content in a browser.” ‘926 patent Comments on Statement of Reasons for Allowance dated Sept. 9, 2010, at 47 n.11; <i>see also id.</i> at 42-60 (entire section designated as “Scalable Vector-based Page Layout Information”).</p> <p>“Since the content in this example is laid out relative to the body block, the upper left hand corner of the body block container box is defined as the 0, 0 datum for the examples herein rather than the upper left hand corner of the page” ‘926 patent Comments on Statement of Reasons for Allowance dated Sept. 9, 2010, at 49.</p> <p>“For example, in one embodiment the page layout information including object datums (i.e., XY points) determined by the operations in the flowchart of FIG. 5 (and as illustrated in the XY page layout example of FIGs. 4A-D) are mapped as vectors to a virtual display area.” ‘926 patent Amendment and Response to Office Action dated Apr. 21, 2010, at 31.</p> <p>“This process begins by employing a rendering engine to determine page layout information that includes information from which the location for each display object (i.e., HTML element or block having content that is to be displayed) can be determined. In general, the page layout information includes information from which at least one of a datum or bounding box for each displayed object can be determined. In one embodiment, the page layout is interpreted using the rendering engine based on a default page width in pixels, and the location of the page layout objects is interpreted by the rendering engine are defined by corresponding pixel locations on the page. Accordingly, at this point the page layout information, such as a datum, is defined by corresponding pixel locations in a two-dimensional (XY) pixel-based coordinate space used by the rendering engine.” ‘926 patent Comments on Statement of Reasons for Allowance dated Sept. 9, 2010, at 43.</p> <p>“Browser rendering engines typically determine page layout based on the size of the browser window content area (i.e., the portion of the browser application</p>

Term/Phrase in Issue and Asserted Claim(s) in Which Term/Phrase Appears	Defendants' Proposed Construction	Defendants' Citations to Intrinsic Evidence
		<p>window in which the page content is rendered). In particular, the width of the content area is employed.” ‘926 patent Comments on Statement of Reasons for Allowance dated Sept. 9, 2010, at 47; <i>see also id.</i> at 42-60 (entire section designated as “Scalable Vector-based Page Layout Information”).</p> <p>“In summary, in one non-limiting embodiment the element of ‘translating the HTML-based Web content to produce scalable vector-based page layout information’ includes employing a rendering engine to interpret the page layout and mapping selected page layout coordinates from a pixel-based coordinate system to a vector based virtual coordinate space. Once the layout information is defined by vector-based points in the virtual coordinate space, the coordinates of the layout information can be scaled to produce scaled page layout information. Scaled views of Web page content can then be generated by rendering display objects using applicable scale factors at locations defined the scaled coordinates corresponding to each display object.” ‘926 patent Comments on Statement of Reasons for Allowance dated Sept. 9, 2010, at 59-60; <i>see also id.</i> at 42-60 (entire section designated as “Scalable Vector-based Page Layout Information”).</p> <p>“datum point defined for page / “page datum” and bounding box for each object (17:45-64), relative to a consistent point of reference” ‘353 patent Order Granting <i>Inter Partes</i> Reexamination dated Aug. 8, 2011, at 15.</p>
12 enabling the user to zoom and pan a view of the Web page ‘353 Patent: 317	using the scalable content to allow the user to resize and move around the web page	<p>“The mobile devices employ software based processing of original Web content, including HTML-based content, XML, cascade style sheets, etc. to generate scalable content. The scalable content and/or data derived therefrom are then employed to enable the Web content to be rapidly rendered, zoomed, and panned.” ‘353 patent, Abstract.</p> <p>“The mobile devices employ novel processing of original Web content, including HTML-based content, XML, cascade style sheets, etc. to generate scalable content. The scalable content and/or data derived therefrom are then employed to enable the Web content to be rapidly rendered, zoomed, and panned.” ‘353 patent, col. 2, ll. 26-31.</p>

Term/Phrase in Issue and Asserted Claim(s) in Which Term/Phrase Appears	Defendants' Proposed Construction	Defendants' Citations to Intrinsic Evidence
		<p>“As discussed above, the present invention supports a wide variety of clients, including land-based clients and wireless clients. Each client requires some client-side software that enables the scalable vector content data provided to it to be rendered at a user-selectable scale factor and offset on the client's display, such as a monitor or built-in LCD screen.” ‘353 patent, col. 18, ll. 40-46.</p> <p>“As explained below, through use of the invention's scalable vector representation and client-side processing, users are enabled to view the entire content of billions of existing Web pages using handheld devices in a simple and reasonable way.” ‘353 col. 18, l. 63 to col. 19, l. 2.</p> <p>“In a block 162, a display list of vectors is built. This process is well known in the CAD arts, and is enabling rapid zooming of vector-based objects. . . . Based on various user interactions with the user-interface of the client, the user is enabled to control the zoom (size) and offset of the rendered page.” ‘353 col. 19, ll. 20-26.</p> <p><i>See also</i> “scalable content” above.</p>
13 original original width and height of the Web page	as viewed on a conventional desktop browser, such as Internet Explorer or Netscape Navigator, when the browser window is set to the full width resolution for which the page was designed the width and height of the Web page as viewed on a conventional desktop browser, such as Internet Explorer or Netscape Navigator, when the browser window is set to the full width resolution for which the page was designed	<p>“In accordance with aspects of the invention, mobile devices enabled to support resolution-independent scalable display of Internet (Web) content to allow Web pages to be scaled (zoomed) and panned for better viewing on smaller screen sizes are disclosed. The mobile devices employ novel processing of original Web content, including HTML-based content, XML, cascade style sheets, etc. to generate scalable content. The scalable content and/or data derived therefrom are then employed to enable the Web content to be rapidly rendered, zoomed, and panned. Moreover, the rendered displays provide substantially the same or identical layout as the original Web page, enabling users to easily navigate to selected content and features on familiar Web pages. Display lists may also be employed to provide further enhancements in rendering speed. Additionally, hardware-based programmed logic may also be employed to facilitate various operations.” ‘353 patent, col. 2, ll. 22-38.</p> <p>“For illustrative purposes, it will be assumed for the present example that the desired content comprises a web page that is stored on web server 26. Typically,</p>

Term/Phrase in Issue and Asserted Claim(s) in Which Term/Phrase Appears	Defendants' Proposed Construction	Defendants' Citations to Intrinsic Evidence
<p>original page layout, functionality, and design</p> <p>original page layout</p> <p>'353 Patent: 33 by reference to 1, 43 by reference to 36, 48 by reference to 36, 52 by reference to 36, 58, 59 by reference to 36, and 58, 66 by reference to 36, 139 by reference to 118, 183 by reference to 149, 283 by reference to 252</p> <p>'926 Patent: 30, 31 by reference to 30, 40 by reference to 30, 41 by reference to 30, 43 by reference to 30, 55 by reference to 52, 59 by reference to 52, 72, 75 by reference to 52</p>	<p>the page layout, functionality, and design of the Web page as viewed on a conventional desktop browser, such as Internet Explorer or Netscape Navigator, when the browser window is set to the full width resolution for which the page was designed</p> <p>the page layout of the Web page as viewed on a conventional desktop browser, such as Internet Explorer or Netscape Navigator, when the browser window is set to the full width resolution for which the page was designed</p>	<p>when the requested content comprises a web page, the content may be retrieved using conventional web content retrieval techniques, such as that employed by various modem browser clients, including Netscape Navigator and Internet Explorer." '353 patent, col. 7, ll. 25-32.</p> <p>"The proxy server responds to client content requests by delivering content in one of the requested formats, by retrieving the content in an appropriate format from its cache, or from an upstream content source (again using standard HTTP content negotiation features), or by translating upstream content from a supported original format to SVF or the client bitmap format." '353 patent, col. 11, ll. 3-9.</p> <p>"As discussed above, HTML translator 58 creates a scalable vector representation of the original HTML content of a requested web page." '353 patent, col. 12, ll. 12-14.</p> <p>"At this point, the present invention deviates substantially from the prior art by using the various object layout data generated during the pre-rendering process to generate a scalable vector representation of the original page content." '353 patent, col. 17, ll. 42-45.</p> <p>"By enabling original content from a web site to be displaced in such a resolution-independent manner, users will be able to view content in a manner that did not previously exist, greatly enhancing the user experience. For example, in some implementations the client may be a personal computer (PC). Using a least-common denominator approach, many web pages are designed for a smaller resolution (for example 640.times.480 pixels, a minimum resolution commonly supported by nearly all PC's, including legacy PC's) than the resolution provided by the video output capabilities available with many of today's PC's, such as 1024.times.768 pixels, 1280.times.1024 pixels, and even 1600.times.1200 pixels. As a result, when these web pages are displayed on a high-resolution display, they occupy only a portion of the display, making portions of the pages, especially those portions containing small text, difficult to read. By enabling users to selectively magnify the entire page, these design flaws are easily overcome. Alternatively, the</p>

Term/Phrase in Issue and Asserted Claim(s) in Which Term/Phrase Appears	Defendants' Proposed Construction	Defendants' Citations to Intrinsic Evidence
		<p>client may be a small device, such as a hand held computer or a cell phone, which has a smaller display resolution than common Web pages are designed for. As explained below, through use of the invention's scalable vector representation and client-side processing, users are enabled to view the entire content of billions of existing Web pages using hand-held devices in a simple and reasonable way." '353 patent, col. 18, l. 47 to col. 19, l. 3.</p> <p>"Mobile devices enabled to support resolution-independent scalable display of Internet (Web) content to allow Web pages to be scaled (zoomed) and panned for better viewing on smaller screen sizes. The mobile devices employ software-based processing of original Web content, including HTML-based content, XML, cascade style sheets, etc. to generate scalable content. The scalable content and/or data derived therefrom are then employed to enable the Web content to be rapidly rendered, zoomed, and panned. Moreover, the rendered displays provide substantially the same or identical layout as the original Web page, enabling users to easily navigate to selected content and features on familiar Web pages." '353 patent, Abstract.</p> <p>"The pending claims in the present application are patentably distinct from the issued claims in the '099 patent. Each of the pending claims includes the element of retrieving HTML-based Web page content corresponding to the requested Web page - that is, <i>they begin with Web page content in its original form as stored on (a) Web server(s) and made available for download to conventional desktop browsers.</i>" '353 patent Amendment and Response to Office Action dated Dec. 9, 2007, Remarks at 83 (emphasis in original).</p> <p>"However, [Roy's] HTML element has nothing to do with defining an original width and height of a Web Page, but rather merely defines the Width and Height to be used by an embedded MapGuide Viewer when the Web page is initially rendered." '353 patent Amendment and Response to Office Action dated Dec. 9, 2007, Remarks at 121.</p> <p>"Under the claimed inventions of claims 383-390 and 393, users are enabled to</p>

Term/Phrase in Issue and Asserted Claim(s) in Which Term/Phrase Appears	Defendants' Proposed Construction	Defendants' Citations to Intrinsic Evidence
		<p>view, zoom, and pan the HTML-based Web page content of substantially any Web page in a manner that preserves the original layout and attributes of the Web page content. This capability will generally be dependent on the rendering engine compatibility with the Web page definition (as defined by the Web page's corresponding HTML). That is, this will generally depend on whether the rendering engine can render the original HTML-based Web page content appropriately at its original resolution or as originally defined by the Web page's HTML. Under the principles and teachings of the present disclosure, Web pages that can be rendered (by the applicable rendering engine) at the original resolution or as originally defined by the Web page's corresponding HTML are enabled to be viewed, zoomed, and panned in a manner that preserves the original layout and attributes of the Web page (as rendered at the original resolution).” ‘353 patent Supplemental Amendment dated Jan. 12, 2008, Remarks at 66.</p> <p>“The new claim language introduces the term ‘functionality.’ Preserving functionality generally pertains to preserving the interoperability of various HTML-based Web page content, such as hyperlinks and UI controls such as input forms defined via corresponding HTML-based code. It is noted that the HTML code defining a Web page's overall layout, functionality and design does not define how a user interaction with the Web content is to be supported, but rather defines the existence of a corresponding function within the Web content to support the interaction. For example, a hyperlink definition within a Web page merely defines a link (hyperlink reference of href) to corresponding content, it does not define how the hyperlink associated control is to appear on the screen nor how the hyperlink is to be activated. That is up to the browser's implementation, which varies by browser.” ‘353 patent Supplemental Amendment dated May 20, 2008, Remarks at 90; <i>see also id.</i> at 65-96.</p> <p>‘353 patent Supplemental Information Disclosure Statement Remarks dated Jan. 22, 2008, at 2-18.</p> <p>‘353 patent Information Disclosure Statement dated June 8, 2001.</p>

Term/Phrase in Issue and Asserted Claim(s) in Which Term/Phrase Appears	Defendants' Proposed Construction	Defendants' Citations to Intrinsic Evidence
<p>14 fit across</p> <p>fit across the display</p> <p>rendered to fit across the display</p> <p>displayed to fit across the touch-sensitive display</p> <p>displayed to fit across at least one of a width and height of a display area of the touch-sensitive display</p> <p>'353 Patent: 33 by reference to 1, 48, 52 by reference to 51, 139 by reference to 118</p> <p>'926 Patent: 40, 41</p>	<p>fully fill</p> <p>fully fill the display</p> <p>rendered to fully fill the display</p> <p>displayed to fully fill the touch-sensitive display</p> <p>displayed to fully fill at least one of a width and height of a display area of the touch-sensitive display</p>	<p>App. No. 11/868,124, Remarks dated Nov. 24, 2010, at 18-139.</p> <p>App. No. 11/868,124, Reply to Action of January 24, 2011, dated July 22, 2011, at 14-18.</p> <p>App. No. 11/868,124, Response to Office Action of October 13, 2011, dated Apr. 3, 2012, at 14-19.</p> <p>App. No. 11/868,124, Office Action mailed Apr. 18, 2012, at 2-12.</p> <p>"In addition to directly scaling and offsetting content, the client user-interface software for PDA's provides additional functionality. For instance, a user may select to view a column (results represented in FIG. 7B) by tapping that column with a stylus, as shown in FIG. 7A. Similarly, the user may select to zoom in on an image by tapping the image with the stylus, as shown in FIGS. 8A and 8B, or select to view a paragraph in an article by tapping on the paragraph, as shown in FIGS. 9A and 9B." '353 patent, col. 20, ll. 56-67; <i>see also id.</i> Fig. 8A, Fig. 8B Fig. 9A, Fig. 9B.</p> <p><i>See</i> '353 patent Office Action dated Oct. 23, 2007, at 4-11 (rejecting multiple claims, <i>e.g.</i>, prosecution claim 71, as indefinite for reciting the limitation "substantially").</p> <p><i>See</i> '353 patent Supplemental Amendment dated May 20, 2008 (deleting "substantially" before "across" in multiple claims, <i>e.g.</i>, prosecution claim 71: "scaling the scalable content to render the web page on the display such that the entire a width of the Web page is rendered to fit substantially across the display . . ."); <i>see also id.</i>, Remarks at 67-75.</p> <p><i>See</i> '926 patent Office Action dated Apr. 1, 2010, at 4 (rejecting multiple claims, <i>e.g.</i>, prosecution claim 13, as indefinite for reciting the limitation "substantially").</p> <p><i>See</i> '926 patent Amendment and Response to Office Action dated Apr. 21, 2010 (deleting "substantially" before "across" in multiple claims, <i>e.g.</i>, prosecution claim</p>

Term/Phrase in Issue and Asserted Claim(s) in Which Term/Phrase Appears	Defendants' Proposed Construction	Defendants' Citations to Intrinsic Evidence
<p>15</p> <p>tapping</p> <p>'353 Patent: 52, 317</p> <p>'926 Patent: 40, 41</p>	<p>making contact with the display using a stylus</p>	<p>13: "content corresponding to the selected column is displayed substantially to fit across the touch-sensitive display"); <i>see also id.</i>, Remarks at 28.</p> <p>"In addition to directly scaling and offsetting content, the client user-interface software for PDA's provides additional functionality. For instance, a user may select to view a column (results represented in FIG. 7B) by tapping that column with a stylus, as shown in FIG. 7A. Similarly, the user may select to zoom in on an image by tapping the image with the stylus, as shown in FIGS. 8A and 8B, or select to view a paragraph in an article by tapping on the paragraph, as shown in FIGS. 9 A and 9B. It is noted that in some instances, the display of the paragraph may be reformatted to fit the characteristics of the display, rather than following the original format in the zoomout view." '353 patent, col. 20, ll. 56-67.</p>
<p>16</p> <p>preserve(s) / preserved / preserving / preservation</p> <p>'353 Patent: 33 by reference to 1, 43 by reference to 36, 48 by reference to 36, 52 by reference to 36, 59 by reference to 36, 66, 139 by reference to 118, 183 by reference to 149, 283 by reference to 252</p> <p>'926 Patent: 30, 31 by reference to 30, 40 by reference to 30, 41 by reference to 30, 43 by reference to 30, 55 by reference to 52, 59 by reference to 52, 72 by reference to 52, 75 by reference to 52</p>	<p>These terms are indefinite and therefore cannot be construed.</p>	<p>"In accordance with aspects of the invention, mobile devices enabled to support resolution-independent scalable display of Internet (Web) content to allow Web pages to be scaled (zoomed) and panned for better viewing on smaller screen sizes are disclosed.... Moreover, the rendered displays provide substantially the same or identical layout as the original Web page, enabling users to easily navigate to selected content and features on familiar Web pages. Display lists may also be employed to provide further enhancements in rendering speed. Additionally, hardware-based programmed logic may also be employed to facilitate various operations." '353 patent, col. 2, ll. 22-38.</p> <p>"In addition to directly scaling and offsetting content, the client user-interface software for PDA's provides additional functionality. For instance, a user may select to view a column (results represented in FIG. 7B) by tapping that column with a stylus, as shown in FIG. 7A. Similarly, the user may select to zoom in on an image by tapping the image with the stylus, as shown in FIGS. 8A and 8B, or select to view a paragraph in an article by tapping on the paragraph, as shown in FIGS. 9 A and 9B. It is noted that in some instances, the display of the paragraph may be reformatted to fit the characteristics of the display, rather than following the original format in the zoomout view." '353 patent, col. 20, ll. 56-67.</p> <p>"First, as explained below in further detail, one of skill in the art would appreciate the fact that the same Web page (in its original HTML-based format) may not be</p>

Term/Phrase in Issue and Asserted Claim(s) in Which Term/Phrase Appears	Defendants' Proposed Construction	Defendants' Citations to Intrinsic Evidence
		<p>rendered identically by conventional desktop browsers due to various factors. Moreover, one of skill in the art would understand that due to the combination of both incrementally scalable and mathematically scalable content that could be derived by translating the original HTML-based Web page content, the corresponding scaled page may not result in an exact scaling of the Web page content. For example, under embodiments of the present application the overall layout of a web page may be scaled by a non-integer mathematical factor, such as 40%. However, since Web page text definitions and most browser (more accurately Operating System) text definitions are defined by incremental font sizes (e.g., a font size defined by an integer, such as 6, 8, 9, 10, 11, 12, 14, etc.), a mathematical (i.e., vector) scaling of a Web page layout may produce a slightly different scaling of the Web page text content. . . . This does not imply that it is not possible to have truly (i.e., mathematically) scalable fonts, as the use of such fonts are disclosed in the present application. However, use of fixed size fonts are also disclosed, which may result in scaled Web pages having a slightly different layout (yet substantially similar) than a non-scaled Web page (i.e., the layout of a Web page as defined by its original HTML code).” ‘353 patent Amendment and Response to Office Action dated Dec. 9, 2007, Remarks at 63-64.</p> <p>“Under the principles and teachings of the present disclosure, Web pages that can be rendered (by the applicable rendering engine) at the original resolution or as originally defined by the Web page's corresponding HTML are enabled to be viewed, zoomed, and panned in a manner that preserves the original layout and attributes of the Web page (as rendered at the original resolution).” ‘353 patent Supplemental Amendment dated Jan. 12, 2008, Remarks at 66.</p> <p>‘353 patent Supplemental Information Disclosure Statement Remarks dated Jan. 22, 2008, at 2-18.</p> <p>‘353 patent Information Disclosure Statement dated June 8, 2001.</p> <p>App. No. 11/868,124, Remarks dated Nov. 24, 2010, at 18-139.</p>

Term/Phrase in Issue and Asserted Claim(s) in Which Term/Phrase Appears	Defendants' Proposed Construction	Defendants' Citations to Intrinsic Evidence
17 machine-readable medium '353 Patent: 283	a medium for electronic instructions, such as a floppy disk, optical disk, CD-ROM, magneto-optical disk, ROM, RAM, EPROM, EEPROM, magnetic or optical card, flash memory, or carrier wave	App. No. 11/868,124, Reply to Action of January 24, 2011, dated July 22, 2011, at 14-18. App. No. 11/868,124, Response to Office Action of October 13, 2011, dated Apr. 3, 2012, at 14-19. App. No. 11/868,124, Office Action mailed Apr. 18, 2012, at 2-12. "The present invention may be provided as a computer program product that may include one or more machine readable mediums having stored thereon instructions, which may be used to program a computer (or other electronic devices) to perform a process according to the present invention. The machine-readable medium may include, but is not limited to, floppy diskettes, optical disks, CD-ROMs, and magneto-optical disks, ROMs, RAMs, EPROMs, EEPROMs, magnetic or optical cards, flash memory, or other type of media/machine-readable medium suitable for storing electronic instructions. Moreover, the present invention may also be downloaded as a computer program product, wherein the program may be transferred from a remote computer (e.g., a server) to a requesting computer (e.g., a client) by way of data signals embodied in a carrier wave or other propagation medium via a communication link (e.g., a modem or network connection). Accordingly, herein, a carrier wave shall be regarded as comprising a machine-readable medium." '353 patent, col. 4, ll. 24-41.
18 storage means '353 Patent: 33 by reference to 1 '926 Patent: 30, 31 by reference to 30, 40 by reference to 30, 41 by reference to 30, 43 by reference to 30	Function: storing a plurality of instructions <u>Corresponding structure</u> : a floppy disk, optical disk, CD-ROM, magneto-optical disk, ROM, RAM, EPROM, EEPROM, magnetic or optical card, flash memory or carrier wave	"The present invention may be provided as a computer program product that may include one or more machine readable mediums having stored thereon instructions, which may be used to program a computer (or other electronic devices) to perform a process according to the present invention. The machine-readable medium may include, but is not limited to, floppy diskettes, optical disks, CD-ROMs, and magneto-optical disks, ROMs, RAMs, EPROMs, EEPROMs, magnetic or optical cards, flash memory, or other type of media/machine-readable medium suitable for storing electronic instructions. Moreover, the present invention may also be downloaded as a computer program product, wherein the program may be transferred from a remote computer (e.g., a server) to a requesting computer (e.g., a client) by way of data signals embodied in a carrier wave or other propagation medium via a communication link (e.g., a modem or network connection).

Term/Phrase in Issue and Asserted Claim(s) in Which Term/Phrase Appears	Defendants' Proposed Construction	Defendants' Citations to Intrinsic Evidence
<p>19 processing means</p> <p>'353 Patent: 33 by reference to 1</p>	<p>Function #1: rendering a browser interface via which a user is enabled to request access to an original Web page, the Web page comprising HTML-based Web content having an original format defining an original width and height of the Web page and an original page layout, functionality, and design of content on the Web page</p> <p><u>Structure corresponding to Function #1</u>: Indefinite, as the specification does not disclose any algorithm for rendering a browser interface.</p> <p><u>Function #2</u>: retrieving the Web page via the wireless communication means</p>	<p>Accordingly, herein, a carrier wave shall be regarded as comprising a machine-readable medium." '353 patent, col. 4, ll. 24-41.</p> <p>"A data storage device 507 such as a magnetic disk or optical disc and its corresponding drive may also be coupled to bus 501 for storing information and instructions." '353 patent, col. 21, ll. 33-35.</p> <p>"For example, according to one embodiment of the present invention, a cell phone or a hand held computer may comprise only a processor or a micro controller and a memory, such as a micro code ROM or RAM, for storing static or dynamically loaded instructions and/or data." '353 patent, col. 22, ll. 1-6.</p> <p>"FIG. 6 is a flowchart illustrating client-side operations that are performed to create a rendered display page based on the translated content the client receives and user-input." '353 patent, col. 3, ll. 53-55; see also <i>id.</i> Fig. 6.</p>

Term/Phrase in Issue and Asserted Claim(s) in Which Term/Phrase Appears	Defendants' Proposed Construction	Defendants' Citations to Intrinsic Evidence
	<p>Structure corresponding to <u>Function #2</u>: Indefinite, as specification does not disclose any structure corresponding to “wireless communication means.”</p> <p><u>Function #3</u>: translating at least a portion of the HTML-based Web content from its original format into scalable content that supports a scalable resolution independent representation of the Web page that preserves the original page layout, functionality and design of the content defined by its original format when scaled and rendered</p> <p><u>Structure corresponding to Function #3</u>: The algorithm depicted at boxes 150-160 of Fig. 5 (also depicted at box 114 of Fig. 2C), and ‘353 patent col. 3, ll. 50-52, col. 10, ll. 31-55, and col. 15, l. 43 to col. 18, l. 39.</p> <p><u>Function #4</u>: scaling the scalable content to render the Web page on the display such that a width</p>	

Term/Phrase in Issue and Asserted Claim(s) in Which Term/Phrase Appears	Defendants' Proposed Construction	Defendants' Citations to Intrinsic Evidence
	<p>of the Web page is rendered to fit across the display</p> <p>Structure corresponding to <u>Function #4</u>: The algorithm depicted at boxes 160-172 of Fig. 6, and corresponding text at '353 patent col. 3, ll. 53-55 and col. 19, l. 14 to col. 20, l. 47, wherein the scale factor is chosen such that the width of the Web page fits across the display.</p>	
<p>20 wireless communication[s] means</p> <p>'353 Patent: 33 by reference to 1</p> <p>'926 Patent: 30, 31 by reference to 30, 40 by reference to 30, 41 by reference to 30, 43 by reference to 30, 55 by reference to 52, 59 by reference to 52, 72 by reference to 52, 75 by reference to 52</p>	<p><u>Functions (the appropriate function depends upon the specific claim):</u></p> <p>facilitating wireless communication with a network that supports access to the Internet</p> <p>('353 patent claim 33 by reference to claim 1)</p> <p>facilitating communication with a mobile service provider network via which Web content may be accessed</p> <p>('926 patent claim 30 and claims 31, 40, 41, and 43 by reference to claim 30)</p> <p>facilitating wireless</p>	<p>"Currently, in the United States, wireless clients generally access Internet 24 by using the Wireless Application Protocol (WAP). In Japan, the most popular access means is NTT DoCoMo's i-Mode wireless protocol. In addition to these wireless standards, new standards are anticipated to be in force in the near future, including NTT DoCoMo's FOMA (Freedom of Mobile Multimedia Access), which is transported over W-CDMA (Wideband Code Division Multiple Access), and CDMA-2000. For the purposes of the invention herein, it will be understood that those skilled in the mobile telecommunications arts will be knowledgeable about any particular format and/or transport protocol requirements that pertain to the particular protocol that is to be used." '353 patent, col. 9, ll. 44-56.</p>

Term/Phrase in Issue and Asserted Claim(s) in Which Term/Phrase Appears	Defendants' Proposed Construction	Defendants' Citations to Intrinsic Evidence
	<p>communication with a network via which Web content may be accessed ('926 patent claims 55, 59, 72, and 75 by reference to claim 52)</p> <p><u>Corresponding structure:</u> Indefinite, as the specification does not disclose any structures corresponding to this limitation.</p>	

EXHIBIT 1

U.S. Patent No. 7,461,353



US007461353B2

(12) **United States Patent**
Rohrbaugh et al.

(10) **Patent No.:** **US 7,461,353 B2**
(45) **Date of Patent:** ***Dec. 2, 2008**

(54) **SCALABLE DISPLAY OF INTERNET CONTENT ON MOBILE DEVICES**

(75) Inventors: **Gary B. Rohrbaugh**, Bellingham, WA (US); **Scott A. Sherman**, Bellingham, WA (US)

(73) Assignee: **Gary Rohrbaugh**, Bellingham, WA (US)

(*) Notice: Subject to any disclaimer, the term of this patent is extended or adjusted under 35 U.S.C. 154(b) by 376 days.

This patent is subject to a terminal disclaimer.

(21) Appl. No.: **11/045,757**

(22) Filed: **Jan. 28, 2005**

(65) **Prior Publication Data**

US 2005/0131887 A1 Jun. 16, 2005

Related U.S. Application Data

(60) Division of application No. 09/878,097, filed on Jun. 8, 2001, now Pat. No. 7,210,099, which is a continuation-in-part of application No. 09/828,511, filed on Apr. 7, 2001, now abandoned.

(60) Provisional application No. 60/217,345, filed on Jul. 11, 2000, provisional application No. 60/211,019, filed on Jun. 12, 2000.

(51) **Int. Cl.**
G06F 17/00 (2006.01)

(52) **U.S. Cl.** **715/815; 715/760; 715/234; 715/239; 715/249**

(58) **Field of Classification Search** **715/517, 715/700-864**

See application file for complete search history.

(56) **References Cited**

U.S. PATENT DOCUMENTS

5,966,135 A 10/1999 Roy et al.

(Continued)

FOREIGN PATENT DOCUMENTS

WO PCT/US01/40920 12/2001

(Continued)

OTHER PUBLICATIONS

<http://www.w3.org/TR/1999/WD-SVG-19990211/>—Pub Feb. 11, 1999 by W3C pp. 1-7 (A complete details can be downloaded at the provides URL above).*

(Continued)

Primary Examiner—Doug Hutton

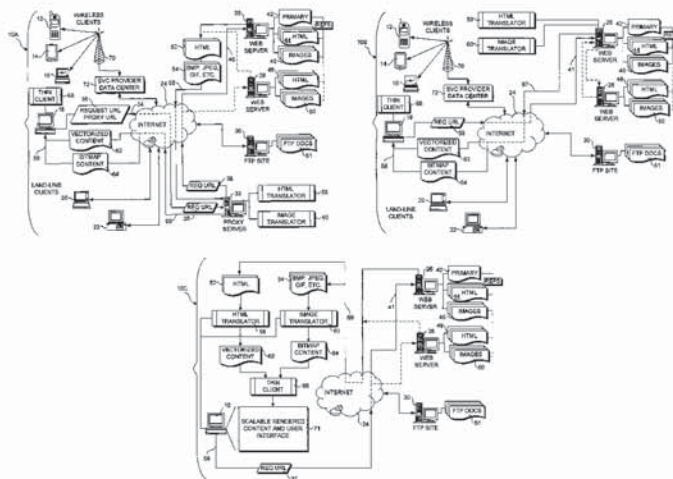
Assistant Examiner—Quoc A Tran

(74) *Attorney, Agent, or Firm*—Law Office of R. Alan Burnett; R. Alan Burnett

(57) **ABSTRACT**

Mobile devices enabled to support resolution-independent scalable display of Internet (Web) content to allow Web pages to be scaled (zoomed) and panned for better viewing on smaller screen sizes. The mobile devices employ software-based processing of original Web content, including HTML-based content, XML, cascade style sheets, etc. to generate scalable content. The scalable content and/or data derived therefrom are then employed to enable the Web content to be rapidly rendered, zoomed, and panned. Moreover, the rendered displays provide substantially the same or identical layout as the original Web page, enabling users to easily navigate to selected content and features on familiar Web pages. Display lists may also be employed to provide further enhancements in rendering speed. Additionally, hardware-based programmed logic may be employed to facilitate various operations.

319 Claims, 22 Drawing Sheets



U.S. PATENT DOCUMENTS

6,011,905	A	1/2000	Huttenlocher et al.	
6,057,854	A	5/2000	Davis, Jr. et al.	
6,185,625	B1	2/2001	Tso et al.	
6,300,947	B1	10/2001	Kanevsky	
6,421,733	B1	7/2002	Tso et al.	
6,466,203	B2	10/2002	Van Ee	
6,546,397	B1	4/2003	Rempell	
6,615,212	B1	9/2003	Dutta et al.	
6,642,925	B2*	11/2003	Roy et al.	345/427
6,674,445	B1*	1/2004	Chithambaram et al.	345/619
6,857,102	B1	2/2005	Bickmore et al.	
6,886,034	B2*	4/2005	Blumberg	709/217
7,219,309	B2	5/2007	Kaasila et al.	
2001/0047428	A1	11/2001	Hunter	
2002/0112237	A1	8/2002	Ketts	
2004/0049598	A1	3/2004	Tucker et al.	
2005/0144256	A1	6/2005	Blumberg	

FOREIGN PATENT DOCUMENTS

WO PCT/US01/40920 6/2002

OTHER PUBLICATIONS

Combined Printout of HTML Help Files for Opera 3.60 (HTML pages printed to PDF docs and combined), Available on May 12, 1999.

Screenshots compilations of Web pages using Opera 3.60, including: www.Altavista.com—May 1, 1999 www.cnet.com—May 8, 1999 www.excite.com—May 8, 1999 www.nytimes.com—Apr. 22, 1999 www.softsource.com—Apr. 17, 1999 www.utexas.edu—Apr. 27, 1999 www.uspto.gov—May 5, 1999 www.yahoo.com—May 8, 1999.

Benjamin B. Bederson et al., Pad++: A Zoomable Graphical Sketchpad For Exploring Alternate Interface Physics, Sep. 19, 1995, <http://www.cs.unm.edu/pad++>.

Benjamin B. Bederson et al., A Zooming Web Browser, SPIE 1996, <http://www.cs.umd.edu/hcil/jazz/learn/papers/spie-96-webbrowser.pdf>.

Specification for Simple Vector Format (SVF) v1.1 Jan. 16, 1995.

Specification for Simple Vector Format (SVF) v2.0 Dec. 6, 2000, <http://www.svf.org/spec.html>.

Changes to SVF, (Date unknown).

Scalable Vector Graphics (SVG) Specification, W3C Working Draft Feb. 11, 1999 WD-SVG-19990211, <http://www.w3.org/TR/1999/WD-SVG-19990211/> (HTML format—initial page).

Scalable Vector Graphics (SVG) Specification, W3C Working Draft Apr. 12, 1999 WD-SVG-19990412, <http://www.w3.org/TR/1999/WD-SVG-19990412/> (HTML format—initial page).

Scalable Vector Graphics (SVG) 1.0 Specification, W3C Candidate Recommendation Nov. 2, 2000.

Introduction to SVG, part of WD-SVG-19990211, <http://www.w3.org/TR/1999/WD-SVG-19990211/intro.html#Document> . . .

Steve Mulder, Sneak Peak at SVG, Mar. 4, 1999, <http://www.webmonkey.com/99/10/index3a.html?tw=eg1999102>.

Janus Boye, SVG Brings Fast Vector Graphics to Web, Jul. 29, 1999 <http://www.irt.org/articles/js176/>.

Vector Markup Language (VML), World Wide Web Consortium Note May 13, 1998, NOTE-VML-19980513, http://www.w3.org/TR/1998/NOTE-VML_19980513.

Precision Graphics Markup Language (PGML), World Wide Web Consortium NOTE Apr. 10, 1998, NOTE-PGML-19980410, <http://www.w3.org/TR/1998/NOTE-PGML-19980410>.

Rick Graham, Mobile SVG at BitFlash Inc., May 2001, <http://www.w3.org/Talks/2002/1007-DI-Helsinki/bitflash/index.html>.

U.S. Appl. No. 60/296,327 Available on PAIR.

U.S. Appl. No. 60/288,287, filed May 2, 2001. Available on PAIR—electronic copy previously filed via IDS submission of Sep. 18, 2007.

U.S. Appl. No. 60/296,275, filed Jun. 5, 2001. Available on PAIR—electronic copy previously filed via IDS submission of Sep. 18, 2007.

U.S. Appl. No. 60/296,237, filed Jun. 5, 2001. Available on PAIR—electronic copy previously filed via IDS submission of Sep. 18, 2007.

U.S. Appl. No. 60/296,274, filed Jun. 5, 2001. Available on PAIR—electronic copy previously filed via IDS submission of Sep. 18, 2007.

U.S. Appl. No. 60/296,284, filed Jun. 5, 2001. Available on PAIR—electronic copy previously filed via IDS submission of Sep. 18, 2007.

U.S. Appl. No. 60/296,231, filed Jun. 5, 2001. Available on PAIR—electronic copy previously filed via IDS submission of Sep. 18, 2007.

U.S. Appl. No. 60/296,224, filed Jun. 5, 2001. Available on PAIR—electronic copy previously filed via IDS submission of Sep. 18, 2007.

U.S. Appl. No. 60/296,426, filed Jun. 5, 2001. Available on PAIR—electronic copy previously filed via IDS submission of Sep. 18, 2007.

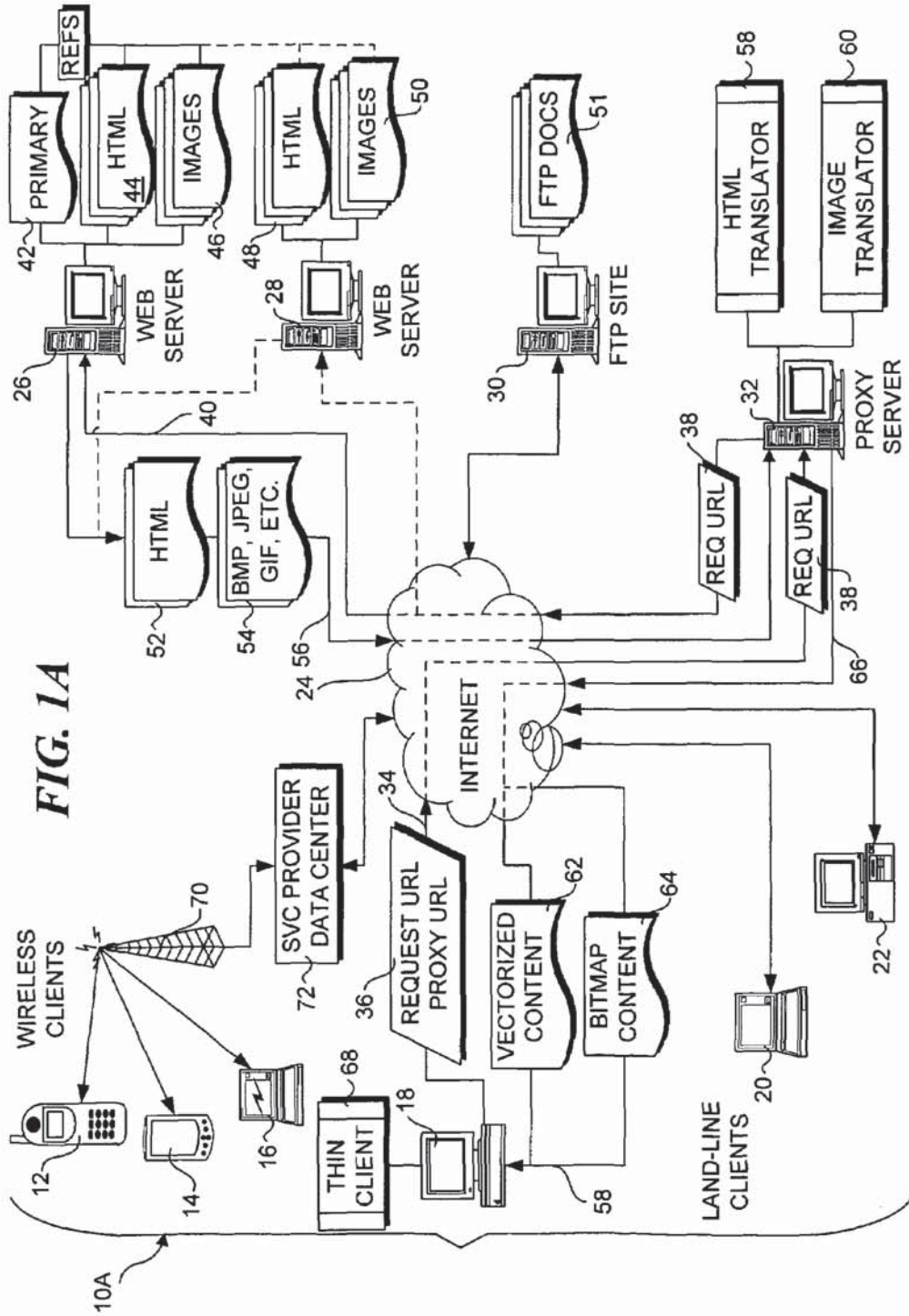
U.S. Appl. No. 60/296,273, filed Jun. 5, 2001. Available on PAIR—electronic copy previously filed via IDS submission of Sep. 18, 2007.

U.S. Appl. No. 60/296,283, filed Jun. 5, 2001. Available on PAIR—electronic copy previously filed via IDS submission of Sep. 18, 2007.

U.S. Appl. No. 60/296,281, filed Jun. 5, 2001. Available on PAIR—electronic copy previously filed via IDS submission of Sep. 18, 2007.

U.S. Appl. No. 60/296,327, filed Jun. 5, 2001. Available on PAIR—electronic copy previously filed via IDS submission of Sep. 18, 2007.

* cited by examiner



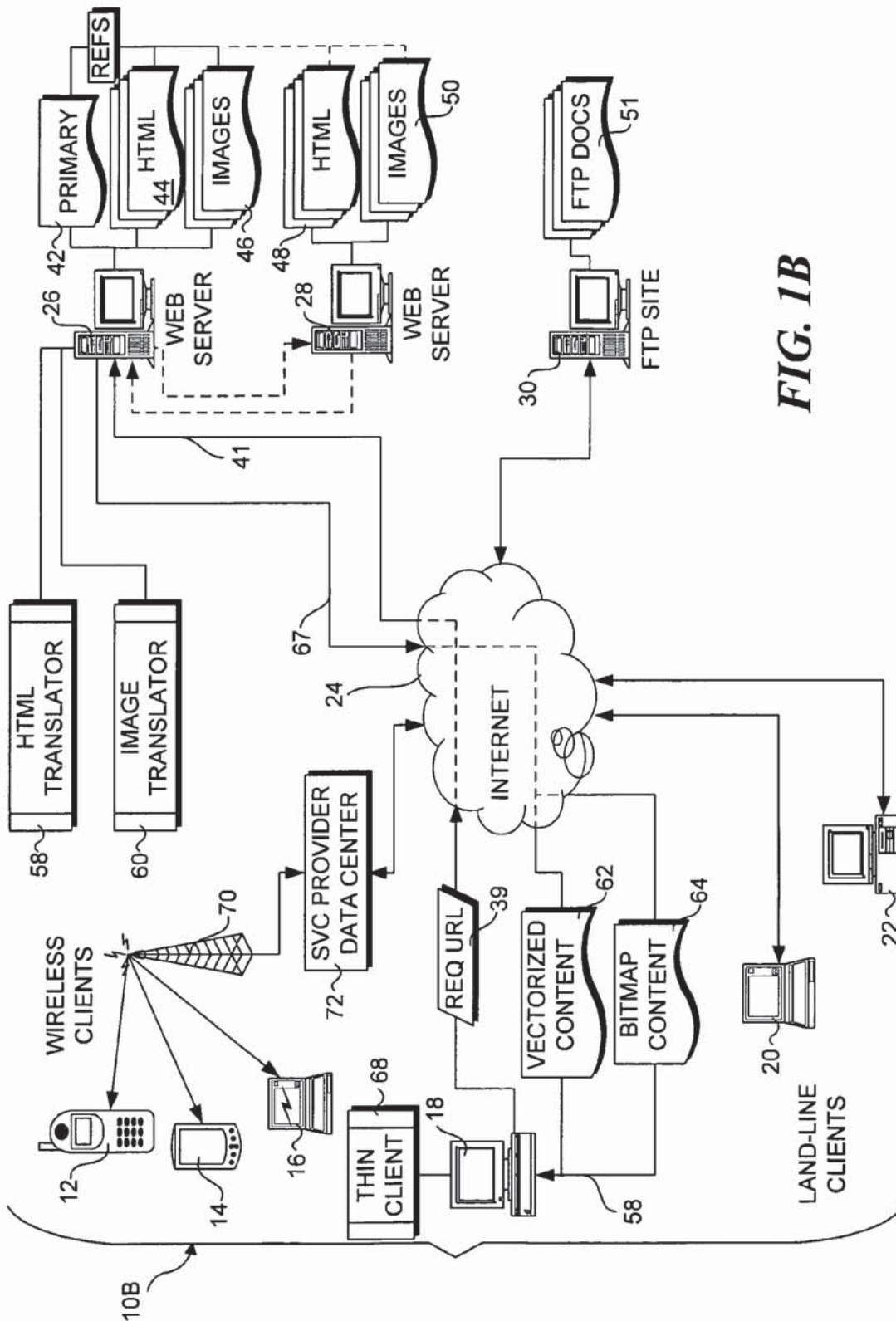


FIG. 1B

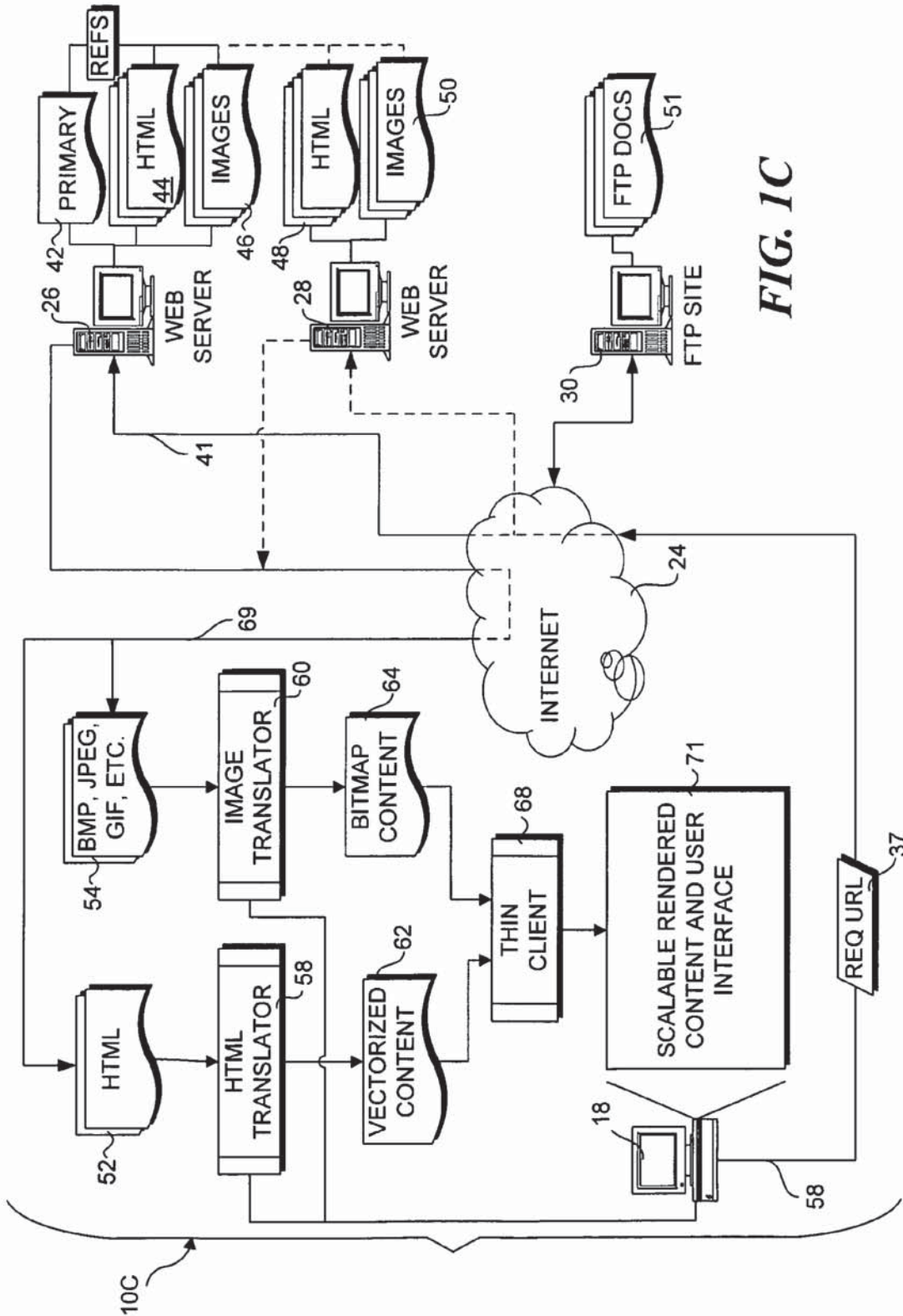


FIG. 1C

FIG. 2A

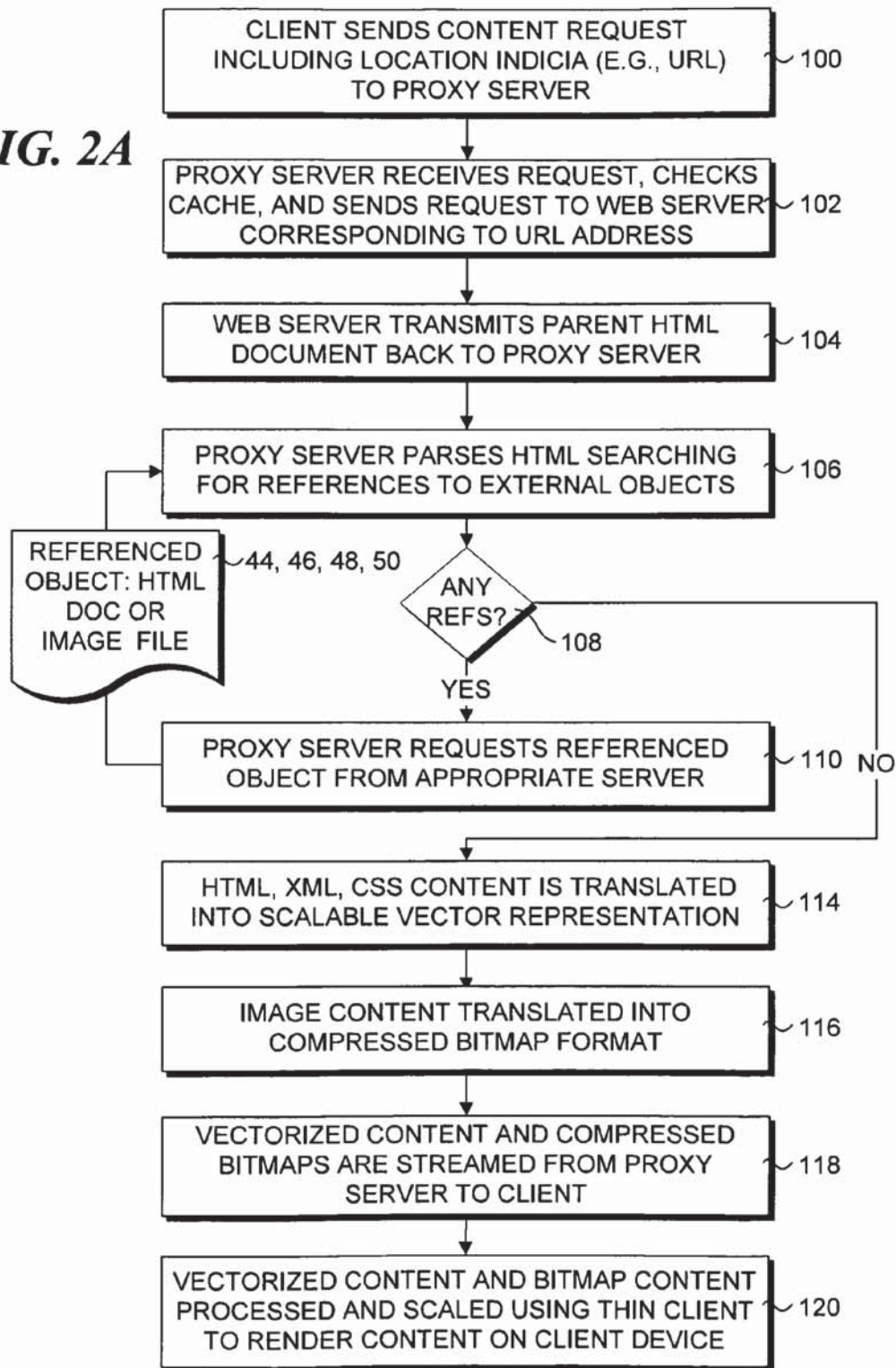


FIG. 2B

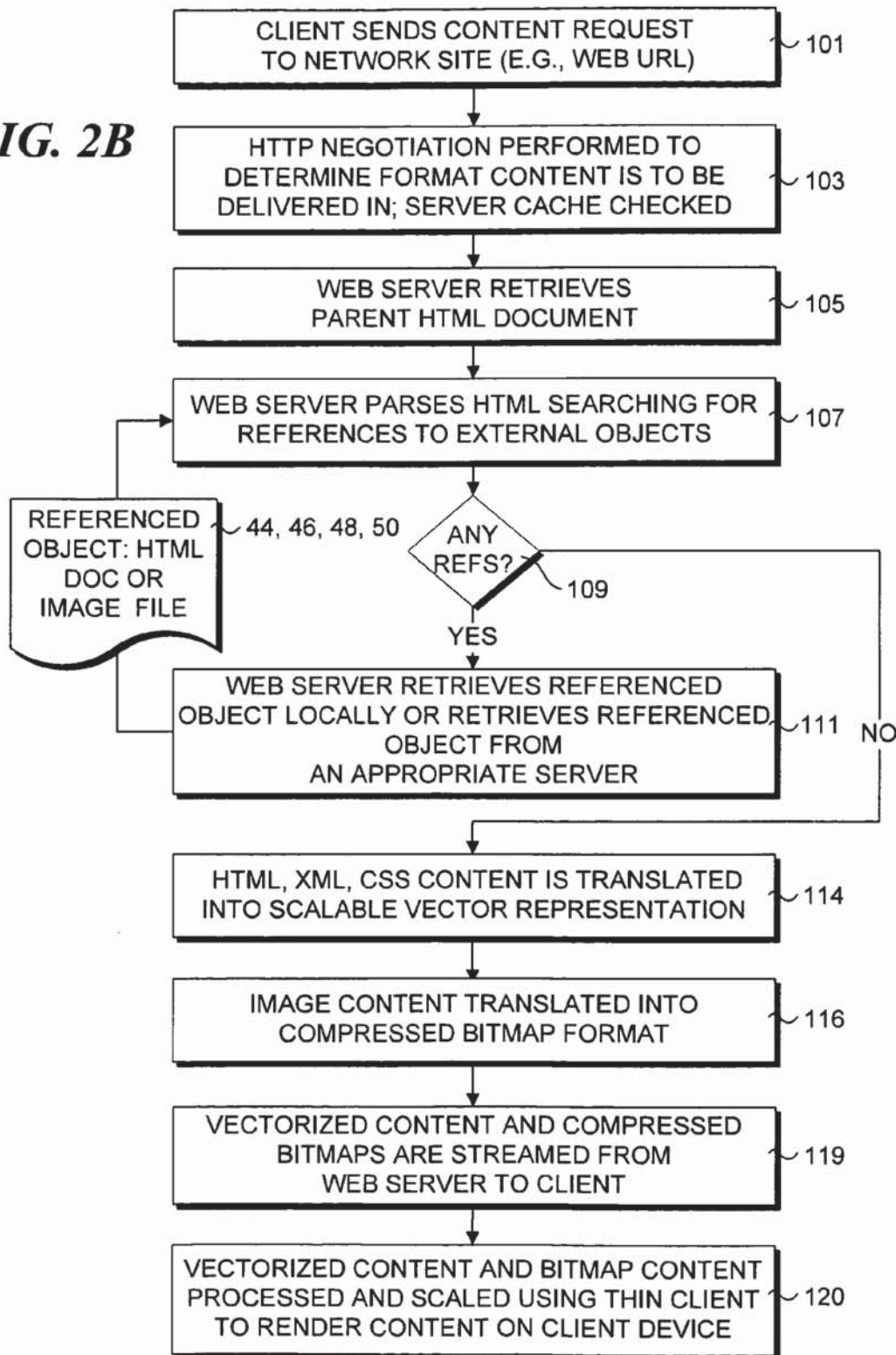
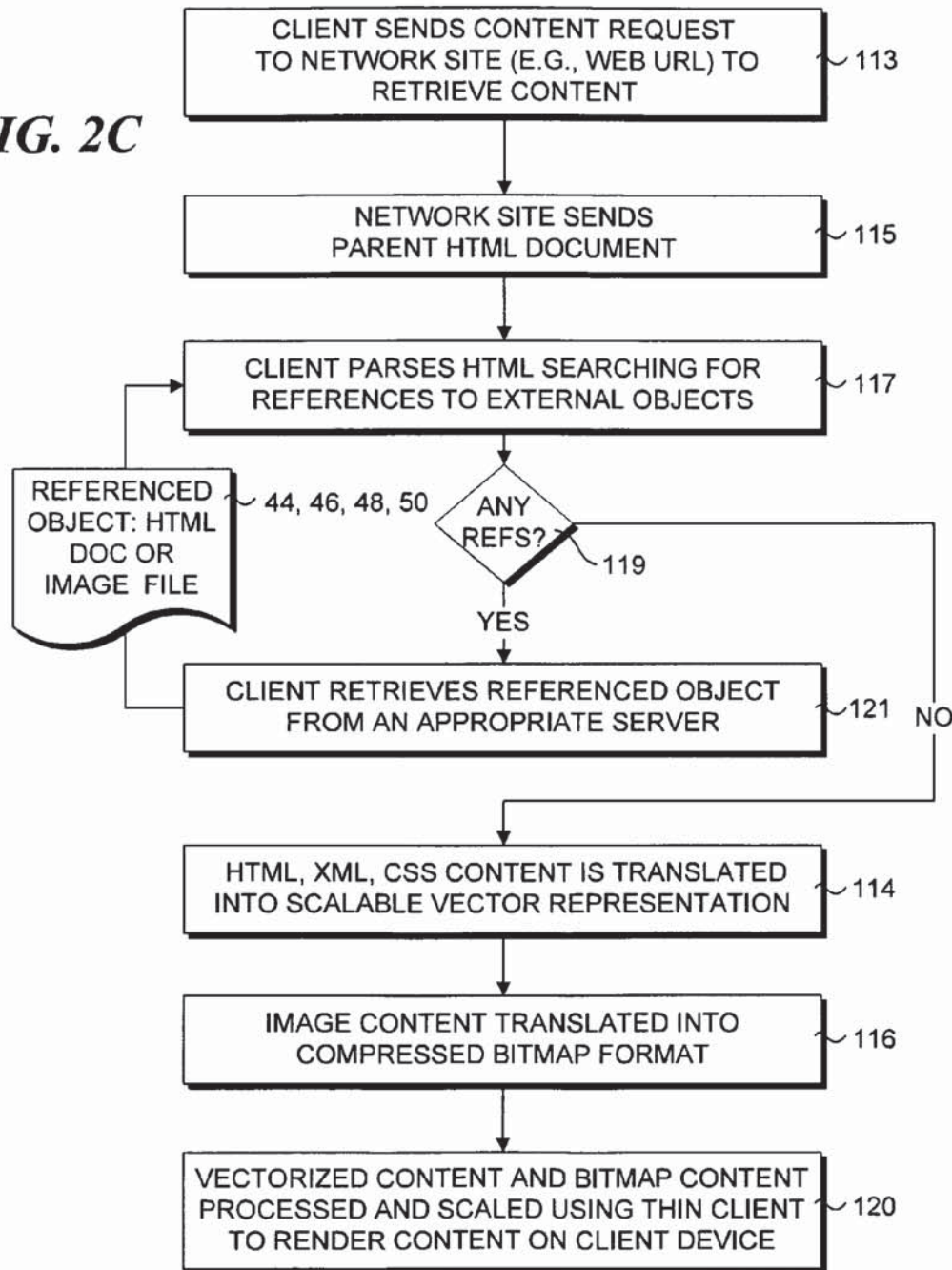


FIG. 2C



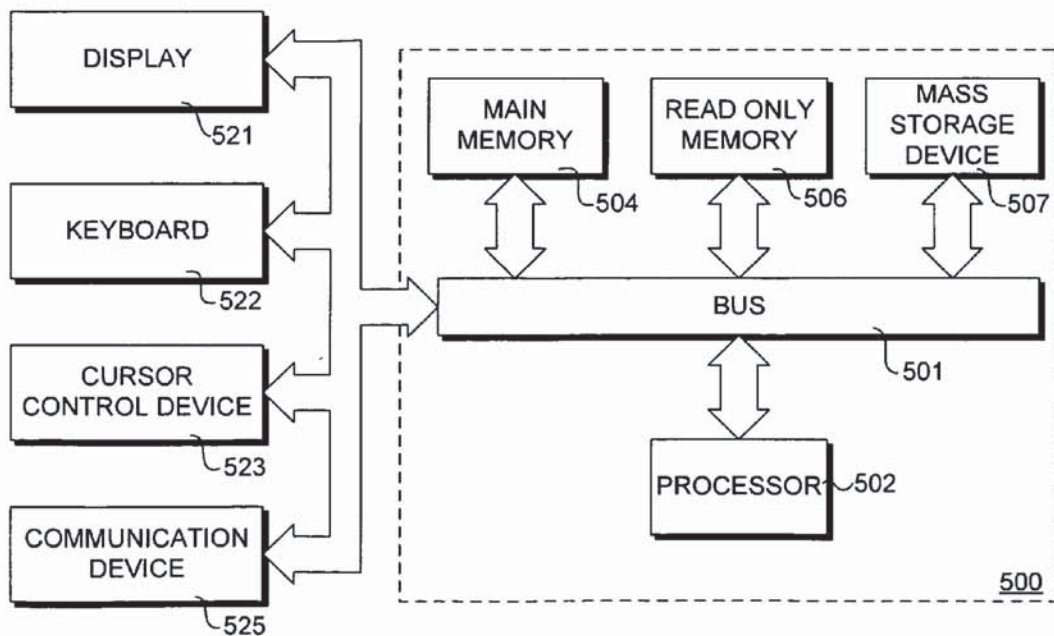
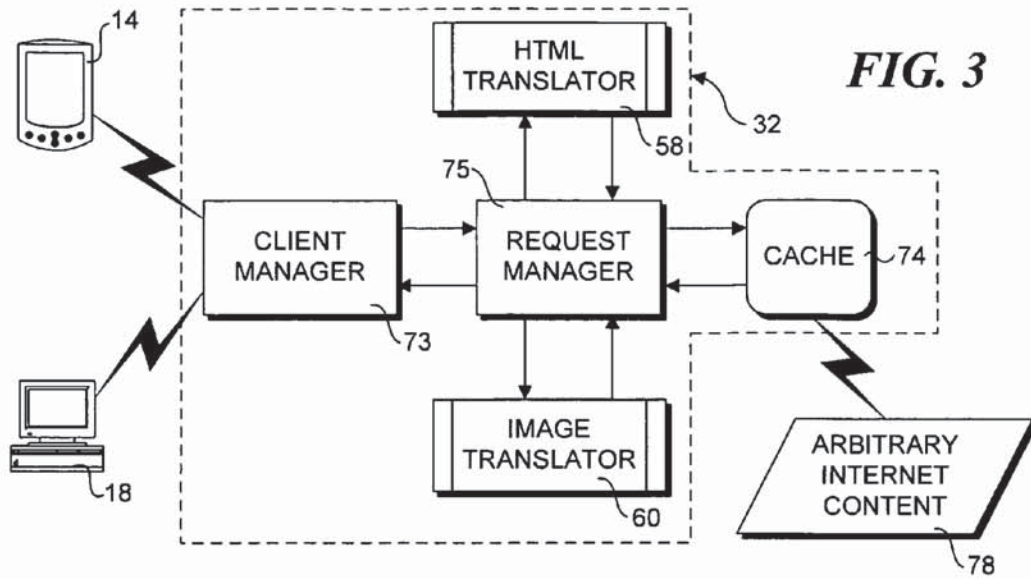


FIG. 10

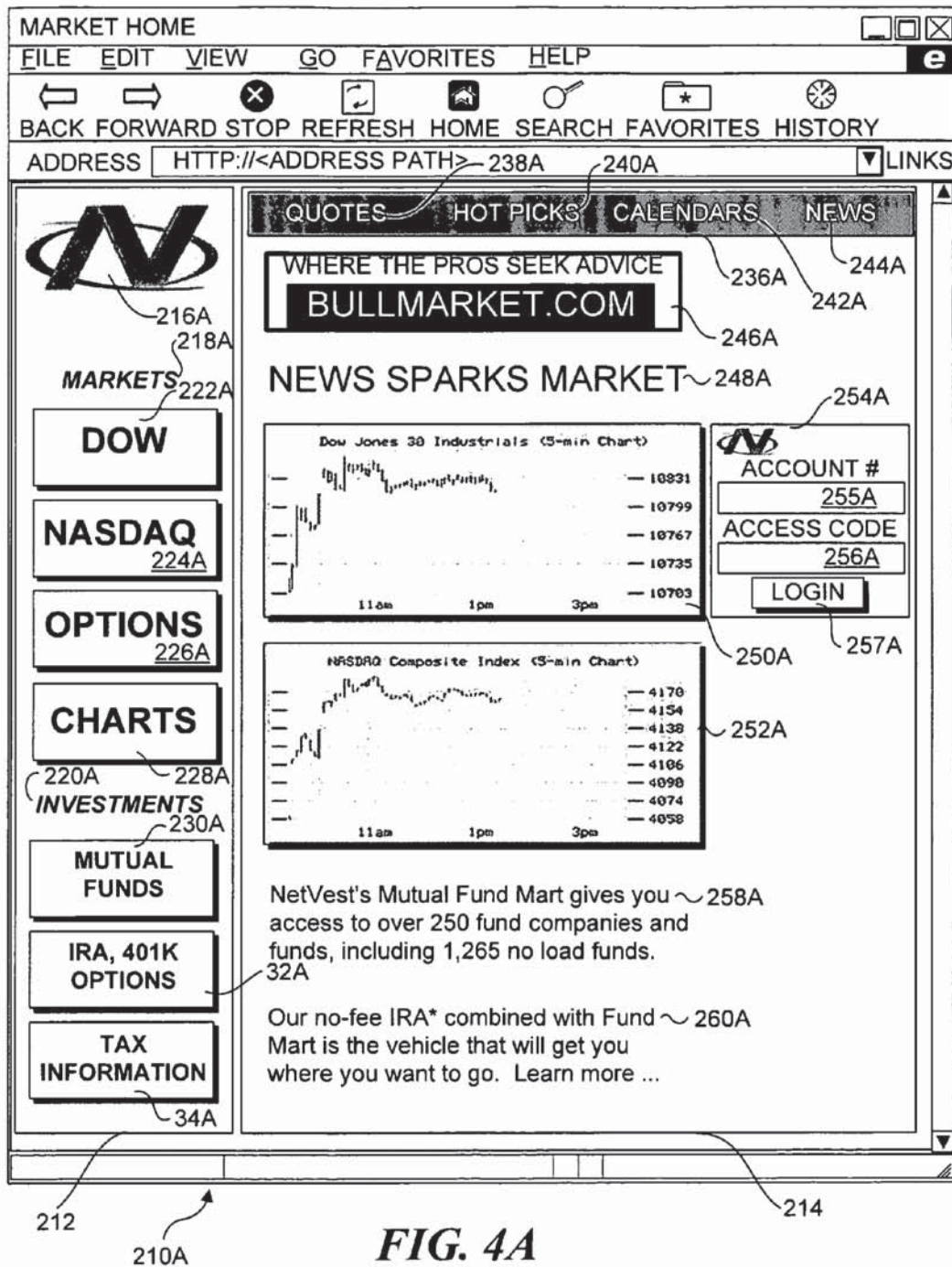


FIG. 4A

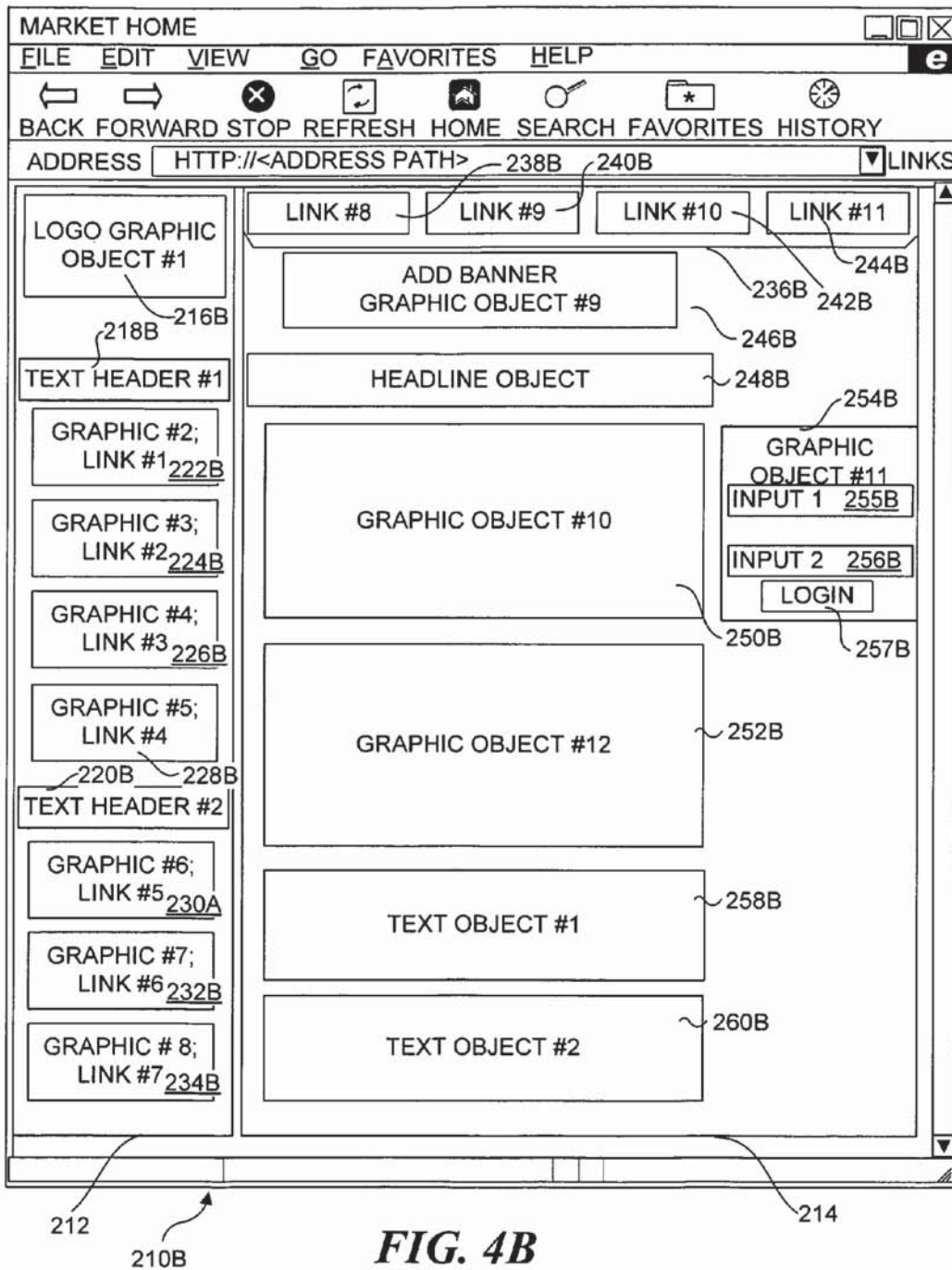
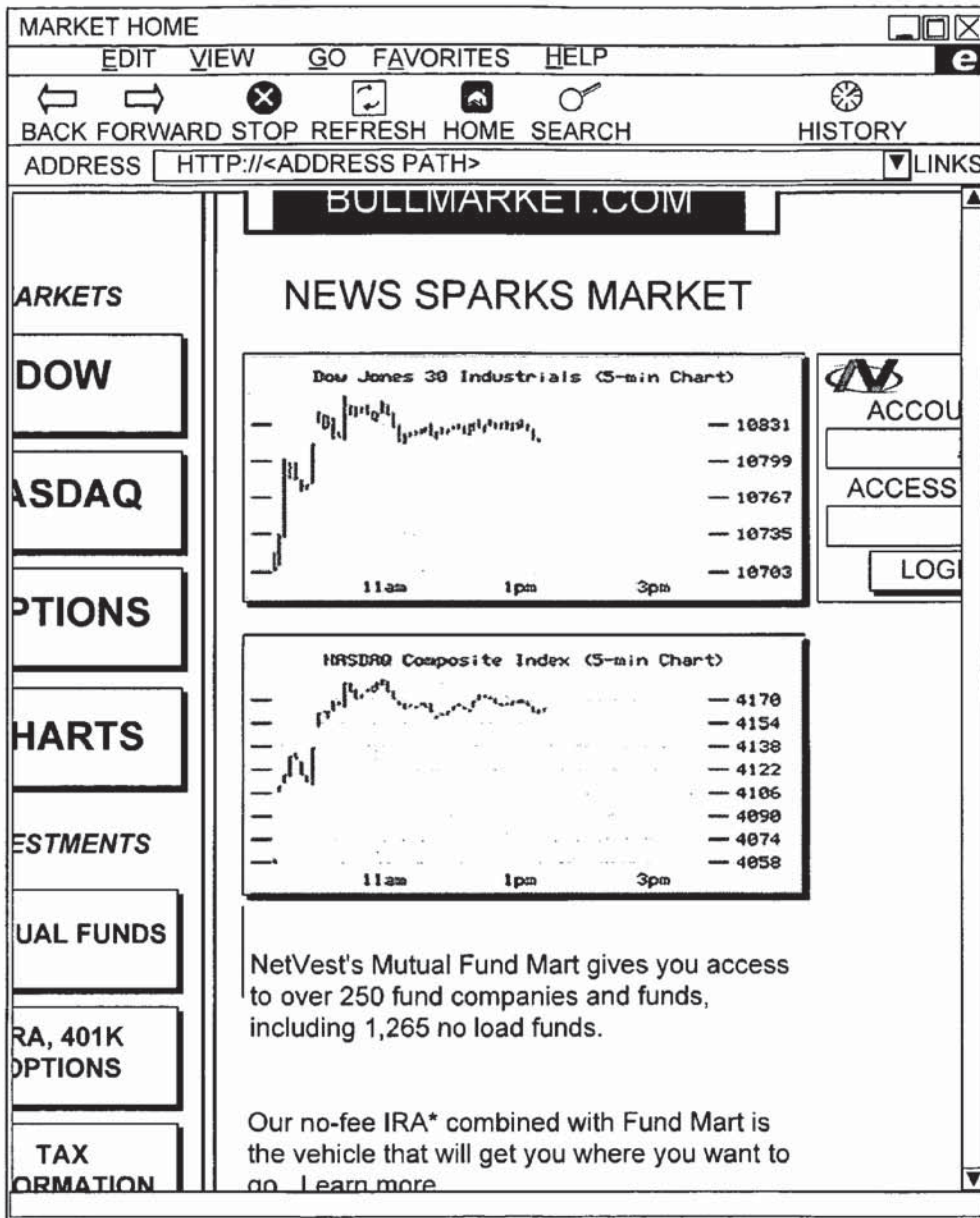


FIG. 4B



210E

FIG. 4E

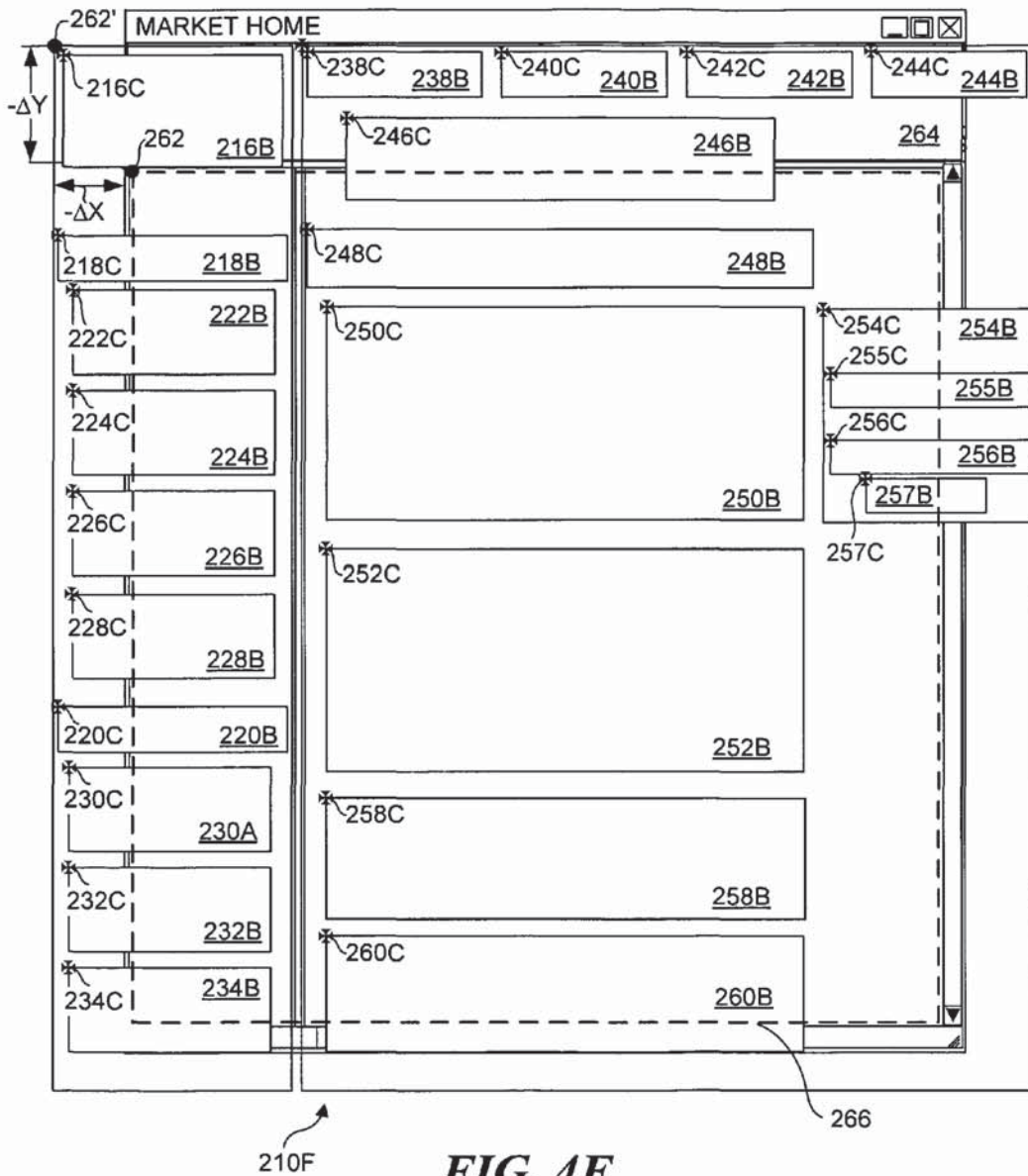


FIG. 4F

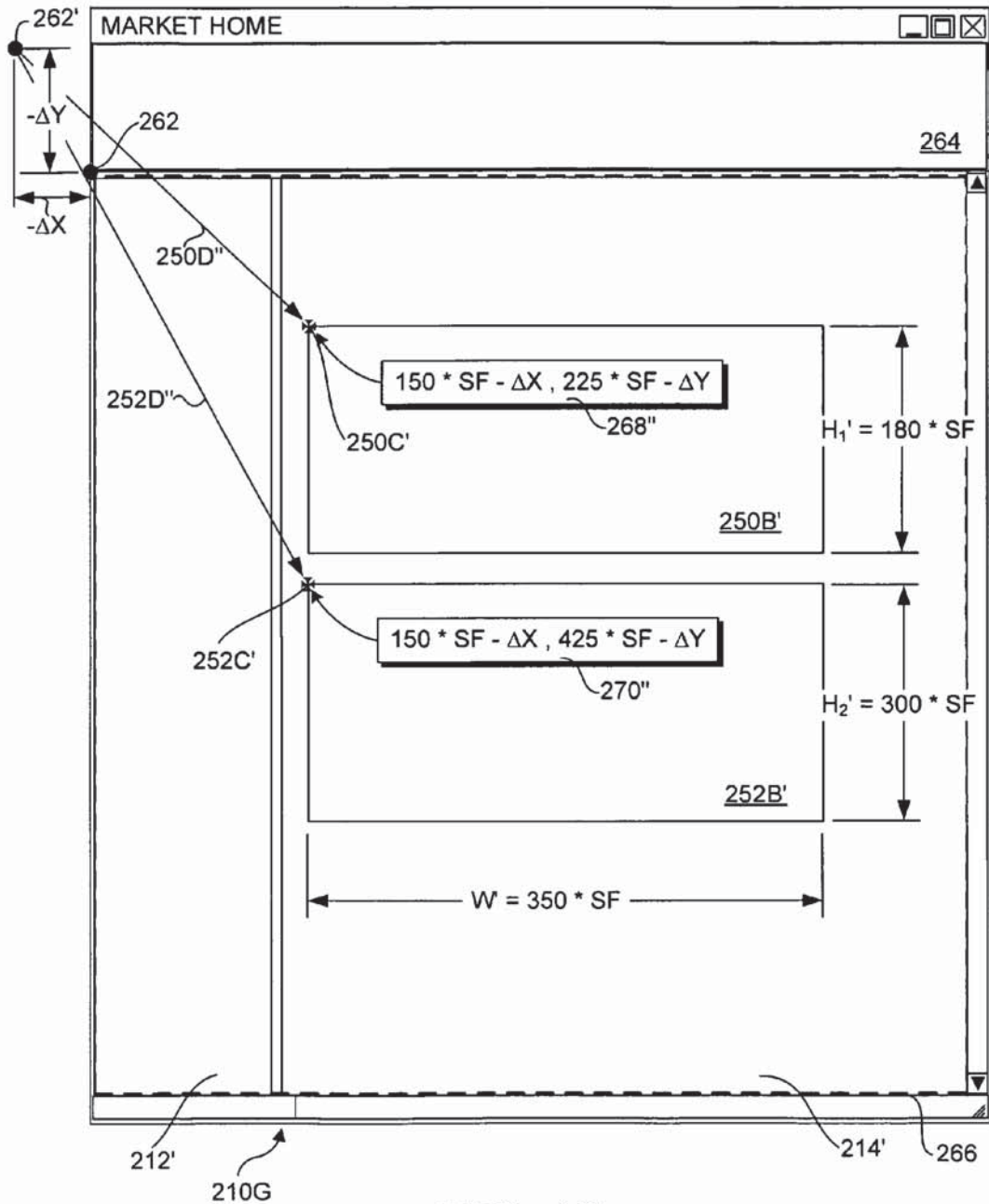
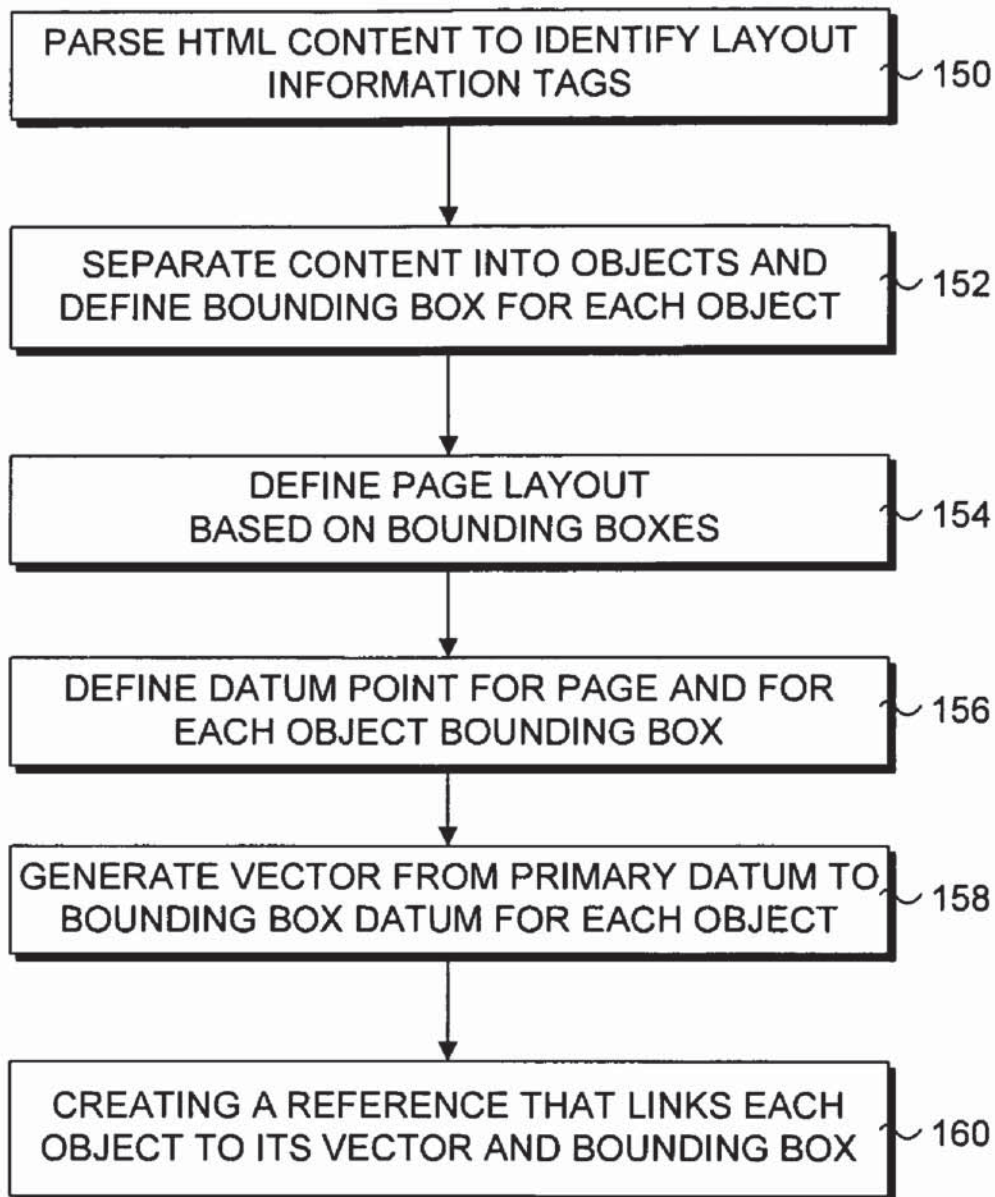
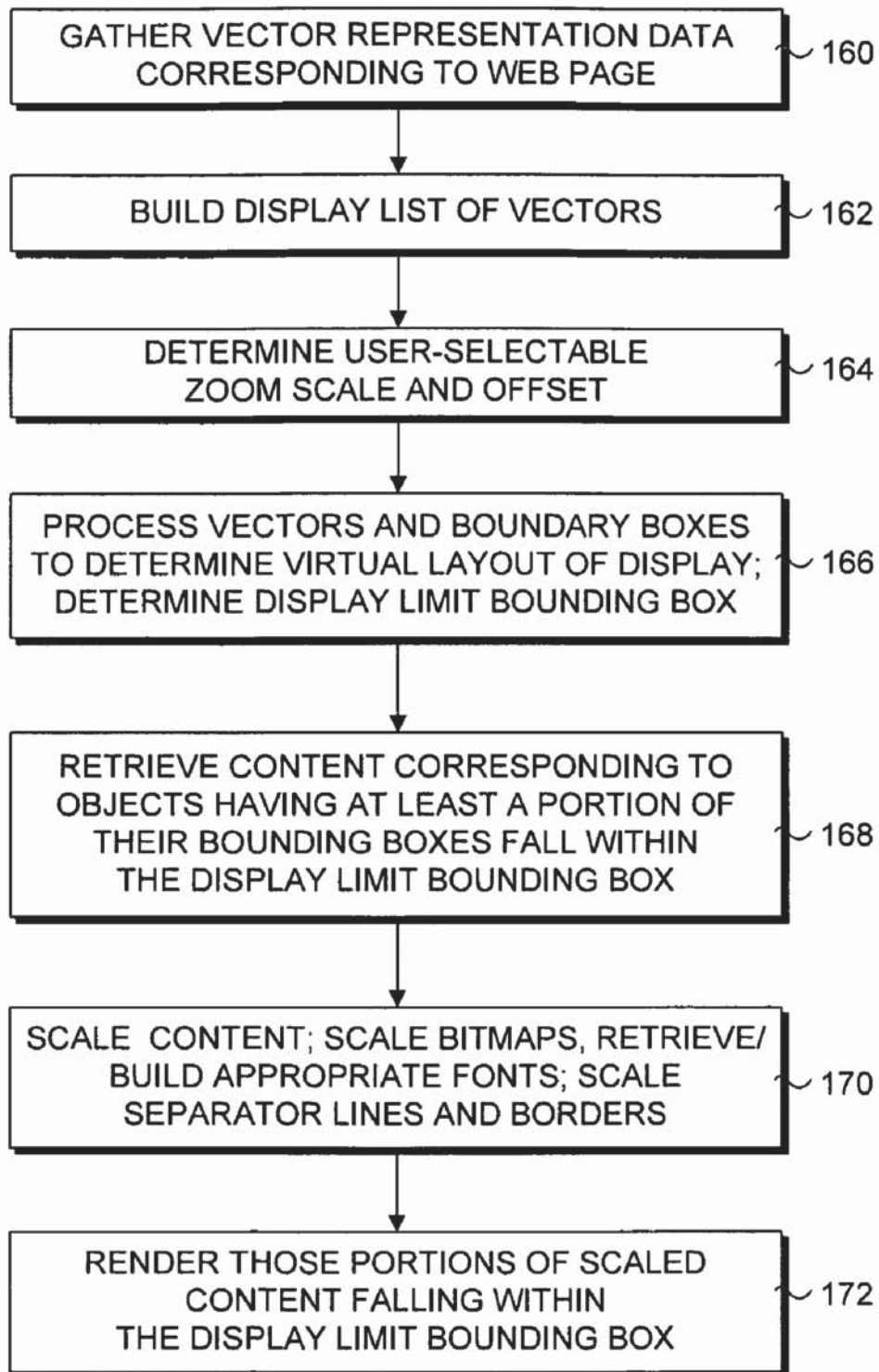


FIG. 4G

**FIG. 5**

**FIG. 6**

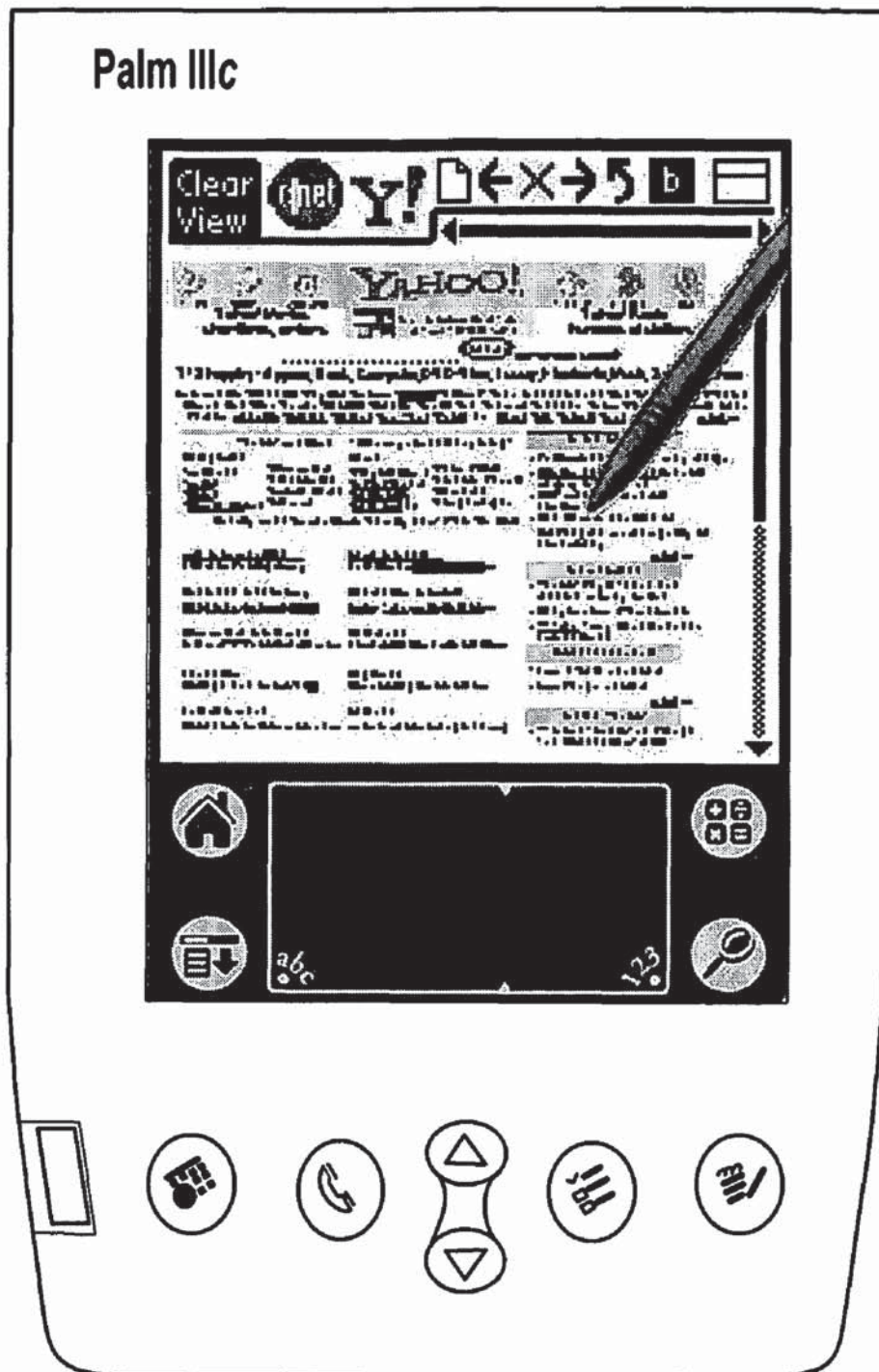


FIG. 7A

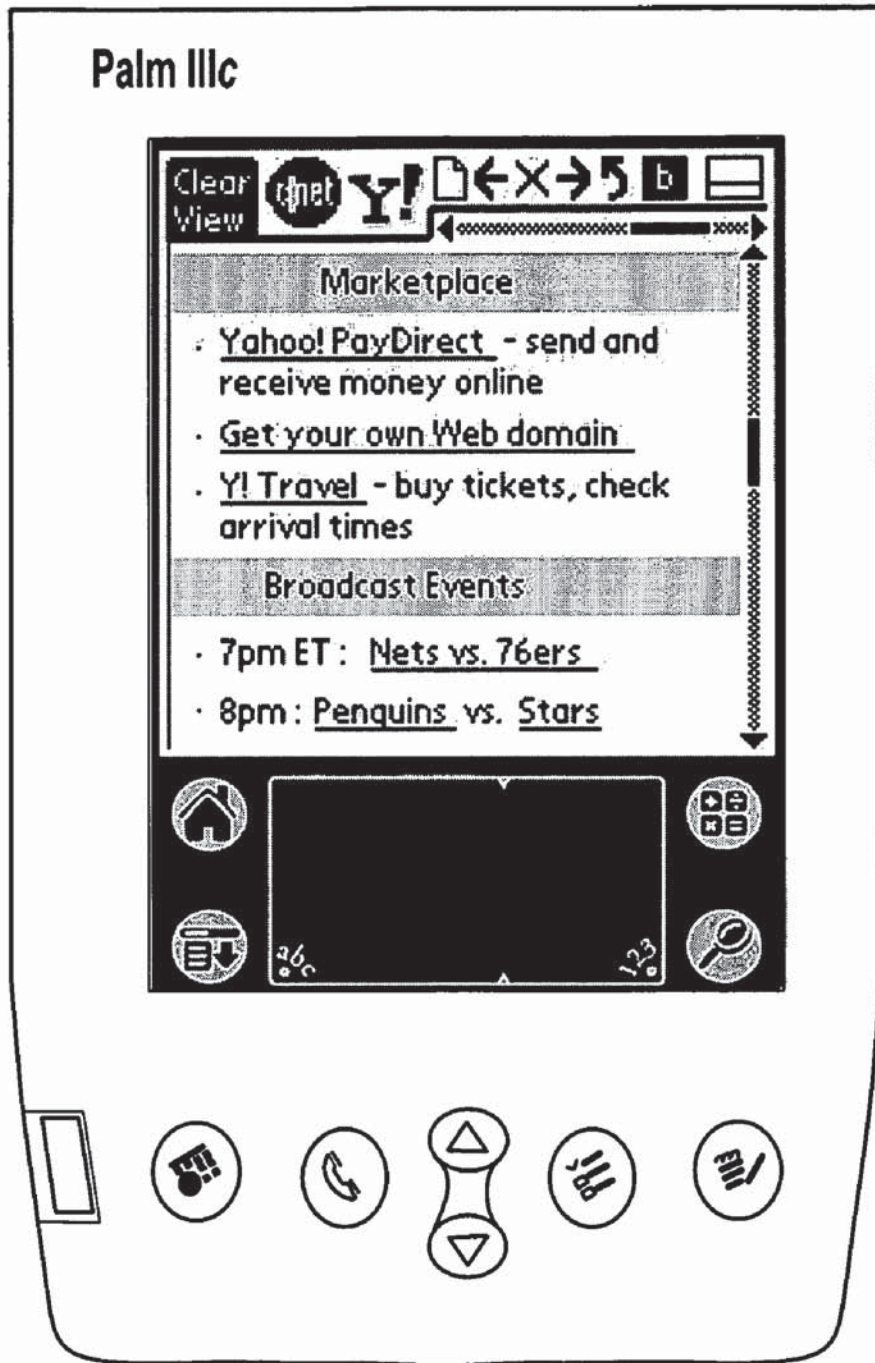


FIG. 7B

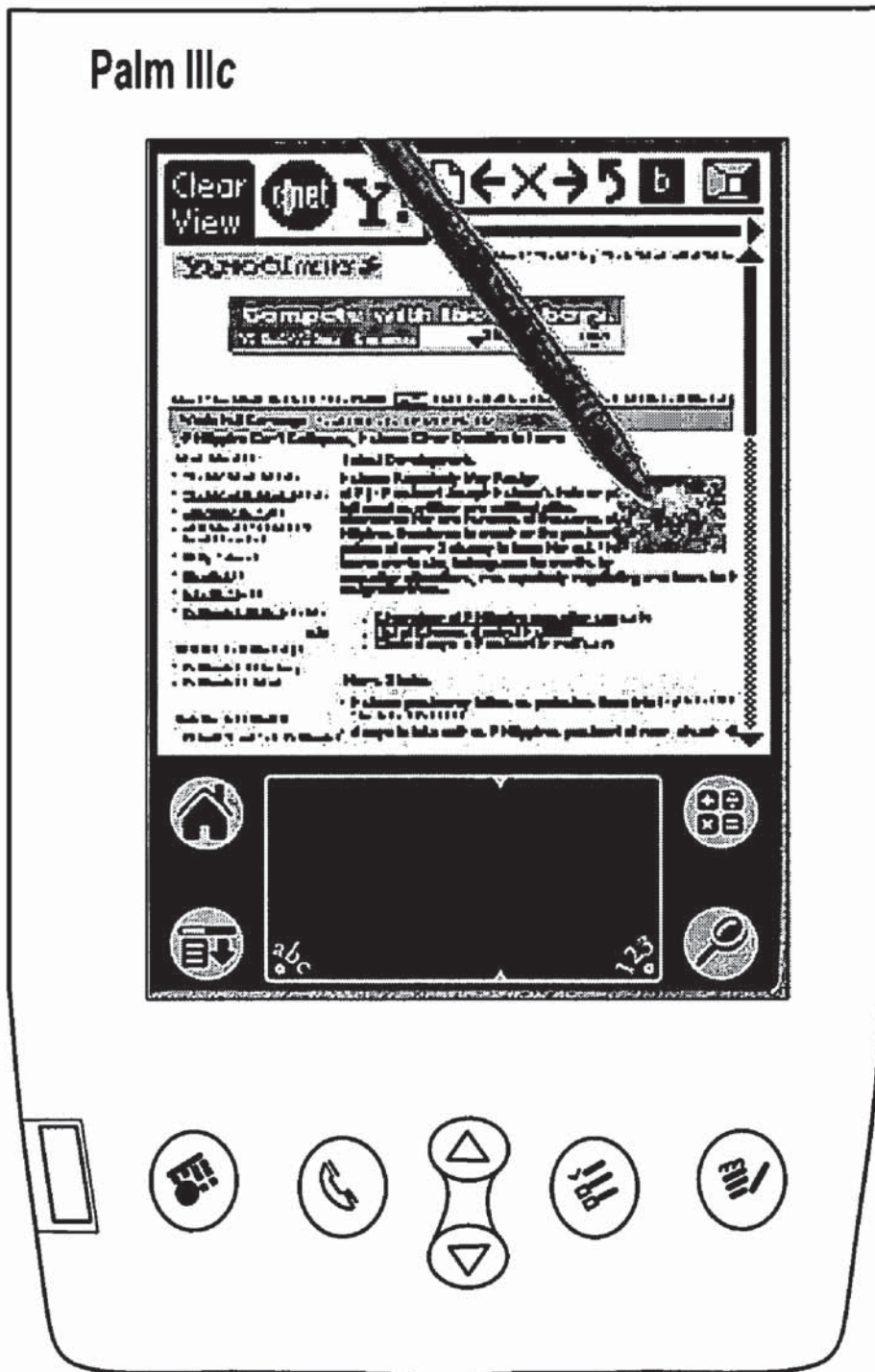


FIG. 8A



FIG. 8B

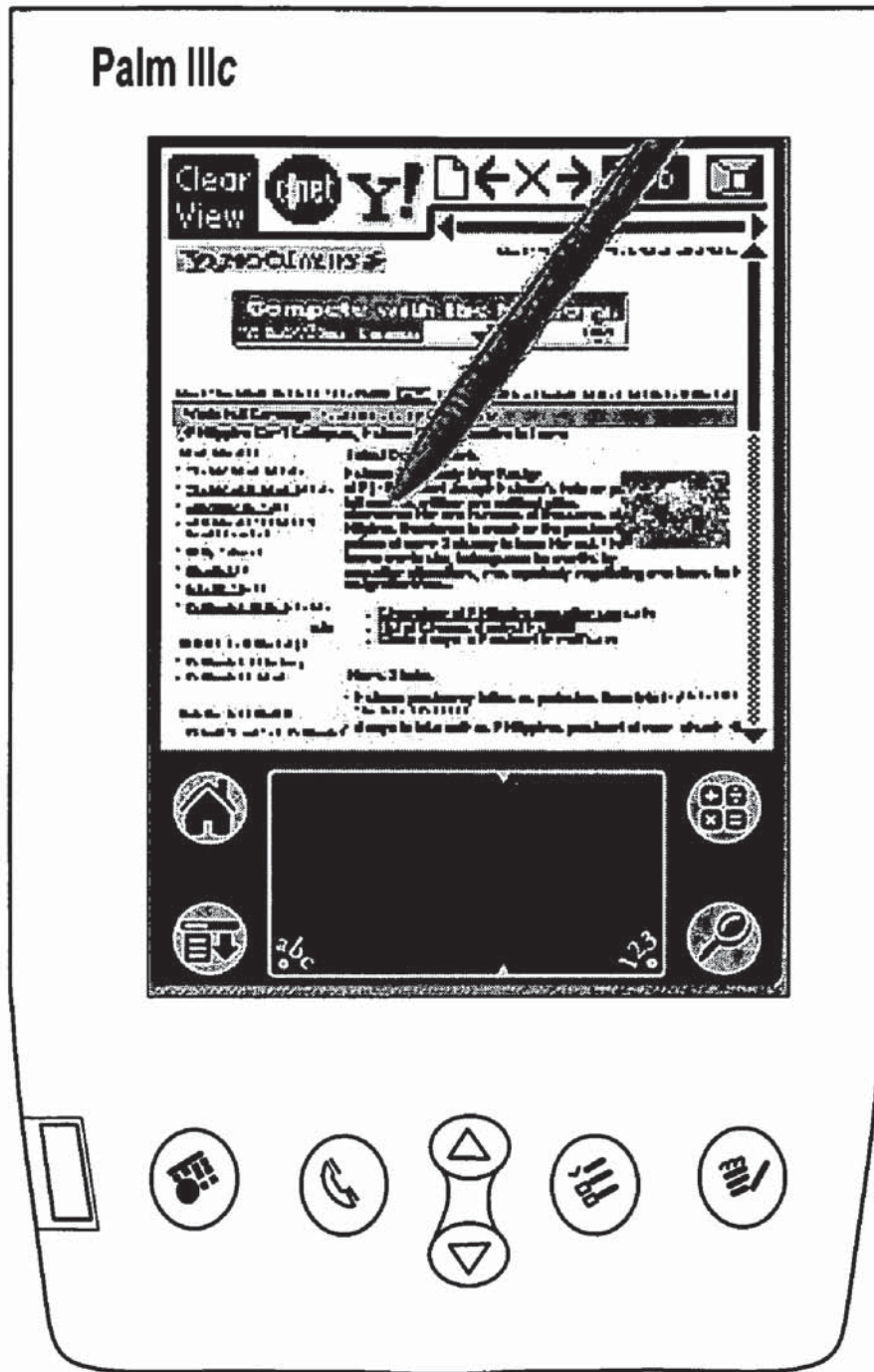


FIG. 9A

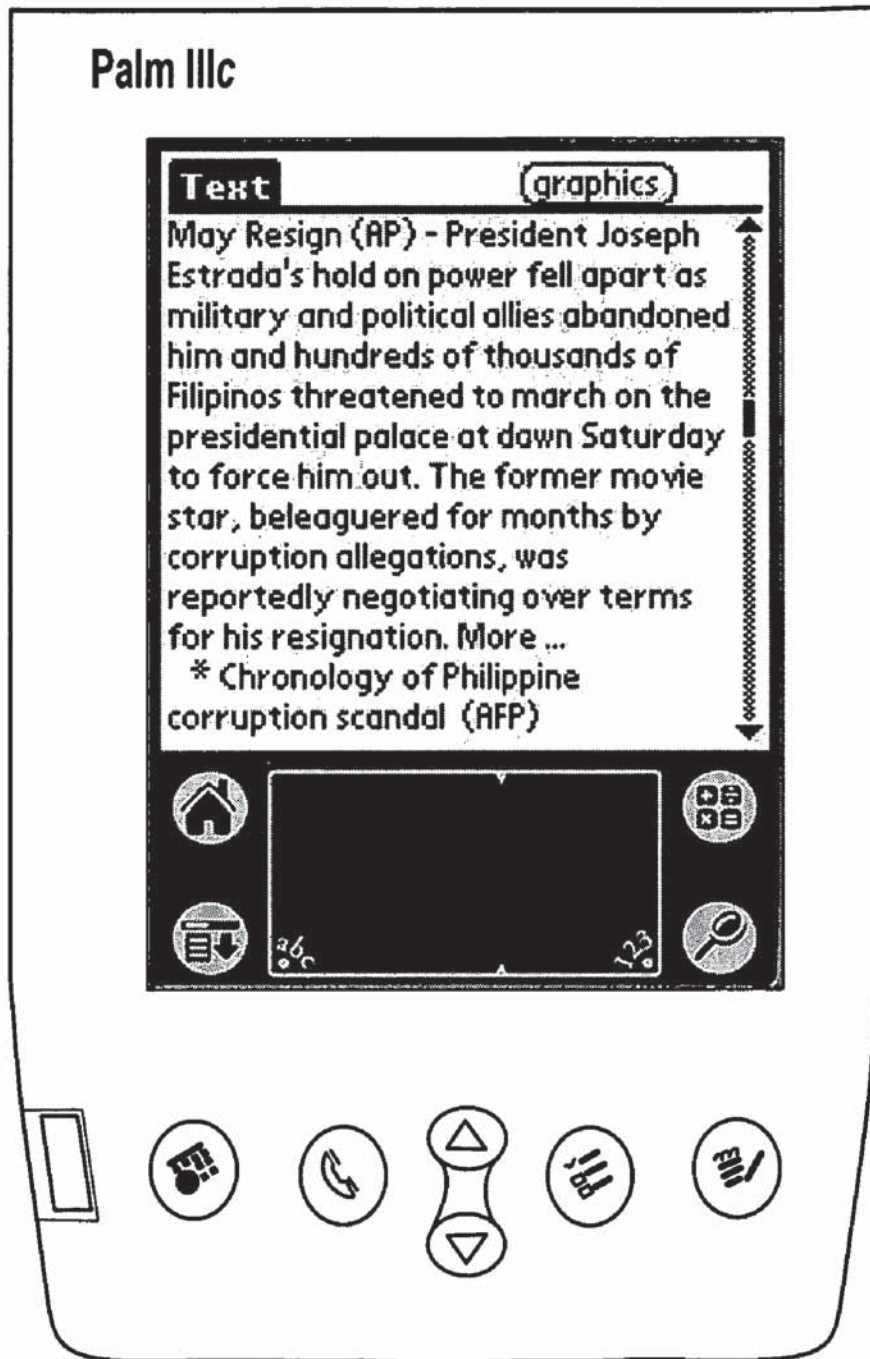


FIG. 9B

1

SCALABLE DISPLAY OF INTERNET CONTENT ON MOBILE DEVICES

RELATED APPLICATIONS

The present application is a Divisional Application of U.S. application Ser. No. 09/878,097, filed Jun. 8, 2001, now U.S. Pat. No. 7,210,099, which is a Continuation-in-Part of U.S. Non-Provisional application Ser. No. 09/828,511, filed Apr. 7, 2001, now abandoned, entitled "RESOLUTION INDEPENDENT VECTOR DISPLAY OF INTERNET CONTENT," the benefit of the filing date of which is claimed under 35 U.S.C. § 120. This application further claims the benefit of the filing dates of U.S. Provisional Application No. 60/211,019, filed Jun. 12, 2000, entitled "METHOD AND SYSTEM FOR RESOLUTION INDEPENDENT DISPLAY OF HTML AND XML CONTENT" and U.S. Provisional Application No. 60/217,345, filed Jul. 11, 2000, entitled "METHOD AND SYSTEM FOR SELECTION, RETRIEVAL, AND CONVERSION OF COMPUTER CONTENT TO VECTOR FORMAT FOR RESOLUTION INDEPENDENT DISPLAY," under 35 U.S.C. § 119(e).

The present application is also related to U.S. application Ser. No. 11/735,477 filed on Apr. 15, 2007, U.S. application Ser. No. 11/735,482 filed on Apr. 15, 2007, U.S. application Ser. No. 11/738,486 filed on Apr. 21, 2007, and U.S. application Ser. No. 11/738,932 filed on Apr. 23, 2007, each of which are continuations of U.S. application Ser. No. 09/878,097, now U.S. Pat. No. 7,210,099.

COPYRIGHT NOTICE

Contained herein is material that is subject to copyright protection. The copyright owner has no objection to the facsimile reproduction of the patent disclosure by any person as it appears in the Patent and Trademark Office patent files or records, but otherwise reserves all rights to the copyright whatsoever.

BACKGROUND OF THE INVENTION

1. Field of the Invention

The invention relates generally to translation of Internet and World Wide Web content to scalable vector representation. More particularly, the invention relates to apparatus and methods for zoom enabling the display of content in an Internet information browser by retrieving and translating Hyper-Text Markup Language (HTML), extensible Markup Language (XML), and other Internet content to vector representations of that content.

2. Description of the Related Art

Text only Internet information browsers began as a project at the CERN, European Organization for Nuclear Research, facility in Geneva Switzerland. From its inception the intent was to provide a mesh or web of access to data with a common user interface. Browsers moved from the academic environment when NCSA, the National Center for Supercomputing Applications at the University of Illinois in Urbana-Champaign developed Mosaic, an Internet information browser and World Wide Web client.

Internet content is stored in multiple file formats. These formats include HTML (Hyper Text Markup Language) and XML (extended Markup Language) as well as graphic file format GIF (Graphics Interchange Format) and JPEG (Joint Photographic Experts Group). These four file formats constitute the majority of Internet content. Font size and resizing display area for content can alter the size of the display of

2

Internet content in existing browsers. The majority of Internet content displays as a flat single resolution with no browser support for zoom.

Much of the Internet content has been designed for display on desktop computers with a single target resolution. Even though HTML has the ability to adapt to changes in screen resolution, major Internet content providers have chosen to create their Web pages using fixed resolution structures, such as tables. This gives them the ability to control the look and feel of their Web sites. This fixed resolution approach has evolved to the point that the fixed resolution layout of Web pages has become the most common method to brand or uniquely identify Web sites. While this fixed resolution approach is good for site branding and product differentiation it does present a daunting technical problem for display of Internet content (designed for desktop computers) on small screen, low resolution, or different aspect ratio devices, such as cell phones and hand held computers.

BRIEF SUMMARY OF THE INVENTION

In accordance with aspects of the invention, mobile devices enabled to support resolution-independent scalable display of Internet (Web) content to allow Web pages to be scaled (zoomed) and panned for better viewing on smaller screen sizes are disclosed. The mobile devices employ novel processing of original Web content, including HTML-based content, XML, cascade style sheets, etc. to generate scalable content. The scalable content and/or data derived therefrom are then employed to enable the Web content to be rapidly rendered, zoomed, and panned. Moreover, the rendered displays provide substantially the same or identical layout as the original Web page, enabling users to easily navigate to selected content and features on familiar Web pages. Display lists may also be employed to provide further enhancements in rendering speed. Additionally, hardware-based programmed logic may also be employed to facilitate various operations.

According to further aspects, some mobile devices may employ touch-sensitive display screens that enable users to provide various inputs to control display of content within Web pages. Exemplary user inputs include tap-based inputs to selectively zoom in on columns, images, and paragraphs. Users can also define a window to zoom in on via the touch-sensitive display.

According to additional aspects of the invention, methods and software for enabling support for resolution-independent scalable display of Web content are provided. The methods and software enable users of various devices, from handheld devices with small screens, to desktop PC's and laptops, to very large screen devices, to view and interact with Web pages in a manner independent of the screen resolution of such device's built-in or associated display, while maintaining the look and feel of browsing such pages with a conventional desktop browser. Thus, users of various devices having different screen resolutions are enabled to browse Web pages from among literally billions of Web pages while providing a full Web browsing experience.

Other features of the present invention will be apparent from the accompanying drawings and from the detailed description that follows.

BRIEF DESCRIPTION OF THE DRAWINGS

The appended claims set forth the features of the invention with particularity. The invention, together with its advan-

tages, may be best understood from the following detailed description taken in conjunction with the accompanying drawings of which:

FIG. 1A is a block schematic diagram illustrating a first exemplary system infrastructure in accordance with the present invention in which content translation services are performed by a third-party proxy service that translates content requested from a client that is retrieved from one or more network resources into a scalable vector representation and delivers the translated content to the client;

FIG. 1B is a block schematic diagram illustrating a second exemplary system infrastructure in which the translation of content is performed at a content provider's web site and delivered directly to the requesting client;

FIG. 1C is a block schematic diagram illustrating a third exemplary system infrastructure in which content received from one or more network sources is translated into a scalable vector representation at the client;

FIG. 2A is a flowchart illustrating how data is retrieved, processed and transferred in accordance with the system infrastructure of FIG. 1A;

FIG. 2B is a flowchart illustrating how data is retrieved, processed and transferred in accordance with the system infrastructure of FIG. 1B;

FIG. 2C is a flowchart illustrating how data is retrieved, processed and transferred in accordance with the system infrastructure of FIG. 1C;

FIG. 3 is a block schematic diagram illustrating an exemplary architecture corresponding to the proxy server of FIG. 1A;

FIG. 4A is a representation of an exemplary web page has displayed on a conventional browser;

FIG. 4B is a schematic diagram illustrates various objects that are generated based on the HTML code of the web page of FIG. 4A;

FIG. 4C is a schematic diagram illustrating a set of vectors and bounding boxes corresponding to the objects generated in FIG. 4B;

FIG. 4D is a schematic diagram illustrating how various vectors and bounding boxes may be defined in accordance with the invention;

FIG. 4E is a representation of the web page of FIG. 4A after it has been offset and scaled in accordance with the invention;

FIG. 4F is a schematic diagram illustrating new datum points and bounding boxes corresponding to the scaled and offset web page;

FIG. 4G is a schematic diagram illustrating new vectors and bounding box parameters for a pair of objects in the scaled and offset web page;

FIG. 5 is a flowchart illustrating the logic used by the invention when translating content into a scalable vector representation of that content;

FIG. 6 is a flowchart illustrating client-side operations that are performed to create a rendered display page based on the translated content the client receives and user-input;

FIGS. 7A and 7B are representations of a nominal and a zoomed in column view of an exemplary web page as they might appear on a Palm device;

FIGS. 8A and 8B are representation of nominal and zoomed in view of an exemplary graphic image as they might appear on the Palm device;

FIGS. 9A and 9B are representations of a nominal and zoomed in view of a text portion of a web page as they might appear on the Palm device; and

FIG. 10 illustrates an exemplary computer system that may be used for implementing various aspects of embodiments of the invention.

DETAILED DESCRIPTION OF THE INVENTION

Apparatus and methods are described for creating resolution independent vector display of Internet content to allow it to be scaled (zoomed) larger and smaller for better viewing or to fit any resolution or screen size. In addition, infrastructure and methods are provided for delivering such content to clients.

In the following description, for the purposes of explanation, numerous specific details are set forth in order to provide a thorough understanding of the present invention. It will be apparent, however, to one skilled in the art that the present invention may be practiced without some of these specific details. In other instances, well-known structures and devices are shown in block diagram form.

The present invention includes various operations, which will be described below. The operations of the present invention may be performed by hardware components or may be embodied in machine-executable instructions, which may be used to cause a general-purpose or special-purpose processor or logic circuits programmed with the instructions to perform the operations. Alternatively, the operations may be performed by a combination of hardware and software.

The present invention may be provided as a computer program product that may include one or more machine-readable mediums having stored thereon instructions, which may be used to program a computer (or other electronic devices) to perform a process according to the present invention. The machine-readable medium may include, but is not limited to, floppy diskettes, optical disks, CD-ROMs, and magneto-optical disks, ROMs, RAMs, EPROMs, EEPROMs, magnetic or optical cards, flash memory, or other type of media/machine-readable medium suitable for storing electronic instructions. Moreover, the present invention may also be downloaded as a computer program product, wherein the program may be transferred from a remote computer (e.g., a server) to a requesting computer (e.g., a client) by way of data signals embodied in a carrier wave or other propagation medium via a communication link (e.g., a modem or network connection). Accordingly, herein, a carrier wave shall be regarded as comprising a machine-readable medium.

Client Overview

According to one embodiment, an ultra-thin client-side viewer provides the graphics, linking, caching, and function handling capabilities necessary for extending the web to almost any platform. It is designed as a lightweight browser (micro-browser) running directly on device operating systems. In alternative embodiments, the client-side viewer may be deployed as a standard browser plug-in, or Java applet for extending browser functionality. In one embodiment, the client-side viewer attains its small size and efficiency by taking advantage of the power of SVF (Simple Vector Format) to describe almost any current web content. SVF files can be handled with a tiny fraction of the client code required by normal web browsers because current browsers must interpret a large and growing number of file types and their idiosyncrasies. SVF was originally designed to handle a superset of the most commonly used file formats in the complex world of CAD. It can accommodate not only new graphical functions, but the storage and transfer of almost any foreseeable new functional capability. SVF has been under consideration by the W3C (World Wide Web Consortium) for adoption as a standard for vector content on the World Wide Web.

By working tightly with a server-side content translator, web content and functionality can be passed seamlessly to the end user platform without any degradation in the look or feel of the output. In addition, because the resulting file graphics

5

are handled as vectors, the end user can control real time changes in the size of text and graphics as well as what portion of the file is viewable in the display. This “zoom and pan” capability, familiar to CAD and other vector content software users, adds dramatically to the usability of non-standard display sizes. For very small displays, real time zooming and panning allows the user to see graphics and text at sizes that make them easily readable, and then “back up” to view an entire page for context or pan in any direction for navigation. Because the client-side viewer manipulates vectors, there is no loss in quality as the display is zoomed. The graphics rendering engine within the client is so efficient that file manipulation happens in a fraction of a second. There is no perceptible wait for the user as the file is resized, or the window is repositioned. Content created for one display resolution now can be sized, real time, for any other display without degradation. Besides making small displays eminently usable, this technology extends web content into some surprising new arenas. For example, it enables normal desktop displays to be effective for individuals with visual impairment, or for content designed for 640x480 standard PC monitors to be shown without degradation on web billboards now appearing in cities like Seattle and San Francisco.

With a client of such extraordinary power packed in a tiny footprint, end user device manufacturers can free up valuable memory space for pre-fetching, caching and pre-loading content, dramatically improving performance for use in low bandwidth and portable applications. In the example of a wireless handheld device where expensive flash memory must be used instead of more cost effective bulk storage technology, the difference between consuming 10’s of megabytes of flash memory with a standard browser versus running the client-side viewer described herein is dramatic.

Those “saved” megabytes of memory are now available for impressive interfaces, caching of often used content, and pre-fetching of intelligently selected linked files or pre-loading of content for targeted applications. For example, in a mapping application, the map tiles surrounding the viewed map could be downloaded and stored while the user was working with the initial tile, enabling an experience remarkably free from the current frustrations of waiting for a new map to be transferred for even the smallest change in magnification or coverage. If the user knows ahead of time what city they will visit on a business trip, maps and additional travel information in great detail could also be pre-loaded using a high bandwidth connection at home or in the office before heading out to shop or conduct business in the city. Additionally, SVF is a more efficient way to store web content. Resulting content files are reduced in size by anywhere from 20 to 80 percent over their source. SVF is also very compressible. With target file size reduction in the range of 90%, SVF files can take up as little as $\frac{1}{10}^{th}$ the space of the web files in current use. This means that pre-translated content can be moved up to 10 times the rate of current web pages, and as much as 10 times as many pages, maps, stock charts, etc. can be stored for instant retrieval on the hand held platform as can be handled with current web technology.

When used on content created natively in SVF, additional capability can be extended to the client-side viewer.

Graphing the performance of stocks over time is only one use of SVF’s ability to handle streams of data. Handling the output from financial systems, transactional systems, ERP packages, and CRM systems becomes easier and more flexible. Of course, systems integrators don’t have to use these powerful capabilities to start with. If the target system pro-

6

vides web interfaces, these can be viewed, as designed, with no additional software to write, and no changes to the design or layout of the interface.

Server Overview

5 Enabling the client-side viewer to be so small and powerful is the server-side content translator. The server-side content translator rapidly translates Web content to SVF, compresses and encrypts the SVF results if desired, and transfers the vector formatted results to the client-side viewer. Alternatively, SVF files can be cached or stored in a file system for fetching and transfer at a later time. Pre-translated or cached content transfers are significantly faster as no conversion overhead is incurred, and file sizes are reduced using the more efficient SVF. Combine that with standard compression algorithms selectable for use with the client-side viewer for additional performance improvements.

During the translation process, and in the process of serving cached, pre-translated, or native SVF content, output files are “streamed” to the client-side viewer. Although this does not decrease the total time for file transfer, it can significantly improve the effective system performance for the end user. Content can be selectively streamed, with text and links coming through first, followed by graphic images and other content, for example. Should the user be accessing a link, rather than having interest in the entire file served, links can be selected early in the transfer and the next file download started immediately. In addition to streaming, the server-side content converter may also layer the content by type. This means that text can be put in one layer, links in another, GIF images in another, Javascript in another and so on. Layers can be turned on or off depending upon client capabilities, making files for less capable clients, or for users interested in a reduced functionality, higher transfer performance mode to be handled automatically.

35 All operational modes may be controlled through an administrative interface or accessible through a straightforward API (Application Program Interface). Furthermore, the system works with existing firewalls and within standard security protocols. In more secure modes, the server-side content converter and the client-side viewer may operate using Public/Private key authentication and encryption.

Exemplary System Infrastructures

In the following paragraphs, a description of three exemplary system infrastructures is provided. Schematic illustrations of these system infrastructures are shown in FIGS. 1A, 1B, and 1C. It is noted that like-numbered components in these Figures perform substantially the same function. Therefore, any discussion of the functions of a component with reference to one or more of the infrastructures generally may apply to the other infrastructures as well, unless specifically noted otherwise.

A first of exemplary system infrastructure 10A for implementing the invention is shown in FIG. 1A. Infrastructure 10A enables various clients, including wireless devices such as a cellular phone 12, a wireless-enabled PDA 14, and a wireless-enabled laptop computer 16, as well as landline computers 18, 20, and 22, to request content that is accessible via a network such as the Internet 24 to be retrieved from selected network resources, including web servers 26 and 28 and an FTP site 30, wherein the content is translated into a scalable vector representation (e.g., SVF, also referred to herein as “vectorized content”) through use of a proxy server 32 and sent to the requesting client. Upon being received by the client, the vectorized content is processed and rendered using a thin client to enable a user to view the content on the client device.

7

With reference to the flowchart of FIG. 2A, the foregoing process is initiated by a client in a block 100, wherein the client submits a request to proxy server 32 to retrieve and convert selected content. As depicted by a transfer path 34, this comprises sending data 36, which includes content network location indicia from which the content can be retrieved and proxy server network location information by which the content request may be delivered to over Internet 24 to proxy server 32. Typically, it will be desired to retrieve a particular web page. Accordingly, the content network location indicia will comprise a URL (uniform resource locator) for the web page. Similarly, the proxy server network location information may also comprise a URL corresponding to a network access point for the proxy server. Optionally, the location information may comprise a network IP address for one or both of the content location and the proxy server location. If the content is to be retrieved from an Internet resource, the request will typically be sent using the HyperText Transfer Protocol (HTTP) over the TCP/IP transport.

Next, in a block 102, the request is received by the proxy server and the proxy server checks its cache to see if it already has the request content in its cache. If it does, it sends this cached content back to the client. If it does not have the requested content cached, the proxy server sends out a request to retrieve the content from the network resource. For illustrative purposes, it will be assumed for the present example that the desired content comprises a web page that is stored on web server 26. Typically, when the requested content comprises a web page, the content may be retrieved using conventional web content retrieval techniques, such as that employed by various modem browser clients, including Netscape Navigator and Internet Explorer. This generally comprises providing routing information, such as the URL for the web page (URL 38) to routing services provided by Internet 24, which routes the request to an appropriate network resource (e.g., web server 26), as depicted by a transfer path 40.

Typically, the URL will correspond to a web page whose content is stored by the web server in an HTML (HyperText Markup Language) document comprising HTML code and embedded text content, in addition to other optional content languages, that may contain references to other objects (e.g., HTML documents and graphic image files) stored locally to the server or stored on a remote server. For example, the HTML content corresponding to a single-frame web page is often stored in a single file, while multiple-frame web pages may comprise content that is stored in a single file or in multiple files. These files may be stored locally on the web server (e.g., on one of the server's hard disks), or on a local storage device connected to the web server via a local area network (LAN), such as a network attached storage (NAS) filer. Optionally, some of the web page's content may comprise one or more documents that are stored at remote locations that may be accessed via a WAN (wide area network) or the Internet.

HTML is a standardized language that describes the layout of content on a web page, and attributes of that content. This layout and attribute information is defined by sets of tags contained in HTML code corresponding to the page. The tags define various HTML layout and display information, including tables, paragraph boundaries, graphic image positions and bounding box sizes, typeface styles, sizes, and colors, borders, and other presentation attributes. A portion or all of a web page's text content may be contained in the parent HTML document corresponding to the URL. In addition to basic HTML, web page documents may contain XML (eXtensible markup language) code, as well as scripting lan-

8

guage code, such as javascript. However, for simplicity, any documents containing web page content other than only graphic content that are discussed herein will be referred to as HTML documents.

In addition to HTML and other markup and scripting language content, it is very common for web pages to include graphical content. In general, graphical content is usually stored in an image file or files that are external from the parent HTML document for the web page. For example, the parent HTML document may contain one or more embedded image tags that reference the location where those images are stored. As before, the graphic images may be stored locally, or may be stored on remote servers that are accessed by the web server via a WAN, or the Internet. These files will typically comprise data stored in one of several well-known graphic formats, including bitmap files (BMP), GIF (Graphics Interchange Format) files, and JPEG (Joint Photographic Experts Group) files.

In response to receiving the request for content, web server 26 begins sending a parent HTML document 42 back to proxy server 32 in a block 104. In a block 106, the HTML content of the parent HTML document is parsed to search for references to external objects such as HTML frames and graphics. In a decision block 108, a determination is made to whether any references are found. For each reference to an external object that is found, proxy server 32 requests to have the object retrieved from an appropriate network resource (e.g., a web server) in a block 110, and data corresponding to the object is transmitted back to the proxy server, as depicted by locally accessible HTML documents 44 and graphic images 46, as well as remotely accessible HTML documents 48 and graphic images 50, which may be accessed via web server 28. If the external object is a graphic image, there is no further processing of the object at this point. If the object is an HTML document, the functions provided by blocks 106 and 108 are repeated. Generally, this set of processing functions is repeated iteratively until all of the external objects are retrieved. However, as described below, there will be some instances in which certain objects will be retrieved at a later point in time. In addition to content stored on web servers that are accessed using HTTP, content may also be retrieved from various network sites using the File Transfer Protocol (FTP), such as FTP documents 51, which are accessed via FTP server 30.

In general, HTML documents and graphic files will be sent as packetized data streams using HTTP over one or more TCP/IP network connections, wherein the data streams will usually be asynchronous. Retrieval of HTML documents and graphic files corresponding to the embedded references will usually require additional transfer time. Furthermore, graphic content oftentimes comprises significantly larger file sizes than HTML content, leading to significant transfer times in some instances. For simplicity, the transfer of the various HTML documents and graphic files for the content request are depicted by HTML documents 52 and graphic documents 54, which are transferred over a transfer path 56.

When the HTML documents and graphic content are received by proxy server 32, a scalable vector representation of the web page is generated in a block 114 by an HTML translator 58. In brief, HTML translator 58 translates HTML, XML, and cascaded style sheet (CSS) layout content into a scalable vector representation, such as SVF. Details of the HTML translation process are contained below. In addition, the graphic images are converted into a compressed bitmap format in a block 116 by a graphics translator 60. The vectorized content 62 and compressed bitmaps 64 are then streamed back to the client (i.e., computer 18) in a block 118,

as depicted by a transfer path 66. In one embodiment, the content portions are sent in separate streams using multiple connections. In another embodiment, the content portions are sent via a multiplexed stream using a single connection. As the vectorized content and compressed bitmap data are received by the client device, they are processed by a thin client 68 running on the client device, whereby a representation of the original web page content may be rendered on the client device's display screen at various user-selectable scaled resolutions and pan offsets in a block 120, thereby enabling a user to more clearly see an overview or details in the web page. Further details of the client side processing are provided below.

As discussed above, wireless clients may also access the vectorized network (e.g., web site) content provided via proxy server 24. The majority of this process is identical to that described above for land-line clients (e.g., computers 18, 20, and 22), except for provisions required for sending data to and receiving data from wireless devices. In general, most wireless devices will access the Internet via a wireless service provider (i.e., a wireless telecommunications carrier) that is particular to that wireless device. Accordingly, a portion of the transmission path to and from proxy server 24 will comprise infrastructure provided by that service provider and/or shared with other service providers. For simplicity, this infrastructure is shown as a cellular tower 70 and a service provider data center 72, although it will be understood by those skilled in the art that the connection path may comprise additional infrastructure components, including appropriate gateways and routers, that enable wireless devices to access proxy server 24.

In some implementations, there will be no special formatting/protocol services that need to be performed by proxy service 24—from the viewpoint of the proxy service, it will be immaterial whether the client is a land-based or wireless client; the special handling provisions for wireless devices will be handled entirely by the service providers infrastructure transparently at both ends of the communications path. In other instances, it may be desired or necessary to reformat the data content delivered to the wireless device at the proxy service. This will generally be dependent on the particular wireless protocol used, and what services are provided by the service provider for the wireless client.

Currently, in the United States, wireless clients generally access Internet 24 by using the Wireless Application Protocol (WAP). In Japan, the most popular access means is NTT DoCoMo's i-Mode wireless protocol. In addition to these wireless standards, new standards are anticipated to be in force in the near future, including NTT DoCoMo's FOMA (Freedom of Mobile Multimedia Access), which is transported over W-CDMA (Wideband Code Division Multiple Access), and CDMA-2000. For the purposes of the invention herein, it will be understood that those skilled in the mobile telecommunications arts will be knowledgeable about any particular format and/or transport protocol requirements that pertain to the particular protocol that is to be used.

A second exemplary system infrastructure 10B for implementing the invention is shown in FIG. 1B. As will be readily recognized, much of infrastructure 10B is similar to infrastructure 10A; however, rather than have a separate proxy server perform the proxy functions (retrieve and translate content), these functions are performed on machines operated by the web site in infrastructure 10B.

The logic implemented by the invention when providing content to a client using infrastructure 10B is illustrated in the flowchart of FIG. 2B, wherein the process begins in a block 101 in which the client sends a content request 39 directly to

the network site (e.g., web server 26), as depicted by a transfer path 41. In a block 103, HTTP negotiations are performed to determine the format the content is to be delivered in. For example, the request may contain indicia identifying the type of content requested, such as an SVF MIME type (e.g., image/vnd.svf). This is to inform the web server that the request is for specially-formatted content rather than conventional content. The server first checks to see if it already has cached the requested content. If it has, it sends the content to the requesting client; otherwise, it retrieves the parent HTML document in a block 107. It then performs processing steps in blocks 107, 109, and 111 to retrieve content referenced through embedded tags in a manner substantially similar to that discussed above with reference to respective blocks 106, 108, and 110. The primary difference in this instance is that the web server does not receive requests from or send documents to a proxy server—rather, the content is retrieved and processed at the web server, wherein the retrieved content may be stored local to the web server or retrieved from a remote server in a manner similar to that described above.

As before, the retrieved HTML documents are translated into scalable vector representations by HTML translator 58 in a block 114, while the graphic images are translated into a compressed bitmap format by image translator 60 in a block 116, as depicted by vectorized content 62 and bitmap content 64. The vectorized content and bitmap content are then streamed from the web server to the client in a block 119, as depicted by a transfer path 67. Upon arriving at the client, the vectorized content and bitmap content are processed, scaled, and rendered on the client in a block 120.

A third exemplary system infrastructure 10C for implementing the invention is shown in FIG. 1C. In this configuration, the proxy functions are performed at the client. As shown by a block 113 in FIG. 2C, the process for providing vectorized content to a client in accordance with infrastructure 10C begins in a block 113, in which the client sends a content request 37 to a network site, such as web server 26, via Internet 24. In response, the network site retrieves the parent HTML document and sends it to the requesting client in a block 115. In a manner similar to that discussed above with reference to blocks 106, 108, and 110 of FIG. 1A, the client first parses the parent HTML document searching for embedded references to external objects and retrieves these objects, whereupon the embedded reference search is performed on the newly retrieved document until all of the content corresponding to the original content request has been retrieved. This content is depicted by HTML documents 52 and image files 54, which are sent from the network site to the client via a transfer path 69. At this point, the client performs translations on the HTML content and the graphic image content that are substantially similar to that performed by the proxy server in FIG. 1A or at the web site in FIG. 1B, as provided by blocks 114 and 116. The vectorized and image content is then processed and scaled by thin client 68 in a block 120, as depicted by device output 71.

Attention now is focused on the functionality provided by proxy server 24 in system infrastructure 10A of FIG. 1. Fundamentally, the proxy server functions as a proxy. It accepts requests for content from client devices as full URLs using standard HTTP mechanisms carried over a multiplexed TCP connection. Standard HTTP content negotiations features specify the formats in which content is to be delivered (SVF, bitmap, and possibly others, which can be handed off to cooperating client-side display software). As described in further details below, in some embodiments the proxy server appears for the client as a normal proxy (that is, the client

11

knows it is retrieving content via the proxy), while in other embodiments the proxy is transparent to the client.

The proxy server responds to client content requests by delivering content in one of the requested formats, by retrieving the content in an appropriate format from its cache, or from an upstream content source (again using standard HTTP content negotiation features), or by translating upstream content from a supported original format to SVF or the client bitmap format.

Requests from the server installation to its cache and from the cache to upstream content sources are made in HTTP carried over TCP using simple straightforward Web content requests. For example, requests from clients to the proxy server comprise HTTP proxy requests (e.g., "GET http://www.xyz.com/some_page.html HTTP/1.0 . . .") carried over TCP or over a lightweight multiplexing protocol over TCP. The multiplexing protocol allows the server to push image thumbnails to the client before the SVF stream is available, as well as offering a channel for control and status information, more simultaneous channels than the client operating system may support, and a mechanism for prioritizing information flow from server to client under loose client control. In addition to HTTP requests, the proxy server architecture supports other user-level protocols, such as FTP and Gopher.

Details of some of the primary components of the proxy server architecture are shown in FIG. 3. Internally, the proxy server comprises a suite of coordinated processes connecting to upstream content through an HTTP cache 74. In one embodiment all functions except caching are performed in a single process, wherein multiple threads are used to effect asynchronous I/O. Separate processes communicated via persistent multiplexed connections carried over the most efficient reliable transport available (e.g., Unix sockets over single processor and symmetric multiprocessor (SMP) computers; TCP sockets between separate computers). All processes are capable of servicing multiple requests simultaneously. No process maintains client state outside the context of a single request, so all components can be repeated and load balanced across multiple CPU's of an SMP computer or across separate computers on a LAN.

The various content translators used by the proxy server accept (via HTTP PUT) or request (driven by HTTP proxy GET/POST) content in supported, but client-unsupported, formats; and return (via HTTP PUT or GET/POST response) one or more representations of that content in a client-supported format. In the embodiments illustrated in FIG. 1A-C, two translators are used: HTML translator 58 and image translator 60. Future content types may be accommodated by new translators, by extending existing translators to cover the new content types, or by extending the client's capabilities. Standard HTTP content negotiation mechanisms are used to inform the proxy server of the client's capabilities and expectations on each request.

Managers at the proxy server coordinate the operations of other components. Two managers are presently defined; a

12

client manager 73 that handles client proxy requests, and a request manager 75 that handles unproxied HTTP requests from other services. The managers accept requests, attempt to service them from HTTP cache 74, and drive HTML translator 58 and image translator 60 when content does not match the clients' requirements. Managers also handle translator requests for inline content (e.g., image dimensions for page layout), and push translated content into HTTP cache 74. Additionally, the client manager coordinates delivery of primary and inlined content, and provides process and status information to the clients.

As discussed above, HTML translator 58 creates a scalable vector representation of the original HTML content of a requested web page. In order to better explain how translation of HTML content is performed, one embodiment of a translation process is described below as applied to an exemplary web page. In addition, details of conventional web page client and server-side processing are provided so as to clarify how web content is laid out during a pre-rendering process on the client.

FIG. 4 shows a representation of a web page 210 served from an exemplary stock brokerage Internet web site as it would appear when rendered on a modern Internet browser, such as Microsoft's Internet Explorer or Netscape's Navigator. Web page 210 is exemplary of many web pages that implement frames, and includes two adjacent frames 212 and 214. A logo graphic object 216A is displayed at the top of frame 212, which additionally includes a "MARKETS" text header 218A, an "INVESTMENTS" text header 220A, and a plurality of links with overlaying graphic objects, including a "DOW" link 222A, a "NASDAQ" link 224A, an "OPTIONS" link 226A, a "CHARTS" link 228A, a "MUTUAL FUNDS" link 230A, a "IRA, 401K OPTIONS" link 232A, and a "TAX INFORMATION" link 234.

A horizontal group of links 236 is disposed at the top of frame 214, and includes a "QUOTES" link 238A, a "HOT PICKS" link 240A, a "CALENDARS" link 242A, and a "NEWS" link 244A. An advertisement banner 246A is displayed just below the horizontal group of links and just above a "NEWS SPARKS MARKET" headline 248A. Frame 214 also includes a pair of graphic image objects, including a DOW chart 250A and a NASDAQ chart 252A. A set of user input objects is disposed adjacent to DOW chart 250A within a graphic object 254A, including an "ACCOUNT #" input box 255A, an "ACCESS CODE" input box 256A, and a "LOGIN" button 257A. In addition to the foregoing objects, frame 214 also includes text objects 258A and 260A.

An HTML listing corresponding to web page 210 is presented below as LISTING 1. Note that LISTING 1 sometimes refers to object descriptions and link paths rather than the text or path location of actual objects for simplicity, and that other elements commonly found in HTML pages, such as META entries, are omitted for clarity.

Listing 1

```

1. <html>
2. <head><title>"MARKET HOME"</title></head>
3.
4. <body bgcolor="#FFFFFF" link="#0033CC" vlink="#0033CC">
5.
6. <frameset cols="25%,75% frameborder=0 border=0>
7. <frame>
8. <align=left><align=top>
9. 

```

-continued

```

10. <br><br>
11. <t3>TEXT HEADER #1 align=left</t3><br>
12.
13. <table width="90%" border=0 cellspacing=10 cellpadding=0 bgcolor="#000000"
14. align=center>
15. <tr>
16. <a href="URL or path for LINK #5" </a>
18. <tr>
19. <a href="URL or path for LINK #6" </a>
21. <tr>
22. <a href="URL or path for LINK #7" </a>
24. <tr>
25. <a href="URL or path for LINK #8" </a>
27. </table>
28. <br>
29. <t3>TEXT HEADER #1 align=left</t3>
30. <br>
31. <table width="90%" border=0 cellspacing=10 cellpadding=0 bgcolor="#000000"
32. align=center>
33. <tr>
34. <a href="URL or path for LINK #9" </a>
36. <tr>
37. <a href="URL or path for LINK #10" </a>
39. <tr>
40. <a href="URL or path for LINK #11" </a>
42.
43. </table>
44. </frame>
45.
46. <frame>
47.
48. <table>
49. <tr>
50. <table width="100%" border=0 cellspacing=15 cellpadding=0
51. bgcolor="#000000" align=center>
52. <tr>
53. <td><a href="URL or path for link#1"> alt="QUOTES"</a>
54. <td><a href="URL or path for link#2"> alt="HOT PICKS"</a>
55. <td><a href="URL or path for link#3"> alt="CALENDERS"</a>
56. <td><a href="URL or path for link#4"> alt="NEWS"</a>
57. </table><br>
58. <br>
59. 
61. <br><t1>HEADLINE TEXT</t1>
62. <table>
63. <Colgroup span="2">
64. <Col width = "400" align="center">
65. <Col width = "200" align="center">
66. <tr><td>
67. 
69. <td>
70. /* INPUT FOR ACCOUNT NUMBER AND ACCESS CODE */
71. <SCRIPT LANGUAGE = "Javascript">
72. <!--
73. [Javascript variable declarations]
74. [Javascript functions to enable login] ---!>
75. </SCRIPT>
76. <table>
77. <td>
78. 
79. <table width="150" height="25">
80. <td>
81. <font size=-2 face="arial, helvetica, verdana">Account#</font>
82. <tr><input type=text name="USERID" maxlength=9 size=20>
83. <tr><font size=-2 face="arial, helvetica">Access Code:</font>
84. <tr><input type=password name="PASSWORD" maxlength=10 size=20
85. onKeyDown="SuppressEnterBell(event)"
86. onKeyPress="SuppressEnterBell(event)"
87. onKeyUp="SubmitOnEnter(event)">
88. <br>&nbsp;

```


In a similar manner, the foregoing technique is applied to the HTML code in the primary document to identify other types of objects as well. In addition to parsing the primary HTML document, similar processing is performed on referenced documents, such as documents that include frame content that is defined and stored separate from the primary HTML document.

A representation of the results of the functions performed in block 152 are shown in FIG. 4B. In the Figure, objects corresponding to the original content of FIG. 4A are shown with an appended "B" that is added to each object's root reference number, wherein the root reference number for an object is that same as the logically grouped content in FIG. 4A that it corresponds to, e.g., an object 248B is generated for "NEWS SPARKS MARKET" headline 248A, etc.

Next, in a block 154, the page layout is defined based on the bounding boxes. In actuality, generation of the page layout information is performed in conjunction with defining the boundary boxes for the objects, wherein the location of a given object is based on the location of other related (e.g., if within a table) or non-related objects corresponding to HTML content that have been previously parsed. For example, the location of a given paragraph will depend on the other content for the page that are listed prior to the definition for the paragraph in the primary HTML document or referenced document, if applicable. As the HTML content of the primary and any referenced HTML documents are parsed, the page layout is generated based on the various HTML tags and the content embedded between tag pairs and/or referenced by a tag pair statement (e.g., graphic images).

As will be recognized by those skilled in the art, the functions performed in blocks 150, 152, and 154 are commonly performed by conventional browsers during a pre-rendering process. In some browsers, these functions are performed by the Mozilla rendering engine, which comprises open source software that is readily available for use by developers. At present, the software for the Mozilla rendering engine may be accessed via the Internet at www.mozilla.org. Accordingly, in one embodiment, the present invention uses core functionality provided by the Mozilla rendering engine source code to perform the functions of block 150, 152, and 154.

At this point, the present invention deviates substantially from the prior art by using the various object layout data generated during the pre-rendering process to generate a scalable vector representation of the original page content. First, in a block 156, a datum point is defined for the page and the bounding box for each object. For example, as shown in FIG. 4C, a rendered page datum 262 is defined to be coincident with the upper left hand corner of the display frame of the rendered page for the web page. Generally, any point on the page may be used as the page datum—the only requirement is that the page datum that is selected is used consistently throughout the process. The use of the upper left hand corner of the display frame is advantageous since the location of the first object encountered in the HTML code for a page is located relative to this corner.

In general, the datum points for each object may also be located any place on the object, as long as the object datum points are used in a predictable manner. For example, as depicted in FIG. 4C, various datum points for corresponding objects are defined to be coincident with the upper left hand corner of the bounding box for that object, wherein the object's datum point shares the root reference number of the object with an appended "C."

Once the page's datum point and an object's datum point are known, a vector between these points is generated for each object in a block 158. With reference to FIG. 4D, in one

embodiment, wherein the page datum point corresponds to the upper left and corner of the display frame and is assigned an XY value 266 of 0,0, the vector for a given object may be stored as the XY value of the datum point of that object relative to 0,0, such as a value of 150, 225 (ref. num. 268) for a vector 250D pointing to an object datum 250C, and a value of 150, 425 (ref. num. 270) for a vector 252D pointing to an object datum 252C. In another embodiment, each vector may be stored as XY data relative to a 0,0 datum point corresponding to the upper left hand corner of the frame the object belongs to. For example, a vector 250D' from a frame datum 214D to object datum 250C is stored as 20, 200 (ref. num. 268'), while a vector 252D from frame datum 214D to object datum 252C is stored as 20, 425. In this embodiment, offset information for each frame relative to a known datum will also be stored, as depicted by a vector 214D.

The scalable vector representation is completed in a block 160, wherein a reference is created for each object that includes or links an object's content and attributes, such as object type (e.g., text, image), object typeface, and boundary box parameters, to the object's vector. For example, object 250B is a graphic image having a vector 250D and a bounding box that is 180 pixels high and 350 pixels wide, while object 252B is a graphic image having a vector 252D and a bounding box that includes a height of 200 pixels and a width of 350 pixels. This enables client-side operations to be performed that only initially consider the vectors, wherein if it is determined that a vector's endpoint (and/or the bounding box corresponding to the object the vector points to) would appear off of a display, there is no need to retrieve the content and attribute data linked to the vector. This concept is explained in further detail in the following section.

It is noted that a portion of the display content produced on a client device will never contain any rendered content, as this portion is reserved for the browser's user interface. In WINDOWS™ environments, this portion will include the browser's window frame, as well as the pull-down and icon menus provided in the browser's user interface, which are depicted by a box 264 in the Figures herein.

Client-Side Software and Processing

As discussed above, the present invention supports a wide variety of clients, including land-based clients and wireless clients. Each client requires some client-side software that enables the scalable vector content data provided to it to be rendered at a user-selectable scale factor and offset on the client's display, such as a monitor or built-in LCD screen.

By enabling original content from a web site to be displaced in such a resolution-independent manner, users will be able to view content in a manner that did not previously exist, greatly enhancing the user experience. For example, in some implementations the client may be a personal computer (PC). Using a least-common denominator approach, many web pages are designed for a smaller resolution (for example 640×480 pixels, a minimum resolution commonly supported by nearly all PC's, including legacy PC's) than the resolution provided by the video output capabilities available with many of today's PC's, such as 1024×768 pixels, 1280×1024 pixels, and even 1600×1200 pixels. As a result, when these web pages are displayed on a high-resolution display, they occupy only a portion of the display, making portions of the pages, especially those portions containing small text, difficult to read. By enabling users to selectively magnify the entire page, these design flaws are easily overcome. Alternatively, the client may be a small device, such as a hand held computer or a cell phone, which has a smaller display resolution than common Web pages are designed for. As explained below, through use of the invention's scalable vector representation

and client-side processing, users are enabled to view the entire content of billions of existing Web pages using handheld devices in a simple and reasonable way.

In one embodiment, the client software may be a plug-in to a Web browser, such as Netscape Navigator or Microsoft Internet Explorer. Such a plug-in might have the browser download the data and display it in a sub-window of the browser. Alternatively, the client software may be a Java applet running in a browser. As another option, the client software may be a stand-alone program that interfaces with the proxy server or proxy software directly. The client software may bypass the proxy when requesting information that won't be translated to vectors, such as bitmaps.

With reference to FIG. 6, client-side processing proceeds in the following manner. In a block 160, the vector representation data (i.e., vectorized HTML content and compressed bitmap content) for the web page is gathered at the client. Typically, this data will be stored in a cache at the client as it is being received, and the client simply retrieved the data from the cache. In a block 162, a display list of vectors is built. This process is well known in the CAD arts, and is enabling rapid zooming of vector-based objects. In a block 164, user selectable scale and offset (pan) values are determined. Based on various user interactions with the user-interface of the client, the user is enabled to control the zoom (size) and offset of the rendered page. For example, suppose the user provides zoom and offset inputs to produce a rendered page 210E, as shown in FIG. 4E. In this rendered page, the original origin is now off of the screen (the page image is shifted upward and toward the left—see FIG. 4F), and the view has been scaled approximately 1.3 times.

Next, in a block 166, the vectors and boundary boxes are processed based on the scale and offset, and a bounding box defining the limits of the display content is determined. The results of this step are shown in FIG. 4F, while FIG. 4G shows specific details on how the vectors and bounding boxes corresponding to image objects 250B and 250B' (now 250B' and 252B', respectively) are processed. Logically, there are generally two ways to scale and offset the rendered content. In one embodiment, vectors and bounding boxes are mapped to a virtual display area in memory that has much greater resolution (e.g., 100,000×100,000 pixels) than any real display, and a virtual display limit bounding box is scaled and moved around over the virtual display area. Accordingly, during subsequent processing described below, objects falling within the display bounding box are rendered by reducing the scaling of those objects in the virtual display to how the objects will appear on the client device display relative to the virtual display bounding box. In the alternate, a fixed reference frame corresponding to the display resolution of the client device screen is maintained, wherein all vectors and bounding boxes are scaled and offset relative to the fixed reference frame. Each scheme has its advantages and disadvantages. One advantage of the second method is that the display bounding box is always maintained to have a size that matches the resolution of the content display area on the client device.

As shown in FIG. 4G, respective offsets in X and Y, ($-\Delta X$ and $-\Delta Y$ in the Figure) are applied to the starting point of each of the vectors. The vectors are then scaled by a scale factor "SF." The results of the new vectors are depicted by vectors 250D" and 252D". This produces a new datum for each object's bounding box that is relative to rendered page datum 262, which remains fixed. As discussed above, only a portion of the display screen will actually be used to display content (as defined by a display limit bounding box 266 in this embodiment), while other portions of the screen, including box 264, will comprise a generally fixed-size user interface.

Accordingly, rendered page datum 262 is not located at the upper left hand corner of the display area, although it possibly could be located at this point when either the current user interface is inactive (i.e., the display portion of the user interface is temporary disabled) or the user interface is contained in other portions of the display.

This foregoing process establishes a starting point (the new datum) for where the content in each object's bounding box will be rendered. At this point, each object's bounding box is then drawn from its new datum using the scaling factor. For example, in the original web page 210D (FIG. 4D), bounding box 250B had an X-axis datum of 150 pixels, a Y-axis datum of 225 pixels, and a height and width of 180×350 pixels. In contrast, after being offset and scaled, bounding box 250B' has an X-axis datum of $150 * SF - \Delta X$, a Y-axis datum of $225 * SF - \Delta Y$, and a height and width of $180 * SF \times 350 * SF$.

Returning to the flowchart of FIG. 6, once the vectors and bounding boxes are offset and scaled, content corresponding to objects having at least a portion of their bounding boxes falling within the display limit bounding box is retrieved from the client device's display list in a block 168. For examples, as shown in FIG. 4F, content corresponding to all of the objects except for those falling entirely outside of display limit bounding box 266 (objects 216, 238, 240, 242 and 244) is retrieved from the display list. That content is then scaled in a block 170. For image content, this comprises decompressing and scaling the compressed bitmaps corresponding to those images. For text content, this comprises scaling the font (i.e., typeface) that the text content portions of the web page are written in the parent HTML document and any referenced documents. There are various techniques for typeface scaling that may be implemented here, depending on the available resources provided by the operating system of the client device. For example, for WINDOWS™ operating systems, many TRUETYPE™ fonts are available, which use a common scalable definition for each font, enabling those fonts to be scaled to just about any size. In other cases, such as current PDA (e.g., Palm Pilots) operating systems, there is no existing feature that supports scaling fonts. As a result, bitmapped fonts of different font sizes and styles may be used. In addition to scaling image and text content, other types of content, such as separator lines and borders may also be scaled by block 170.

The process is completed in a block 172, wherein those portions of the scaled content falling within the display limit bounding box are rendered on the client device's display.

As discussed above, it is foreseen that the invention will be used with client devices having small, low resolution displays, such as PDAs and pocket PCs. Examples of various views of an exemplary web pages obtained from the YAHOO™ web site are shown in FIGS. 7A-B, 8A-B and 9A-B. For instance, FIG. 7A represents how the YAHOO™ home page might appear on a Palm IIIc color PDA.

In addition to directly scaling and offsetting content, the client user-interface software for PDA's provides additional functionality. For instance, a user may select to view a column (results represented in FIG. 7B by tapping that column with a stylus, as shown in FIG. 7A). Similarly, the user may select to zoom in on an image by tapping the image with the stylus, as shown in FIGS. 8A and 8B, or select to view a paragraph in an article by tapping on the paragraph, as shown in FIGS. 9A and 9B. It is noted that in some instances, the display of the paragraph may be reformatted to fit the characteristics of the display, rather than following the original format in the zoom-out view.

21

It is further noted that that different scaling factors can be applied to the X and Y axis so as to change the aspect ratio of the display. For example, a Web page may be designed to be displayed on a computer having a resolution of 800×600 pixels, or a 4X to 3Y aspect ratio. In this case, the display corresponds to a “landscape” layout, wherein there are more pixels along the X axis than along the Y axis. Conversely, many handheld devices display images having a “portrait” layout, wherein there are more pixels along the Y axis than the X axis. By enabling different scaling factors to be applied to the X and Y axes, the present invention enables the aspect ratio of a rendered display image to be adjusted to better fit the aspect ratio of the client device.

An Exemplary Computer Architecture

An exemplary machine in the form of a computer system 500 in which features of the present invention may be implemented will now be described with reference to FIG. 10. Computer system 500 may represent a workstation, host, server, print server, or printer controller. Computer system 500 comprises a bus or other communication means 501 for communicating information, and a processing means such as processor 502 coupled with bus 501 for processing information. Computer system 500 further comprises a random access memory (RAM) or other dynamic storage device 504 (referred to as main memory), coupled to bus 501 for storing information and instructions to be executed by processor 502. Main memory 504 also may be used for storing temporary variables or other intermediate information during execution of instructions by processor 502. Computer system 500 also comprises a read only memory (ROM) and/or other static storage device 506 coupled to bus 501 for storing static information and instructions for processor 502.

A data storage device 507 such as a magnetic disk or optical disc and its corresponding drive may also be coupled to bus 501 for storing information and instructions. Computer system 500 can also be coupled via bus 501 to a display device 521, such as a cathode ray tube (CRT) or Liquid Crystal Display (LCD), for displaying information to an end user. Typically, an alphanumeric input device 522, including alphanumeric and other keys, may be coupled to bus 501 for communicating information and/or command selections to processor 502. Another type of user input device is cursor control 523, such as a mouse, a trackball, or cursor direction keys for communicating direction information and command selections to processor 502 and for controlling cursor movement on display 521.

A communication device 525 is also coupled to bus 501. Depending upon the particular presentation environment implementation, the communication device 525 may include a modem, a network interface card, or other well-known interface devices, such as those used for coupling to Ethernet, token ring, or other types of physical attachment for purposes of providing a communication link to support a local or wide area network, for example. In any event, in this manner, the computer system 500 may be coupled to a number of clients and/or servers via a conventional network infrastructure, such as a company’s Intranet and/or the Internet, for example.

Importantly, the present invention is not limited to having all of the routines located on the same computer system. Rather, individual objects, program elements, or portions thereof may be spread over a distributed network of computer systems. Additionally, it is appreciated that a lesser or more equipped computer system than the example described above may be desirable for certain implementations. Therefore, the configuration of computer system 500 will vary from implementation to implementation depending upon numerous factors, such as price constraints, performance requirements,

22

and/or other circumstances. For example, according to one embodiment of the present invention, a cell phone or a hand held computer may comprise only a processor or a micro controller and a memory, such as a micro code ROM or RAM, for storing static or dynamically loaded instructions and/or data.

In the foregoing specification, the invention has been described with reference to specific embodiments thereof. It will, however, be evident that various modifications and changes may be made thereto without departing from the broader spirit and scope of the invention. The specification and drawings are, accordingly, to be regarded in an illustrative rather than a restrictive sense.

What is claimed is:

1. A wireless device, comprising:

processing means;

wireless communications means, to facilitate wireless communication with a network that supports access to the Internet;

a display;

memory; and

storage means, in which a plurality of instructions are stored that when executed by the processing means enable the wireless device to perform operations including,

rendering a browser interface via which a user is enabled to request access to an original Web page, the Web page comprising HTML-based Web content having an original format defining an original width and height of the Web page and an original page layout, functionality, and design of content on the Web page; in response to a user request to access the Web page,

retrieving the Web page via the wireless communication means, and translating at least a portion of the HTML-based Web content from its original format into scalable content that supports a scalable resolution-independent representation of the Web page that preserves the original page layout, functionality and design of the content defined by its original format when scaled and rendered; and scaling the scalable content to render the Web page on the display such that a width of the Web page is rendered to fit across the display.

2. The wireless device of claim 1, wherein execution of the instructions performs further operations comprising enabling the user to zoom in on a user-selectable portion of a display of the Web page in response to a corresponding user interface input.

3. The wireless device of claim 2, wherein the display of the Web page is re-rendered in real-time to effect zooming operations.

4. The wireless device of claim 1, wherein the Web page includes at least one hyperlink, and wherein execution of the instructions performs further operations comprising:

enabling the user to select the hyperlink; and, in response thereto,

retrieving and translating Web content associated with the hyperlink to produce additional scalable content; and

employing the additional scalable content to render the Web content associated with the hyperlink on the display.

5. The wireless device of claim 1, wherein execution of the instructions performs further operations comprising:

parsing markup language code to determine the original page layout of display content within the Web page, wherein the original page layout defines a layout loca-

23

tion for a plurality of objects, including at least one of text objects, graphic layout objects, or graphic image objects included in the Web page;

defining a primary datum corresponding to the original page layout; and, for each object,

defining an object datum corresponding to the layout location for the object;

generating a vector from the primary datum to the object datum for the object; and

creating a reference that links the object to the vector that is generated.

6. The wireless device of claim 1, wherein execution of the instructions performs further operations comprising enabling the Web page to be displayed at different resolutions by scaling the scalable content to re-render the display in response to associated user inputs,

wherein the original page layout, functionality, and design of the Web page content are preserved at each of the different resolutions.

7. The wireless device of claim 1, wherein execution of the instructions performs further operations comprising returning the display of the Web page to a previous view in response to a corresponding user input.

8. The wireless device of claim 1, wherein execution of the instructions performs further operations comprising enabling a user to pan a view of the Web page in response to a corresponding user input.

9. The wireless device of claim 8, wherein execution of the instructions performs further operations comprising enabling the view of the Web page to be panned in real-time.

10. The wireless device of claim 1, wherein the page layout of the Web page is defined to have an original aspect ratio, and wherein the scalable content is scaled when rendered so as to produce a display of the Web page having a different aspect ratio.

11. The wireless device of claim 1, wherein execution of the instructions performs further operations comprising enabling a user to zoom on a column of the Web page via a corresponding user input, wherein in response thereto, the display is re-rendered such that content corresponding to the selected column is enlarged.

12. The wireless device of claim 11, wherein the content of the column is reformatted to fit characteristics of the display when the display is re-rendered.

13. The wireless device of claim 11, wherein the corresponding user input comprises tapping on the column via the display.

14. The wireless device of claim 11, wherein the display is re-rendered such that content corresponding to the selected column is rendered to fit across the display.

15. The wireless device of claim 1, wherein the Web content includes at least one image, and wherein execution of the instructions performs further operations comprising enabling a user to zoom on an image via a corresponding user input, wherein in response thereto, the display is re-rendered such that the image is enlarged.

16. The wireless device of claim 15, wherein the corresponding user input comprises tapping on the image via the display.

17. The wireless device of claim 15, wherein the display is re-rendered such that the image is rendered to fit across the display.

18. The wireless device of claim 1, wherein execution of the instructions performs further operations comprising enabling a user to zoom on a paragraph of the Web content via a corresponding user input, wherein in response thereto, the

24

display is re-rendered such that content corresponding to the selected paragraph is enlarged.

19. The wireless device of claim 18, wherein the content of the paragraph is reformatted to fit characteristics of the display when the display is re-rendered.

20. The wireless device of claim 18, wherein the corresponding user input comprises tapping on the paragraph via the display.

21. The wireless device of claim 18, wherein the display is re-rendered such that content corresponding to the selected paragraph is rendered to fit across the display.

22. The wireless device of claim 1, wherein the Web page includes text, layout attributes, and images, and wherein execution of the instructions performs further operations comprising:

receiving content corresponding to the text and layout attributes via a first connection; and

receiving content corresponding to at least one image via a second connection.

23. The wireless device of claim 1, wherein execution of the instructions performs further operations comprising:

generating a vector-based display list associated with the scalable content; and

employing the display list to re-render the display at different scale factors to zoom the Web page.

24. The wireless device of claim 1, wherein execution of the instructions performs further operations comprising:

parsing markup language code corresponding to the retrieved Web page to determine the original page layout of the content on the Web page;

logically grouping selected content into objects;

defining a primary datum corresponding to the original page layout; and,

for each object,

defining an object datum corresponding to a layout location datum for the object's associated display content;

generating a vector from the primary datum to the object datum for the object; and

creating a reference that links the object to the vector that is generated.

25. The wireless device of claim 24, wherein execution of the instructions performs further operations comprising:

generating a bounding box for each object, the bounding box representing a portion of a rendered display page occupied by the object's associated group of content.

26. The wireless device of claim 25, wherein execution of the instructions performs further operations comprising:

mapping the object vectors and associated bounding boxes to a virtual display in memory.

27. The wireless device of claim 26, wherein execution of the instructions performs further operations comprising:

enabling a user to view the Web page at a user-selectable zoom level and panned view by,

determining a first scale factor and offset in response to one or more corresponding user inputs defining a user-selectable zoom level and panned view corresponding to a rendered display of the Web page desired by a user; and

determining a virtual display bounding box for the virtual display associated with the first scale factor and offset;

identifying object bounding boxes having at least a portion falling within the virtual display bounding box; and,

for each of such object bounding boxes,

retrieving content associated with that object bounding box;

25

applying an appropriate scale factor to the content associated with that object bounding box to produce scaled content; and rendering the portion of scaled content within the virtual display bounding box to render the content on the display.

28. The wireless device of claim 1, wherein the scalable content includes scalable text content, and wherein execution of the instructions performs further operations comprising scaling a scalable font to render the scalable text content.

29. The wireless device of claim 1, wherein at least a portion of the instructions comprise Java-based instructions.

30. The wireless device of claim 1, wherein the device comprises a mobile phone.

31. The wireless device of claim 1, wherein the device comprises one of a Personal Digital Assistant (PDA) or hand-held computer.

32. The wireless device of claim 1, wherein the network comprises a mobile service provider network.

33. The wireless device of claim 1, wherein a portion of the scalable content comprises vector-based content.

34. The wireless device of claim 1, wherein the device comprises one of a desktop computer, notebook computer or laptop computer.

35. The wireless device of claim 1, wherein the device enables a user to browse, zoom, and pan billions of Web pages in a manner that preserves the original layout, functionality, and design of the HTML-based Web page content of each Web page.

36. A mobile hand-held device, comprising:

a processor,

a wireless communications device, to facilitate wireless communication with a network that supports access to the Internet;

a display; and

flash memory, operatively coupled to the processor, in which a plurality of instructions are stored that when executed by the processor enable the mobile hand-held device to perform operations including,

rendering a browser interface via which a user is enabled to request access to a Web page comprising HTML-based Web content defining an original page layout, functionality, and design of content on the Web page;

retrieving the Web page via the wireless communications device, and processing HTML-based Web content to produce scalable content; and

employing at least one of the scalable content or data derived therefrom to,

render the Web page on the display; and

re-render the display in response to associated user inputs to enable the Web page to be browsed at various zoom levels and panned views while preserving the original page layout, functionality, and design of the Web page content at each zoom level and panned view.

37. The mobile hand-held device of claim 36, wherein the device comprises a mobile phone.

38. The mobile hand-held device of claim 36, wherein the device comprises one of a Personal Digital Assistant (PDA) or hand-held computer.

39. The mobile hand-held device of claim 36, wherein execution of the instructions performs further operations comprising enabling the user to zoom in on a user-selectable portion of a display of the Web page in response to a user interface input.

26

40. The mobile hand-held device of claim 39, wherein the user interface input enables the user to define an area of a current view of the Web page on which to zoom in on.

41. The mobile hand-held device of claim 36, wherein the display of the Web page is re-rendered in real-time to effect zooming operations.

42. The mobile hand-held device of claim 36, wherein the Web page includes at least one hyperlink, and wherein execution of the instructions performs further operations comprising:

enabling the user to select the hyperlink via the display; and, in response thereto,

retrieving and processing HTML-based Web content associated with the hyperlink to produce additional scalable content; and

employing at least one of the additional scalable content or data derived therefrom to render the Web content associated with the hyperlink on the display.

43. The mobile hand-held device of claim 36, wherein at least a portion of the scalable content comprises scalable vector-based content.

44. The mobile hand-held device of claim 36, wherein execution of the instructions performs further operations comprising returning the display of the Web page to a previous view in response to a corresponding user input made via the display.

45. The mobile hand-held device of claim 36, wherein execution of the instructions performs further operations comprising enabling a user to pan a view of the Web page in response to a corresponding user input made via the display.

46. The mobile hand-held device of claim 36, wherein execution of the instructions performs further operations comprising enabling the panned view of the Web page to be panned in real-time.

47. The mobile hand-held device of claim 36, wherein the page layout of the Web page is defined to have an original aspect ratio, and wherein said at least one of scalable content or data derived therefrom is scaled to render a display of the Web page having a different aspect ratio.

48. The mobile hand-held device of claim 36, wherein execution of the instructions performs further operations comprising enabling a user to zoom on a column of the Web content via a corresponding user input, wherein in response thereto, the display is re-rendered such that content corresponding to the selected column is rendered to fit across the display.

49. The mobile hand-held device of claim 48, wherein the content of the column is reformatted to fit characteristics of the display when the display is re-rendered.

50. The mobile hand-held device of claim 48, wherein the corresponding user input comprises tapping on the column via the display.

51. The mobile hand-held device of claim 36, wherein the Web content includes at least one image, and wherein execution of the instructions performs further operations comprising enabling a user to zoom on an image via a corresponding user input, wherein in response thereto, the display is re-rendered such that the image is rendered to fit across the display.

52. The mobile hand-held device of claim 51, wherein the corresponding user input comprises tapping on the image via the display.

53. The mobile hand-held device of claim 36, wherein execution of the instructions performs further operations comprising enabling a user to zoom on a paragraph of the Web page via a corresponding user input, wherein in response

27

thereto, the display is re-rendered such that content corresponding to the selected paragraph is rendered to fit across a display area of the display.

54. The mobile hand-held device of claim 53, wherein the content of the paragraph is reformatted to fit characteristics of the display area when the display is re-rendered.

55. The mobile hand-held device of claim 53, wherein the corresponding user input comprises tapping on the paragraph via the display.

56. The mobile hand-held device of claim 36, wherein the Web page includes text, layout attributes, and images, and wherein execution of the instructions performs further operations comprising:

receiving content corresponding to the text and layout attributes via a first connection; and
receiving content corresponding to at least one image via a second connection.

57. The mobile hand-held device of claim 36, further comprising dynamic memory having at least a portion employed for rendering purposes, wherein execution of the instructions performs further operations comprising:

building a display list via use of the scalable content and rendering display list content on a virtual display in the dynamic memory; and
scaling the display list content to re-render the display of the Web page.

58. The mobile hand-held device of claim 36, wherein execution of the instructions performs further operations comprising:

parsing HTML-based code corresponding to the retrieved Web page to determine the original page layout of the content on the Web page;
logically grouping selected content into objects;
defining a primary datum corresponding to the original page layout; and,
for each object,
defining an object datum corresponding to a layout location datum for the object's associated display content;
generating a vector from the primary datum to the object datum for the object; and
creating a reference that links the object to the vector that is generated.

59. The mobile hand-held device of claim 58, wherein execution of the instructions performs further operations comprising:

generating a bounding box for each object, the bounding box representing a portion of a rendered display page occupied by the object's associated group of content.

60. The mobile hand-held device of claim 59, further comprising dynamic memory having at least a portion employed for rendering purposes, wherein execution of the instructions performs further operations comprising:

mapping the object vectors and associated bounding boxes to a virtual display in the dynamic memory.

61. The mobile hand-held device of claim 60, wherein execution of the instructions performs further operations comprising:

enabling a user to view the Web page at a user-selectable zoom level and panned view by,
determining a first scale factor and offset in response to one or more corresponding user inputs defining a user-selectable zoom level and panned view corresponding to a rendered display of the Web page desired by a user;
determining a virtual display bounding box for the virtual display associated with the first scale factor and offset;

28

identifying object bounding boxes having at least a portion falling within the virtual display bounding box; and,

for each of such object bounding boxes,
retrieving content associated with that object bounding box;
applying an appropriate scale factor to the content associated with that object bounding box to produce scaled content; and
rendering the portion of scaled content within the virtual display bounding box to render the content on the display.

62. The mobile hand-held device of claim 36, wherein the scalable content includes scalable text content, and wherein execution of the instructions performs further operations comprising scaling a scalable font to render the scalable text content.

63. The mobile hand-held device of claim 36, wherein the original format of the Web page defines a height and width for the Web page, and wherein execution of the instructions performs further operations comprising:

determining an applicable scale factor to display at least one of the width and height of the Web page to fit across a browser display area of the display; and
employing the scale factor to render the browser display area.

64. The mobile hand-held device of claim 36, wherein at least a portion of the instructions comprise Java-based instructions.

65. The mobile hand-held device of claim 36, wherein a portion of the HTML-based Web content comprises XML code.

66. The mobile hand-held device of claim 36, wherein a portion of the HTML-based Web content comprises cascaded style sheet data defining aspects of the Web page design that are preserved at each zoom level and panned view.

67. The mobile hand-held device of claim 36, wherein the network comprises a Local Area Network or Wide Area Network.

68. The mobile hand-held device of claim 36, wherein the device comprises one of a notebook computer or laptop computer.

69. The mobile device of claim 36, wherein the device enables a user to browse, zoom, and pan billions of Web pages in a manner that preserves the original layout, functionality, and design of the HTML-based Web content of each Web page.

70. A mobile device, comprising:

processing means;
wireless communications means, to facilitate wireless communication with a network that supports access to the Internet;
a display, to facilitate user input and display rendered content; and
storage means, in which a plurality of instructions are stored,
wherein, upon execution of the instructions by the processing means, the mobile device is enabled to perform operations, including,
rendering a browser interface via which a user is enabled to request access to a Web page comprising HTML-based Web content defining an original page layout, functionality, and design of content on the Web page;
retrieving the Web page via the wireless communications means, and processing at least a portion of the HTML-based Web content to produce scalable content; and

29

employing at least one of the scalable content or data derived therefrom to,

render the Web page on the display; and

re-render the display in response to associated user inputs made via the display to enable the Web page to be browsed at various zoom levels and panned views while preserving the original page layout, functionality, and design of the Web page content at each zoom level and panned view of the Web page.

71. The mobile device of claim 70, wherein the processing means includes a general-purpose processor.

72. The mobile device of claim 70, wherein the processing means includes a special-purpose processor.

73. The mobile device of claim 70, wherein execution of the instructions performs further operations comprising enabling the user to zoom in on a user-selectable portion of a display of the Web page in response to a user interface input.

74. The mobile device of claim 73, wherein the user interface input enables the user to define an area of a current view of the Web page on which to zoom in on.

75. The mobile device of claim 70, wherein the display of the Web page is re-rendered in real-time to effect zooming operations.

76. The mobile device of claim 70, wherein execution of the instructions performs further operations comprising enabling a user to pan a view of the Web content in response to a corresponding user interface input made via the display.

77. The mobile device of claim 70, wherein execution of the instructions performs further operations comprising enabling the view of the Web content to be panned in real-time.

78. The mobile device of claim 70, wherein the Web content includes at least one image, and wherein execution of the instructions performs further operations comprising enabling a user to zoom on an image via a corresponding user input, wherein in response thereto, the display is re-rendered such that the image is rendered to fit across the display.

79. The mobile device of claim 78, wherein the corresponding user input comprises tapping on the image via the display.

80. The mobile device of claim 70, further comprising dynamic memory having at least a portion employed for rendering purposes, wherein execution of the instructions performs further operations comprising:

building a display list via use of the scalable content and rendering display list objects on a virtual display in the dynamic memory; and

scaling display list objects to re-render the display of the Web page.

81. The mobile device of claim 70, wherein the network comprises a mobile service provider network.

82. The mobile device of claim 70, wherein the device comprises a mobile phone.

83. The mobile device of claim 70, wherein the device comprises one of a Personal Digital Assistant (PDA) or hand-held computer.

84. The mobile device of claim 70, wherein a portion of the scalable content comprises vector-based content.

85. The mobile device of claim 70, wherein the processing means includes logic circuitry programmed with a portion of the instructions.

86. The mobile device of claim 70, wherein the device comprises one of a notebook computer or laptop computer.

30

87. A mobile hand-held device, comprising:

a processor,

a wireless communications interface, to facilitate wireless communication with a network that supports access to the Internet;

a display; and

non-volatile memory, operatively coupled to the processor, in which a plurality of instructions are stored that when executed by the processor enable the mobile hand-held device to perform operations including,

rendering a browser interface on the display via which a user is enabled to request access to a Web page comprising HTML-based Web content defining an original page layout, functionality, and design of content on the Web page;

in response to a user request of the Web page,

retrieving the Web page via the wireless communications interface;

rendering the Web page such that a width of the Web page is rendered to fit across the display; and

re-rendering the display in response to associated user inputs to enable the Web page to be browsed at various zoom levels and panned views while preserving the original page layout, functionality, and design of the Web page content at each zoom level and panned view.

88. The mobile hand-held device of claim 87, wherein execution of the instructions performs further operations comprising enabling a user to zoom on a column of the Web page via a corresponding user input, wherein in response thereto, the display is re-rendered such that content corresponding to the selected column is enlarged.

89. The mobile hand-held device of claim 88, wherein the content of the column is reformatted to fit characteristics of the display when the display is re-rendered.

90. The mobile hand-held device of claim 88, wherein the corresponding user input comprises tapping on the column via the display.

91. The mobile hand-held device of claim 88, wherein the display is re-rendered such that content corresponding to the selected column is rendered to fit across the display.

92. The mobile hand-held device of claim 87, wherein the Web page includes at least one image, and wherein execution of the instructions performs further operations comprising enabling a user to zoom on an image via a corresponding user input, wherein in response thereto, the display is re-rendered such that the image is enlarged.

93. The mobile hand-held device of claim 92, wherein the corresponding user input comprises tapping on the image via the display.

94. The mobile hand-held device of claim 92, wherein the display is re-rendered such that the image is rendered to fit across the display.

95. The mobile hand-held device of claim 87, wherein execution of the instructions performs further operations comprising enabling a user to zoom on a paragraph of the Web page via a corresponding user input, wherein in response thereto, the display is re-rendered such that content corresponding to the selected paragraph is enlarged.

96. The mobile hand-held device of claim 95, wherein the content of the paragraph is reformatted to fit characteristics of the display when re-rendered.

97. The mobile hand-held device of claim 95, wherein the corresponding user input comprises tapping on the paragraph via the display.

98. The mobile hand-held device of claim 95, wherein the display is re-rendered such that content corresponding to the selected paragraph is rendered to fit across the display.

31

99. The mobile hand-held device of claim 87, wherein execution of the instructions performs further operations comprising enabling a user to pan a view of the web page while in a zoomed state under which a portion of the web page is displayed.

100. The mobile hand-held device of claim 87, wherein execution of the instructions performs further operations comprising returning the display of the Web page to a previous view in response to a corresponding user input.

101. The mobile hand-held device of claim 87, wherein the display of the Web page is re-rendered in real-time to effect zooming operations.

102. The mobile hand-held device of claim 87, wherein execution of the instructions performs further operations comprising enabling a user to pan a view of the Web page in response to a corresponding user input made via the display.

103. The mobile hand-held device of claim 87, wherein execution of the instructions performs further operations comprising enabling the view of the Web page to be panned in real-time.

104. The mobile hand-held device of claim 87, wherein the Web page includes text, layout attributes, and images, and wherein execution of the instructions performs further operations comprising:

- receiving content corresponding to the text and layout attributes via a first connection; and
- receiving content corresponding to at least one image via a second connection.

105. The mobile hand-held device of claim 87, wherein a portion of the HTML-based Web content comprises XML code.

106. The mobile hand-held device of claim 87, wherein a portion of the HTML-based Web content comprises cascaded style sheet data defining aspects of the Web page design that are preserved at each zoom level and panned view.

107. The mobile hand-held device of claim 87, wherein the network comprises a mobile service provider network.

108. The mobile hand-held device of claim 87, wherein the device comprises a mobile phone.

109. The mobile hand-held device of claim 87, wherein the device comprises one of a Personal Digital Assistant (PDA) or hand-held computer.

110. The mobile device of claim 87, wherein the network comprises a Local Area Network or a Wide Area Network.

111. The mobile hand-held device of claim 87, wherein the device enables a user to browse, zoom, and pan billions of Web pages in a manner that preserves the original layout, functionality, and design of the HTML-based Web page content of each Web page.

112. A wireless device, comprising:

- a processor;
- a wireless communications interface, to facilitate wireless communication with a network that supports access to the Internet;
- a display;
- memory; and
- a storage device, on which a plurality of instructions are stored that when executed by the processor enable the wireless device to perform operations including, rendering a browser interface via which a user is enabled to request access to a Web page, the Web page comprising HTML-based Web content having an original format including HTML code defining an original page layout, functionality, and design of corresponding content on the Web page;
- retrieving, via the wireless communications interface, and translating at least a portion of the HTML-based

32

Web content into scalable content that supports a scalable resolution-independent representation of the Web page that preserves the original page layout, functionality and design of the content defined by its original format when scaled and rendered;

employing the scalable content to render the Web page on the display using a first scale factor; and enabling the Web page to be displayed at a different resolution by scaling the scalable content using a second scale factor to re-render the display,

wherein the original page layout, functionality, and design of the Web page content are preserved under both the first and second scale factors.

113. The wireless device of claim 112, wherein the display is re-rendered in real-time.

114. The wireless device of claim 112, wherein the device comprises a hand-held device.

115. The wireless device of claim 112, wherein the device comprises one of a desktop computer, notebook computer or laptop computer.

116. The wireless device of claim 112, wherein execution of the instructions performs further operations comprising enabling a user to pan a view of the Web page in response to a corresponding user input.

117. The wireless device of claim 116, wherein execution of the instructions performs further operations comprising enabling the view of the Web page to be panned in real-time.

118. A method, comprising:

rendering a browser interface on a display of a device via which a user is enabled to request access to a Web page, the Web page comprising HTML-based Web content having an original format defining an original width and height of the Web page and an original page layout, functionality, and design of content on the Web page;

in response to a user request to access the Web page, retrieving the Web page via the device, and translating at least a portion of the HTML-based Web content from its original format into scalable content that supports a scalable resolution-independent representation of the Web page that preserves the original page layout, functionality and design of the content defined by its original format when scaled and rendered; and scaling the scalable content to render the Web page on the display such that a width of the Web page is rendered to fit across the display.

119. The method of claim 118, further comprising enabling the user to zoom in on a user-selectable portion of a display of the Web page in response to a corresponding user interface input.

120. The method of claim 119, wherein the display of the Web page is re-rendered in real-time to effect zooming operations.

121. The method of claim 118, wherein the Web page includes at least one hyperlink, the method further comprising:

enabling the user to select the hyperlink; and, in response thereto, retrieving and translating Web content associated with the hyperlink to produce additional scalable content; and employing the additional scalable content to render the Web content associated with the hyperlink on the display.

122. The method of claim 118, further comprising: parsing markup language code to determine the original page layout of display content within the Web page, wherein the original page layout defines a layout loca-

33

tion for a plurality of objects, including at least one of text objects, graphic layout objects, or graphic image objects included in the Web page;

defining a primary datum corresponding to the original page layout; and,
for each object,

defining an object datum corresponding to the layout location for the object;

generating a vector from the primary datum to the object datum for the object; and

creating a reference that links the object to the vector that is generated.

123. The method of claim **118**, further comprising enabling the Web page to be displayed at different resolutions by scaling the scalable content to re-render the display in response to associated user inputs,

wherein the original page layout, functionality, and design of the Web page content are preserved at each of the different resolutions.

124. The method of claim **118**, further comprising returning the display of the Web page to a previous view in response to a corresponding user input.

125. The method of claim **118**, further comprising enabling a user to pan a view of the Web page in response to a corresponding user input.

126. The method of claim **125**, further comprising enabling the view of the Web page to be panned in real-time.

127. The method of claim **118**, wherein the page layout of the Web page is defined to have an original aspect ratio, and wherein the scalable content is scaled when rendered so as to produce a display of the Web page having a different aspect ratio.

128. The method of claim **118**, further comprising enabling a user to zoom on a column of the Web page via a corresponding user input, wherein in response thereto, the display is re-rendered such that content corresponding to the selected column is rendered to fit across the display.

129. The method of claim **128**, wherein the corresponding user input comprises tapping on the column via the display.

130. The method of claim **128**, wherein the content of the column is reformatted to fit characteristics of the display when the display is re-rendered.

131. The method of claim **118**, wherein the Web content includes at least one image, the method further comprising enabling a user to zoom on an image via a corresponding user input, wherein in response thereto, the display is re-rendered such that the image is rendered to fit across the display.

132. The method of claim **131**, wherein the corresponding user input comprises tapping on the image via the display.

133. The method of claim **118**, further comprising enabling a user to zoom on a paragraph of the Web content via a corresponding user input, wherein in response thereto, the display is re-rendered such that content corresponding to the selected paragraph is rendered to fit across the display.

134. The method of claim **133**, wherein the corresponding user input comprises tapping on the paragraph via the display.

135. The method of claim **133**, wherein the content of the paragraph is reformatted to fit characteristics of the display when the display is re-rendered.

136. The method of claim **118**, wherein the Web page includes text, layout attributes, and images, the method further comprising:

receiving content corresponding to the text and layout attributes via a first connection; and

receiving content corresponding to at least one image via a second connection.

34

137. The method of claim **118**, further comprising: generating a vector-based display list associated with the scalable content; and

employing the display list to re-render the display at different scale factors to zoom the Web page.

138. The method of claim **118**, further comprising: parsing markup language code corresponding to the retrieved Web page to determine the original page layout of the content on the Web page;

logically grouping selected content into objects; defining a primary datum corresponding to the original page layout; and,
for each object,

defining an object datum corresponding to a layout location datum for the object's associated display content;

generating a vector from the primary datum to the object datum for the object; and

creating a reference that links the object to the vector that is generated.

139. The method of claim **138**, further comprising: generating a bounding box for each object, the bounding box representing a portion of a rendered display page occupied by the object's associated group of content.

140. The method of claim **139**, further comprising: mapping the object vectors and associated bounding boxes to a virtual display in memory.

141. The method of claim **140**, further comprising: enabling a user to view the Web page at a user-selectable zoom level and panned view by,

determining a first scale factor and offset in response to one or more corresponding user inputs defining a user-selectable zoom level and panned view corresponding to a rendered display of the Web page desired by a user; and

determining a virtual display bounding box for the virtual display associated with the first scale factor and offset;

identifying object bounding boxes having at least a portion falling within the virtual display bounding box; and,
and,

for each of such object bounding boxes, retrieving content associated with that object bounding box;

applying an appropriate scale factor to the content associated with that object bounding box to produce scaled content; and

rendering the portion of scaled content within the virtual display bounding box to render the content on the display.

142. The method of claim **118**, wherein the scalable content includes scalable text content, the method further comprising scaling a scalable font to render the scalable text content.

143. The method of claim **118**, wherein the method is facilitated, at least in part, via execution of Java-based instructions.

144. The method of claim **118**, wherein the device comprises a mobile phone.

145. The method of claim **118**, wherein the device comprises a hand-held device.

146. The method of claim **118**, further comprising accessing the Internet via a wireless connection to retrieve the Web page.

147. The method of claim **118**, wherein a portion of the scalable content comprises vector-based content.

35

148. The method of claim **118**, wherein the device comprises one of a desktop computer, notebook computer or laptop computer.

149. A method, comprising:

rendering a browser interface on a hand-held device via which a user is enabled to request access to a Web page comprising HTML-based Web content defining an original page layout, functionality, and design of content on the Web page;

retrieving the Web page via the hand-held device, and processing HTML-based Web content to produce scalable content; and

employing at least one of the scalable content or data derived therefrom to,

render the Web page on a display of the hand-held device; and

re-render the display in response to associated user inputs to enable the Web page to be browsed at various zoom levels and panned views while preserving the original page layout, functionality, and design of the Web page content at each zoom level and panned view.

150. The method of claim **149**, wherein the hand-held device comprises a mobile phone.

151. The method of claim **149**, wherein the hand-held device comprises one of a Personal Digital Assistant (PDA) or hand-held computer.

152. The method of claim **149**, further comprising enabling the user to zoom in on a user-selectable portion of a display of the Web page in response to a user interface input.

153. The method of claim **152**, wherein the user interface input enables the user to define an area of a current view of the Web page on which to zoom in on.

154. The method of claim **149**, wherein the display of the Web page is re-rendered in real-time to effect zooming operations.

155. The method of claim **149**, wherein the Web page includes at least one hyperlink, the method further comprising:

enabling the user to select the hyperlink via the display; and, in response thereto,

retrieving and processing HTML-based Web content associated with the hyperlink to produce additional scalable content; and

employing at least one of the additional scalable content or data derived therefrom to render the Web content associated with the hyperlink on the display.

156. The method of claim **149**, wherein at least a portion of the scalable content comprises scalable vector-based content.

157. The method of claim **149**, further comprising returning the display of the Web page to a previous view in response to a corresponding user input made via the display.

158. The method of claim **149**, further comprising enabling a user to pan a view of the Web page in response to a corresponding user input made via the display.

159. The method of claim **149**, further comprising enabling the view of the Web page to be panned in real-time.

160. The method of claim **149**, wherein the page layout of the Web page is defined to have an original aspect ratio, and wherein said at least one of scalable content or data derived therefrom is scaled to render a display of the Web page having a different aspect ratio.

161. The method of claim **149**, further comprising enabling a user to zoom on a column of the Web content via a corresponding user input, wherein in response thereto, the display is re-rendered such that content corresponding to the selected column is enlarged.

36

162. The method of claim **161**, wherein the corresponding user input comprises tapping on the column via the display.

163. The method of claim **161**, wherein the content of the column is reformatted to fit characteristics of the display when the display is re-rendered.

164. The method of claim **161**, wherein the display is re-rendered such that content corresponding to the selected column is rendered to fit across the display.

165. The method of claim **149**, wherein the Web content includes at least one image, the method further comprising enabling a user to zoom on an image via a corresponding user input, wherein in response thereto, the display is re-rendered such that the image is enlarged.

166. The method of claim **165**, wherein the corresponding user input comprises tapping on the image via the display.

167. The method of claim **165**, wherein the display is re-rendered such that the image is rendered to fit across the display.

168. The method of claim **149**, further comprising enabling a user to zoom on a paragraph of the Web page via a corresponding user input, wherein in response thereto, the display is re-rendered such that content corresponding to the selected paragraph is enlarged.

169. The method of claim **168**, wherein the content of the paragraph is reformatted to fit characteristics of the display area when the display is re-rendered.

170. The method of claim **168**, wherein the display is re-rendered such that content corresponding to the selected paragraph is rendered to fit across the display.

171. The method of claim **168**, wherein the corresponding user input comprises tapping on the paragraph via the display.

172. The method of claim **149**, wherein the Web page includes text, layout attributes, and images, the method further comprising:

receiving content corresponding to the text and layout attributes via a first connection; and

receiving content corresponding to at least one image via a second connection.

173. The method of claim **149**, wherein the hand-held device includes dynamic memory having at least a portion employed for rendering purposes, the method further comprising:

building a display list via use of the scalable content and rendering display list content on a virtual display in dynamic memory; and

scaling the display list content to re-render the display of the Web page.

174. The method of claim **149**, further comprising:

parsing HTML-based code corresponding to the retrieved Web page to determine the original page layout of the content on the Web page;

logically grouping selected content into objects;

defining a primary datum corresponding to the original page layout; and,

for each object,

defining an object datum corresponding to a layout location datum for the object's associated display content;

generating a vector from the primary datum to the object datum for the object; and

creating a reference that links the object to the vector that is generated.

175. The method of claim **174**, further comprising:

generating a bounding box for each object, the bounding box representing a portion of a rendered display page occupied by the object's associated group of content.

37

176. The method of claim 175, wherein the hand-held device includes dynamic memory having at least a portion employed for rendering purposes, the method further comprising:

mapping the object vectors and associated bounding boxes to a virtual display in the dynamic memory. 5

177. The method of claim 176, further comprising:

enabling a user to view the Web page at a user-selectable zoom level and panned view by,

determining a first scale factor and offset in response to one or more corresponding user inputs defining a user-selectable zoom level and panned view corresponding to a rendered display of the Web page desired by a user; 10

determining a virtual display bounding box for the virtual display associated with the first scale factor and offset; 15

identifying object bounding boxes having at least a portion falling within the virtual display bounding box; and, 20

for each of such object bounding boxes,

retrieving content associated with that object bounding box;

applying an appropriate scale factor to the content associated with that object bounding box to produce scaled content; and 25

rendering the portion of scaled content within the virtual display bounding box to render the content on the display. 30

178. The method of claim 149, wherein the scalable content includes scalable text content, the method further comprising scaling a scalable font to render the scalable text content. 35

179. The method of claim 149, wherein the original format of the Web page defines a height and width for the Web page, the method further comprising:

determining an applicable scale factor to display at least one of the width and height of the Web page across a browser display area of the display; and 40

employing the scale factor to render the browser display area.

180. The method of claim 149, wherein the method is facilitated, at least in part, via execution of Java-based instructions. 45

181. The method of claim 149, wherein a portion of the HTML-based Web content comprises XML code.

182. The method of claim 149, wherein a portion of the HTML-based Web content comprises cascaded style sheet data defining aspects of the Web page design that are preserved at each zoom level and panned view. 50

183. The method of claim 149, wherein a portion of the scalable content comprises vector-based content. 55

184. The method of claim 149, further comprising enabling a user to browse, zoom, and pan billions of Web pages in a manner that preserves the original layout, functionality, and design of the HTML-based Web page content of each Web page. 60

185. A method, comprising:

rendering a browser interface on a display of a device via which a user is enabled to request access to a Web page comprising HTML-based Web content defining an original page layout, functionality, and design of content on the Web page; 65

38

in response to a user request of the Web page via the browser interface,

retrieving the Web page via the device;

rendering the Web page via the device such that a full width of the Web page is rendered on the display; and

re-rendering the Web page in response to associated user inputs to the hand-held device to enable the Web page to be browsed at various zoom levels and panned views while preserving the original page layout, functionality, and design of the Web page content at each zoom level and panned view of the Web page,

wherein the method enables a user of the device to browse, zoom, and pan billions of Web pages in a manner that preserves the original layout, functionality, and design of the HTML-based Web page content of each Web page.

186. The method of claim 185, further comprising enabling a user to zoom on a column of the Web page via a corresponding user input, wherein in response thereto, the display is re-rendered such that content corresponding to the selected column is enlarged.

187. The method of claim 186, wherein the corresponding user input comprises tapping on the column via the display.

188. The method of claim 186, wherein the content of the column is reformatted to fit characteristics of the display when the display is re-rendered.

189. The method of claim 186, wherein the display is re-rendered such that content corresponding to the selected column is rendered to fit across the display.

190. The method of claim 185, wherein the Web page includes at least one image, the method further comprising enabling a user to zoom on an image via a corresponding user input, wherein in response thereto, the display is re-rendered such that the image is enlarged. 35

191. The method of claim 190, wherein the corresponding user input comprises tapping on the image via the display.

192. The method of claim 190, wherein the display is re-rendered such that the image is rendered to fit across the display.

193. The method of claim 185, further comprising enabling a user to zoom on a paragraph of the Web page via a corresponding user input, wherein in response thereto, the display is re-rendered such that content corresponding to the selected paragraph is enlarged. 40

194. The method of claim 193, wherein the corresponding user input comprises tapping on the paragraph via the display.

195. The method of claim 193, wherein the content of the paragraph is reformatted to fit characteristics of the display when re-rendered.

196. The method of claim 193, wherein the display is re-rendered such that content corresponding to the selected paragraph is rendered to fit across the display.

197. The method of claim 185, further comprising enabling a user to pan a view of the web page while in a zoomed state under which a portion of the web page is displayed in response to a user input made via the display. 55

198. The method of claim 185, further comprising returning the display of the Web page to a previous view in response to a corresponding user input.

199. The method of claim 185, wherein the display of the Web page is re-rendered in real-time to effect zooming operations. 60

200. The method of claim 185, further comprising enabling a user to pan a view of the Web page in response to a corresponding user input made via the display.

201. The method of claim 185, further comprising enabling the view of the Web page to be panned in real-time.

39

202. The method of claim 185, wherein the Web page includes text, layout attributes, and images, the method further comprising:

- receiving content corresponding to the text and layout attributes via a first connection; and
- receiving content corresponding to at least one image via a second connection.

203. The method of claim 185, wherein a portion of the HTML-based Web content comprises XML code.

204. The method of claim 185, wherein a portion of the HTML-based Web content comprises cascaded style sheet data defining aspects of the Web page design that are preserved at each zoom level and panned view.

205. The method of claim 185, wherein the Web page is retrieved via a wireless connection to one of a mobile service provider network, local area network, or wide area network.

206. The method of claim 185, wherein the device comprises a mobile phone.

207. The method of claim 185, wherein the device comprises a hand-held device.

208. The method of claim 185, wherein the device comprises one of a desktop computer, notebook computer or laptop computer.

209. A method, comprising:

- rendering a browser interface on a display via which a user of a device is enabled to request access to a Web page, the Web page comprising HTML-based Web content having an original format including HTML code defining an original page layout, functionality, and design of corresponding content on the Web page;

retrieving the Web page, via the device, and translating at least a portion of the HTML-based Web content into scalable content that supports a scalable resolution-independent representation of the Web page that preserves the original page layout, functionality and design of the content defined by its original format when scaled and rendered; and

employing the scalable content to render the Web page on the display using a first scale factor; and

enabling the Web page to be displayed at a different resolution by scaling the scalable content using a second scale factor to re-render the display,

wherein the original page layout, functionality, and design of the Web page content are preserved under both the first and second scale factors.

210. The method of claim 209, wherein the display is re-rendered in real-time.

211. The method of claim 209, wherein the device comprises a hand-held device.

212. The method of claim 209, wherein the device comprises one of a desktop computer, notebook computer or laptop computer.

213. The method of claim 209, further comprising enabling a user to pan a view of the Web page in response to a corresponding user input.

214. The method of claim 213, further comprising enabling the view of the Web page to be panned in real-time.

215. The method of claim 209, further comprising enabling a user to browse, zoom, and pan billions of Web pages in a manner that preserves the original layout, functionality, and design of the HTML-based Web page content of each Web page.

216. A machine-readable medium having a plurality of instructions tangibly stored thereon, which when executed enable a device to perform operations comprising:

- rendering a browser interface via which a user is enabled to request access to a Web page hosted by an Internet Web

40

site, the Web page comprising HTML-based Web content having an original format defining an original width and height of the Web page and an original page layout, functionality, and design of content on the Web page;

- retrieving the Web page via the wireless communication means, and translating at least a portion of the HTML-based Web content from its original format into scalable content that supports a scalable resolution-independent representation of the Web page that preserves the original page layout, functionality and design of the content defined by its original format when scaled and rendered; and

scaling the scalable content to render the Web page on the display such that a width of the Web page is rendered to fit across the display.

217. The machine-readable medium of claim 216, wherein execution of the instructions performs further operations comprising enabling the user to zoom in on a user-selectable portion of a display of the Web page in response to a corresponding user interface input.

218. The machine-readable medium of claim 217, wherein the display of the Web page is re-rendered in real-time to effect zooming operations.

219. The machine-readable medium of claim 216, wherein the Web page includes at least one hyperlink, and wherein execution of the instructions performs further operations comprising:

- enabling the user to select the hyperlink; and, in response thereto,
- retrieving and translating Web content associated with the hyperlink to produce additional scalable content; and
- employing the additional scalable content to render the Web content associated with the hyperlink on the display.

220. The machine-readable medium of claim 216, wherein execution of the instructions performs further operations comprising:

- parsing markup language code to determine the original page layout of display content within the Web page, wherein the original page layout defines a layout location for a plurality of objects, including at least one of text objects, graphic layout objects, or graphic image objects included in the Web page;
- defining a primary datum corresponding to the original page layout; and,
- for each object,
- defining an object datum corresponding to the layout location for the object;
- generating a vector from the primary datum to the object datum for the object; and
- creating a reference that links the object to the vector that is generated.

221. The machine-readable medium of claim 216, wherein execution of the instructions performs further operations comprising enabling the Web page to be displayed at different resolutions by scaling the scalable content to re-render the display in response to associated user inputs,

wherein the original page layout, functionality, and design of the Web page content are preserved at each of the different resolutions.

222. The machine-readable medium of claim 216, wherein execution of the instructions performs further operations comprising returning the display of the Web page to a previous view in response to a corresponding user input.

223. The machine-readable medium of claim 216, wherein execution of the instructions performs further operations

41

comprising enabling a user to pan a view of the Web page in response to a corresponding user input.

224. The machine-readable medium of claim 223, wherein execution of the instructions performs further operations comprising enabling the view of the Web page to be panned in real-time.

225. The machine-readable medium of claim 216, wherein the page layout of the Web page is defined to have an original aspect ratio, and wherein the scalable content is scaled when rendered so as to produce a display of the Web page having a different aspect ratio.

226. The machine-readable medium of claim 216, wherein execution of the instructions performs further operations comprising enabling a user to zoom on a column of the Web page via a corresponding user input, wherein in response thereto, the display is re-rendered such that content corresponding to the selected column is enlarged.

227. The machine-readable medium of claim 226, wherein the corresponding user input comprises tapping on the column via the display.

228. The machine-readable medium of claim 226, wherein the display is re-rendered such that content corresponding to the selected column is rendered to fit across the display.

229. The machine-readable medium of claim 226, wherein the content of the column is reformatted to fit characteristics of the display when the display is re-rendered.

230. The machine-readable medium of claim 216, wherein the Web content includes at least one image, and wherein execution of the instructions performs further operations comprising enabling a user to zoom on an image via a corresponding user input, wherein in response thereto, the display is re-rendered such that the image is enlarged.

231. The machine-readable medium of claim 230, wherein the corresponding user input comprises tapping on the image via the display.

232. The machine-readable medium of claim 230, wherein the display is re-rendered such that the image is rendered to fit across the display.

233. The machine-readable medium of claim 216, wherein execution of the instructions performs further operations comprising enabling a user to zoom on a paragraph of the Web content via a corresponding user input, wherein in response thereto, the display is re-rendered such that content corresponding to the selected paragraph is enlarged.

234. The machine-readable medium of claim 233, wherein the corresponding user input comprises tapping on the paragraph via the display.

235. The machine-readable medium of claim 233, wherein the content of the paragraph is reformatted to fit characteristics of the display when the display is re-rendered.

236. The machine-readable medium of claim 233, wherein the display is re-rendered such that content corresponding to the selected paragraph is rendered to fit across the display.

237. The machine-readable medium of claim 216, wherein the Web page includes text, layout attributes, and images, and wherein execution of the instructions performs further operations comprising:

- receiving content corresponding to the text and layout attributes via a first connection; and
- receiving content corresponding to at least one image via a second connection.

238. The machine-readable medium of claim 216, wherein execution of the instructions performs further operations comprising:

- generating a vector-based display list associated with the scalable content; and

42

employing the display list to re-render the display at different scale factors to zoom the Web page.

239. The machine-readable medium of claim 216, wherein execution of the instructions performs further operations comprising:

- parsing markup language code corresponding to the retrieved Web page to determine the original page layout of the content on the Web page;
- logically grouping selected content into objects;
- defining a primary datum corresponding to the original page layout; and,
- for each object,
 - defining an object datum corresponding to a layout location datum for the object's associated display content;
 - generating a vector from the primary datum to the object datum for the object; and
 - creating a reference that links the object to the vector that is generated.

240. The machine-readable medium of claim 239, wherein execution of the instructions performs further operations comprising:

- generating a bounding box for each object, the bounding box representing a portion of a rendered display page occupied by the object's associated group of content.

241. The machine-readable medium of claim 240, wherein execution of the instructions performs further operations comprising:

- mapping the object vectors and associated bounding boxes to a virtual display in memory.

242. The machine-readable medium of claim 241, wherein execution of the instructions performs further operations comprising:

- enabling a user to view the Web page at a user-selectable zoom level and panned view by,
- determining a first scale factor and offset in response to one or more corresponding user inputs defining a user-selectable zoom level and panned view corresponding to a rendered display of the Web page desired by a user; and
- determining a virtual display bounding box for the virtual display associated with the first scale factor and offset;
- identifying object bounding boxes having at least a portion falling within the virtual display bounding box; and,
- for each of such object bounding boxes,
 - retrieving content associated with that object bounding box;
 - applying an appropriate scale factor to the content associated with that object bounding box to produce scaled content; and
 - rendering the portion of scaled content within the virtual display bounding box to render the content on the display.

243. The machine-readable medium of claim 216, wherein the scalable content includes scalable text content, and wherein execution of the instructions performs further operations comprising scaling a scalable font to render the scalable text content.

244. The machine-readable medium of claim 216, wherein at least a portion of the instructions comprise Java-based instructions.

245. The machine-readable medium of claim 216, wherein the device comprises a mobile phone.

246. The machine-readable medium of claim 216, wherein the device comprises a hand-held device.

43

247. The machine-readable medium of claim 216, wherein the Web page is accessed via a mobile service provider network.

248. The machine-readable medium of claim 216, wherein a portion of the scalable content comprises vector-based content.

249. The machine-readable medium of claim 216, wherein the device comprises one of a desktop computer, notebook computer or laptop computer.

250. The machine-readable medium of claim 216, wherein the instructions are embodied as a Web browser.

251. The machine-readable medium of claim 216, wherein execution of the instructions enables a user to browse, zoom, and pan billions of Web pages in a manner that preserves the original layout, functionality, and design of the HTML-based Web page content of each Web page.

252. A machine-readable medium having a plurality of instructions comprising a Web browser tangibly stored thereon, which when executed enable a device to perform operations comprising:

rendering a browser interface on a display associated with the device via which a user is enabled to request access to a Web page comprising HTML-based Web content defining an original page layout, functionality, and design of content on the Web page;

retrieving the Web page and processing HTML-based Web content to produce scalable content; and

employing at least one of the scalable content or data derived therefrom to,

render the Web page on the display; and

re-render the display in response to associated user inputs to enable the Web page to be browsed at various zoom levels and panned views while preserving the original page layout, functionality, and design of the Web page content at each zoom level and panned view of the Web page,

wherein the Web browser enables a user of the device to browse, zoom, and pan billions of Web pages in a manner that preserves the original layout, functionality, and design of the HTML-based Web page content of each Web page at each zoom level and panned view.

253. The machine-readable medium of claim 252, wherein the device comprises a mobile phone.

254. The machine-readable medium of claim 252, wherein the device comprises a hand-held device.

255. The machine-readable medium of claim 252, wherein execution of the instructions performs further operations comprising enabling the user to zoom in on a user-selectable portion of a display of the Web page in response to a user interface input.

256. The machine-readable medium of claim 255, wherein the user interface input enables the user to define an area of a current view of the Web page on which to zoom in on.

257. The machine-readable medium of claim 252, wherein the display of the Web page is re-rendered in real-time to effect zooming operations.

258. The machine-readable medium of claim 252, wherein the Web page includes at least one hyperlink, and wherein execution of the instructions performs further operations comprising:

enabling the user to select the hyperlink; and, in response thereto,

retrieving and processing HTML-based Web content associated with the hyperlink to produce additional scalable content; and

44

employing at least one of the additional scalable content or data derived therefrom to render the Web content associated with the hyperlink on the display.

259. The machine-readable medium of claim 252, wherein at least a portion of the scalable content comprises scalable vector-based content.

260. The machine-readable medium of claim 252, wherein execution of the instructions performs further operations comprising returning the display of the Web page to a previous view in response to a corresponding user input made via the display.

261. The machine-readable medium of claim 252, wherein execution of the instructions performs further operations comprising enabling a user to pan a view of the Web page in response to a corresponding user input made via the display.

262. The machine-readable medium of claim 252, wherein execution of the instructions performs further operations comprising enabling the view of the Web page to be panned in real-time.

263. The machine-readable medium of claim 252, wherein the page layout of the Web page is defined to have an original aspect ratio, and wherein said at least one of scalable content or data derived therefrom is scaled to render a display of the Web page having a different aspect ratio.

264. The machine-readable medium of claim 252, wherein execution of the instructions performs further operations comprising enabling a user to zoom on a column of the Web content via a corresponding user input, wherein in response thereto, the display is re-rendered such that content corresponding to the selected column is rendered to fit across the display.

265. The machine-readable medium of claim 264, wherein the corresponding user input comprises tapping on the column via the display.

266. The machine-readable medium of claim 264, wherein the content of the column is reformatted to fit characteristics of the display when the display is re-rendered.

267. The machine-readable medium of claim 252, wherein the Web content includes at least one image, and wherein execution of the instructions performs further operations comprising enabling a user to zoom on an image via a corresponding user input, wherein in response thereto, the display is re-rendered such that the image is rendered to fit across the display.

268. The machine-readable medium of claim 267, wherein the corresponding user input comprises tapping on the image via the display.

269. The machine-readable medium of claim 252, wherein execution of the instructions performs further operations comprising enabling a user to zoom on a paragraph of the Web page via a corresponding user input, wherein in response thereto, the display is re-rendered such that content corresponding to the selected paragraph is rendered to fit across a browser display area of the display.

270. The machine-readable medium of claim 269, wherein the corresponding user input comprises tapping on the paragraph via the display.

271. The machine-readable medium of claim 269, wherein the content of the paragraph is reformatted to fit characteristics of the display area when the display is re-rendered.

272. The machine-readable medium of claim 252, wherein the Web page includes text, layout attributes, and images, and wherein execution of the instructions performs further operations comprising:

receiving content corresponding to the text and layout attributes via a first connection; and

45

receiving content corresponding to at least one image via a second connection.

273. The machine-readable medium of claim 252, wherein the device includes dynamic memory having at least a portion employed for rendering purposes, and wherein execution of the instructions performs further operations comprising:

building a display list via use of the scalable content and rendering display list content on a virtual display in the dynamic memory; and
scaling the display list content to re-render the display of the Web page.

274. The machine-readable medium of claim 252, wherein execution of the instructions performs further operations comprising:

parsing HTML-based code corresponding to the retrieved Web page to determine the original page layout of the content on the Web page;

logically grouping selected content into objects;

defining a primary datum corresponding to the original page layout; and,

for each object,

defining an object datum corresponding to a layout location datum for the object's associated display content;
generating a vector from the primary datum to the object datum for the object; and
creating a reference that links the object to the vector that is generated.

275. The machine-readable medium of claim 274, wherein execution of the instructions performs further operations comprising:

generating a bounding box for each object, the bounding box representing a portion of a rendered display page occupied by the object's associated group of content.

276. The machine-readable medium of claim 275, wherein the device includes dynamic memory having at least a portion employed for rendering purposes, and wherein execution of the instructions performs further operations comprising:

mapping the object vectors and associated bounding boxes to a virtual display in the dynamic memory.

277. The machine-readable medium of claim 276, wherein execution of the instructions performs further operations comprising:

enabling a user to view the Web page at a user-selectable zoom level and panned view by,

determining a first scale factor and offset in response to one or more corresponding user inputs defining a user-selectable zoom level and panned view corresponding to a rendered display of the Web page desired by a user;

determining a virtual display bounding box for the virtual display associated with the first scale factor and offset;

identifying object bounding boxes having at least a portion falling within the virtual display bounding box; and,

for each of such object bounding boxes,

retrieving content associated with that object bounding box;

applying an appropriate scale factor to the content associated with that object bounding box to produce scaled content; and

rendering the portion of scaled content within the virtual display bounding box to render the content on the display.

278. The machine-readable medium of claim 252, wherein the scalable content includes scalable text content, and

46

wherein execution of the instructions performs further operations comprising scaling a scalable font to render the scalable text content.

279. The machine-readable medium of claim 252, wherein the original format of the Web page defines a height and width for the Web page, and wherein execution of the instructions performs further operations comprising:

determining an applicable scale factor to display at least one of the width and height of the Web page across a browser display area of the display; and
employing the scale factor to render the browser display area.

280. The machine-readable medium of claim 252, wherein at least a portion of the instructions comprise Java-based instructions.

281. The machine-readable medium of claim 252, wherein a portion of the HTML-based Web content comprises XML code.

282. The machine-readable medium of claim 252, wherein a portion of the HTML-based Web content comprises cascaded style sheet data defining aspects of the Web page design that are preserved at each zoom level and panned view.

283. The machine-readable medium of claim 252, wherein a portion of the scalable content comprises vector-based content.

284. The machine-readable medium of claim 252, wherein the device comprises one of a desktop computer, notebook computer or laptop computer.

285. A machine-readable medium having a plurality of instructions tangibly stored thereon, which when executed enable a wireless device to perform operations comprising:

rendering a browser interface on a display of the wireless device via which a user is enabled to request access to a Web page comprising HTML-based Web content defining an original page layout, functionality, and design of content on the Web page;

in response to a user request of the Web page,

retrieving the Web page via the wireless device;

rendering the Web page such that a width of the Web page is rendered to fit across the display; and

re-rendering the Web page in response to associated user inputs to enable the Web page to be browsed at various zoom levels and panned views while preserving the original page layout, functionality, and design of the Web page content at each zoom level and panned view.

286. The machine-readable medium of claim 285, wherein execution of the instructions performs further operations comprising enabling a user to zoom on a column of the Web page via a corresponding user input, wherein in response thereto, the display is re-rendered such that content corresponding to the selected column is enlarged.

287. The machine-readable medium of claim 286, wherein the corresponding user input comprises tapping on the column via the display.

288. The machine-readable medium of claim 286, wherein the content of the column is reformatted to fit characteristics of the display when the display is re-rendered.

289. The machine-readable medium of claim 286, wherein the display is re-rendered such that content corresponding to the selected column is rendered to fit across the display.

290. The machine-readable medium of claim 285, wherein the Web page includes at least one image, and wherein execution of the instructions performs further operations comprising enabling a user to zoom on an image via a corresponding user input, wherein in response thereto, the display is re-rendered such that the image is enlarged.

47

291. The machine-readable medium of claim 290, wherein the corresponding user input comprises tapping on the image via the display.

292. The machine-readable medium of claim 290, wherein the display is re-rendered such that the image is rendered to fit across the display.

293. The machine-readable medium of claim 285, wherein execution of the instructions performs further operations comprising enabling a user to zoom on a paragraph of the Web page via a corresponding user input, wherein in response thereto, the display is re-rendered such that content corresponding to the selected paragraph is enlarged.

294. The machine-readable medium of claim 293, wherein the corresponding user input comprises tapping on the paragraph via the display.

295. The machine-readable medium of claim 293, wherein the content of the paragraph is reformatted to fit characteristics of the display when re-rendered.

296. The machine-readable medium of claim 293, wherein the display is re-rendered such that content corresponding to the selected paragraph is rendered to fit across the display.

297. The machine-readable medium of claim 285, wherein execution of the instructions performs further operations comprising enabling a user to pan a display of the web page while in a zoomed state under which a portion of the web page is displayed.

298. The machine-readable medium of claim 285, wherein execution of the instructions performs further operations comprising returning the display of the Web page to a previous view in response to a corresponding user input.

299. The machine-readable medium of claim 285, wherein the display of the Web page is re-rendered in real-time to effect zooming operations.

300. The machine-readable medium of claim 285, wherein execution of the instructions performs further operations comprising enabling a user to pan a view of the Web page in response to a corresponding user input made via the display.

301. The machine-readable medium of claim 285, wherein execution of the instructions performs further operations comprising enabling the view of the Web page to be panned in real-time.

302. The machine-readable medium of claim 285, wherein the Web page includes text, layout attributes, and images, and wherein execution of the instructions performs further operations comprising:

- receiving content corresponding to the text and layout attributes via a first connection; and
- receiving content corresponding to at least one image via a second connection.

303. The machine-readable medium of claim 285, wherein a portion of the HTML-based Web content comprises XML code.

304. The machine-readable medium of claim 285, wherein a portion of the HTML-based Web content comprises cascaded style sheet data defining aspects of the Web page design that are preserved at each zoom level and panned view.

305. The machine-readable medium of claim 285, wherein the wireless device is configured to connect to a mobile service provider network and retrieve the Web page via the mobile service provider network.

306. The machine-readable medium of claim 285, wherein the wireless device comprises a mobile phone.

307. The machine-readable medium of claim 285, wherein the wireless device comprises a hand-held device.

308. The machine-readable medium of claim 285, wherein the wireless device comprises one of a notebook computer or laptop computer.

48

309. The machine-readable medium of claim 285, wherein the instructions are embodied as a Web browser.

310. The machine-readable medium of claim 285, wherein execution of the instructions enables a user to browse, zoom, and pan billions of Web pages in a manner that preserves the original layout, functionality, and design of the HTML-based Web page content of each Web page.

311. A machine-readable medium having a plurality of instructions comprising a Web browser stored thereon, which when executed enable a device to perform operations comprising:

- launching a Web browser including a browser interface via which a user is enabled to request access to a Web page, the Web page comprising HTML-based Web content having an original format including HTML code defining an original page layout, functionality, and design of corresponding content on the Web page;

- retrieving, and translating at least a portion of the HTML-based Web content into scalable content that supports a scalable resolution-independent representation of the Web page that preserves the original page layout, functionality and design of the content defined by its original format when scaled and rendered; and

- employing the scalable content to render the Web page on the Web browser using a first scale factor; and

- enabling the Web page to be displayed at a different resolution by scaling the scalable content using a second scale factor to re-render the Web page on the Web browser,

- wherein the original page layout, functionality, and design of the Web page content are preserved under both the first and second scale factors, and

- wherein the Web browser enables a user of the device to browse billions of Web pages at multiple scale factors in a manner that preserves the original layout, functionality, and design of the HTML-based Web page content of each Web page at each scale factor.

312. The machine-readable medium of claim 311, wherein the display is re-rendered in real-time.

313. The machine-readable medium of claim 311, wherein the Web browser is configured to be installed on a hand-held device.

314. The machine-readable medium of claim 311, wherein the Web browser is configured to be installed on at least one of a desktop computer, notebook computer or laptop computer.

315. The machine-readable medium of claim 311, wherein execution of the instructions performs further operations comprising enabling a user to pan a view of the Web page in response to a corresponding user input.

316. The machine-readable medium of claim 315, wherein execution of the instructions performs further operations comprising enabling the view of the Web page to be panned in real-time.

317. A hand-held wireless device, comprising:

- a processor,
- a wireless communications interface, to facilitate wireless communication with a network that supports access to the Internet;
- a display; and

- non-volatile memory, operatively coupled to the processor, in which software comprising a browser is stored, the browser comprising a plurality of instructions that when executed by the processor enable the device to perform operations including,

- rendering a browser interface on the display via which a user is enabled to request access to a Web page includ-

49

ing at least one image, at least one column, at least one
 hyperlink to an external reference and having a width
 and height;
 retrieving the Web page via the wireless communica-
 tions interface;
 rendering the Web page on the display such that at least
 one of the width and height of the Web page is fully
 displayed; and
 enabling the user to,
 zoom and pan a view of the Web page;
 activate a currently displayed hyperlink to an external
 reference while at a given zoom level and panned
 view, wherein in response to an activation of a
 hyperlink to an external reference, Web content
 associated with the external reference is retrieved
 and rendered on the display;
 zoom in on an image of the Web page by tapping on
 the image via the display;

50

zoom in on a column of the Web page by tapping on
 the column via the display; and
 zoom out to a previous view of the Web page.

5 **318.** The hand-held wireless device of claim 317, wherein
 the Web page comprises HTML-based Web page content
 defining an original page layout, functionality, and design of
 the Web page content, and wherein the browser renders the
 Web page such that the original page layout, functionality,
 and design of the Web page are preserved at any selectable
 zoom level.

10 **319.** The hand-held wireless device of claim 318, wherein
 the user is enabled to browse, zoom, and pan billions of Web
 pages in a manner that preserves the original layout, function-
 ality, and design of the HTML-based Web page content of
 each Web page.

* * * * *

EXHIBIT 2

U.S. Patent No. 7,831,926



US007831926B2

(12) **United States Patent**
Rohrbaugh et al.

(10) **Patent No.:** **US 7,831,926 B2**
(45) **Date of Patent:** ***Nov. 9, 2010**

(54) **SCALABLE DISPLAY OF INTERNET CONTENT ON MOBILE DEVICES**

5,897,644 A 4/1999 Blumberg
5,952,994 A 9/1999 Ong et al.
5,956,025 A 9/1999 Goulden et al.
5,966,135 A 10/1999 Roy et al.
6,011,905 A 1/2000 Huttenlocher et al.

(75) Inventors: **Gary B. Rohrbaugh**, Bellingham, WA (US); **Scott A. Sherman**, Seattle, WA (US)

(73) Assignee: **SoftView LLC**, Bellingham, WA (US)

(Continued)

(*) Notice: Subject to any disclaimer, the term of this patent is extended or adjusted under 35 U.S.C. 154(b) by 799 days.

FOREIGN PATENT DOCUMENTS

GB 0009129.8 4/2000
GB 2344197 A 5/2000
JP 10-334082 A 12/1998

This patent is subject to a terminal disclaimer.

OTHER PUBLICATIONS

(21) Appl. No.: **11/738,486**

Scalable Vector Graphics (SVG) Specification W3C Working Draft Feb. 11, 1999 <http://www.w3.org/TR/1999/WD-SVG-19990211/>—Pub Feb. 11, 1999 by W3C pp. 1-7.

(22) Filed: **Apr. 21, 2007**

Fisher, B., G. Agelidis, J. Dill, P. Tan, G. Collaud and C. Jones. "CZWeb: Fish-Eye Views for Visualizing the World-Wide Web", Proc. Seventh Int. Conf. on Human-Computer Interaction (HCI International '97), pp. 719-722, 1997.

(65) **Prior Publication Data**

US 2007/0288841 A1 Dec. 13, 2007

Related U.S. Application Data

(Continued)

(63) Continuation of application No. 09/878,097, filed on Jun. 8, 2001, now Pat. No. 7,210,099, which is a continuation-in-part of application No. 09/828,511, filed on Apr. 7, 2001, now abandoned.

Primary Examiner—Doug Hutton
Assistant Examiner—Quoc A Tran

(74) *Attorney, Agent, or Firm*—Law Office of R. Alan Burnett

(60) Provisional application No. 60/211,019, filed on Jun. 12, 2000, provisional application No. 60/217,345, filed on Jul. 11, 2000.

(57) **ABSTRACT**

Mobile devices enabled to support resolution-independent scalable display of Internet (Web) content to allow Web pages to be scaled (zoomed) and panned for better viewing on smaller screen sizes. The mobile devices employ software-based processing of original Web content, including HTML-based content, XML, cascade style sheets, etc. to generate scalable content. The scalable content and/or data derived therefrom are then employed to enable the Web content to be rapidly rendered, zoomed, and panned. Display lists may also be employed to provide further enhancements in rendering speed.

(51) **Int. Cl.**
G06F 17/00 (2006.01)

(52) **U.S. Cl.** **715/800**; 715/234; 715/243; 715/853

(58) **Field of Classification Search** 715/800, 715/238, 249, 204, 234

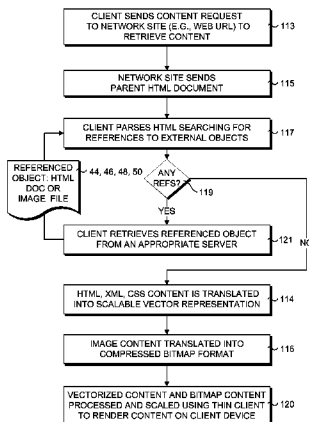
See application file for complete search history.

(56) **References Cited**

U.S. PATENT DOCUMENTS

5,615,384 A 3/1997 Allard et al.

88 Claims, 22 Drawing Sheets



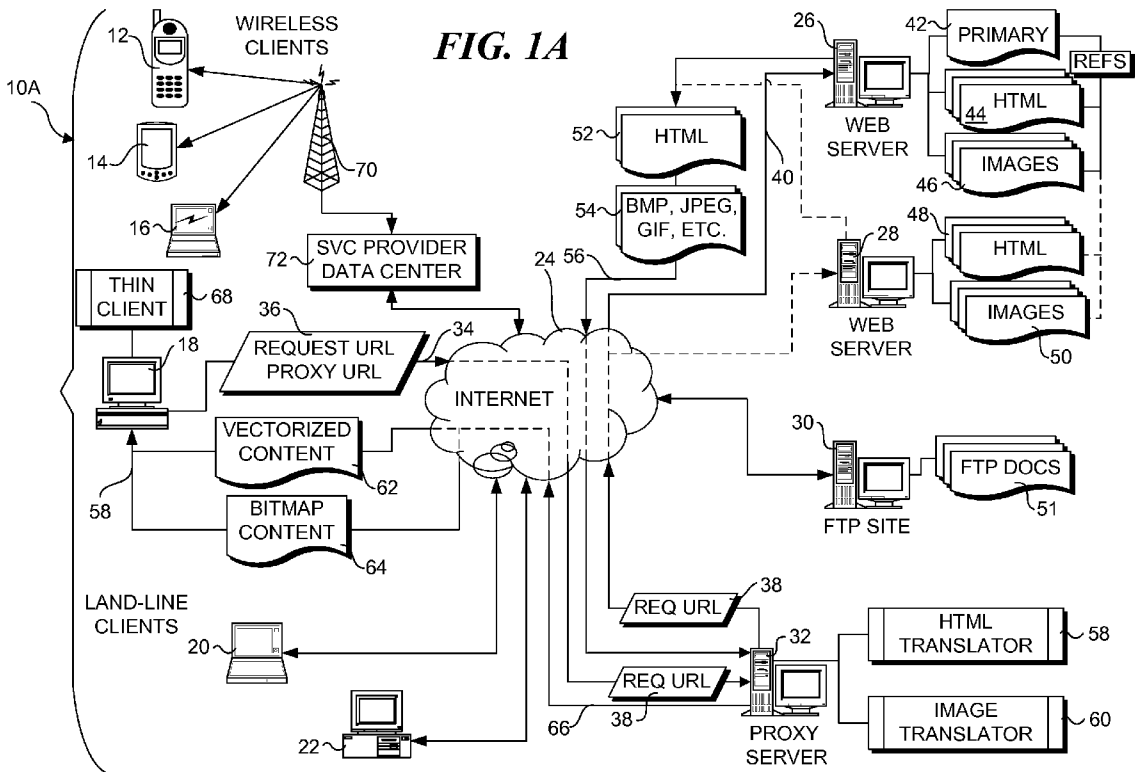
U.S. PATENT DOCUMENTS

6,057,854 A 5/2000 Davis, Jr. et al.
 6,076,166 A 6/2000 Moshfeghi et al.
 6,211,856 B1 4/2001 Choi et al.
 6,300,947 B1 10/2001 Kanevsky
 6,337,693 B1 1/2002 Roy et al.
 6,421,733 B1 7/2002 Tso et al.
 6,449,639 B1 9/2002 Blumberg
 6,456,305 B1 9/2002 Qureshi et al.
 6,456,308 B1 9/2002 Agranet et al.
 6,457,030 B1 9/2002 Adams et al.
 6,466,203 B2 10/2002 Van Ee
 6,546,397 B1 4/2003 Rempell
 6,553,410 B2 4/2003 Kikinis
 6,556,217 B1 4/2003 Mäkipää
 6,615,212 B1 9/2003 Dutta et al.
 6,674,445 B1 1/2004 Chithambaram et al.
 6,704,024 B2 3/2004 Robotham et al.
 6,781,600 B2 8/2004 Anwar
 6,829,746 B1 12/2004 Schwerdtfeger et al.
 6,857,102 B1 2/2005 Bickmore et al.
 6,886,034 B2 4/2005 Blumberg
 6,925,595 B1 8/2005 Whitledge et al.
 6,978,418 B1 12/2005 Bain et al.
 6,996,533 B2 2/2006 Ikeda et al.
 7,023,572 B2 4/2006 Tuli
 7,055,095 B1 5/2006 Anwar
 7,072,984 B1 7/2006 Polonsky et al.
 7,219,309 B2 5/2007 Kaasila et al.
 7,308,649 B2 12/2007 Ehrlich et al.
 7,450,114 B2 11/2008 Anwar
 2001/0015719 A1 8/2001 Van Ee et al.
 2001/0047428 A1 11/2001 Hunter
 2002/0023110 A1 2/2002 Fortin et al.
 2002/0030699 A1 3/2002 Van Ee
 2002/0112237 A1 8/2002 Ketts
 2002/0158908 A1 10/2002 Vaajala et al.
 2004/0049598 A1 3/2004 Tucker et al.
 2005/0144256 A1 6/2005 Blumberg

OTHER PUBLICATIONS

Benjamin B. Bederson et al., Pad++: A Zoomable Graphical Sketchpad For Exploring Alternate Interface Physics, Sep. 19, 1995, pp. 1-30, <http://www.cs.unm.edu/pad++>.
 Benjamin B. Bederson et al., A Zooming Web Browser, SPIE 1996, <http://www.cs.umd.edu/hcil/jazz/learn/papers/spie-96-webbrowser.pdf>, entirety (pages not numbered).
 Specification for Simple Vector Format (SVF) v1.1 Jan. 16, 1995, pp. 1-9.
 Specification for Simple Vector Format (SFV) v2.0 Dec. 6, 2000, <http://www.svf.org/spec.html>, pp. 1-20.
 SVF XML <http://www.svf.org/svfxml.html>, pp. 1-18.
 Changes to SVF (Date unknown), pp. 1-2.
 Scalable Vector Graphics (SVG) 1.0 Specification, W3C Candidate Recommendation Nov. 2, 2000, pp. 1-513.
 Introduction to SVG, part of WD-SVG-19990211, [http://www.w3.org/TR/1999/WD-SVG-19990211/intro.html#Document . . .](http://www.w3.org/TR/1999/WD-SVG-19990211/intro.html#Document...), pp. 1-3.
 W3C Scalable Vector Graphics (SVG)—History, <http://www.w3.org/Graphics/SVG/History>, pp. 1-18.
 Janus Boye, SVG Brings Fast Vector Graphics to Web, Jul. 29, 1999 <http://www.irt.org/articles/js176/>, pp. 1-5.

Vector Markup Language (VML), World Wide Web Consortium Note May 13, 1998, Note-VML-19980513, http://www.w3.org/TR/1998/Note-VML_19980513, pp. 1-47.
 Vector Markup Language, http://en.wikipedia.org/wiki/Vector_Markup_Language, pp. 1-2.
 Precision Graphics Markup Language (PGML), World Wide Web Consortium Note Apr. 10, 1998, NOTE-PGML-19980410, <http://www.w3.org/TR/1998/NOTE-PGML-19980410>, pp. 1-31.
 PCT International Search Report for International Application No. PCT/US01/40920, International Searching Authority, European Patent Office, Dec. 27, 2001, pp. 1-8.
 PCT International Preliminary Exam Report for International Application No. PCT/US01/40920, International Preliminary Examining Authority, European Patent Office, Jun. 20, 2002, pp. 1-5.
 Dennis, "Webraska Joins W3C for Next Generation Internet Graphics," Newsbytes, May 8, 2000, entirety (pages not numbered).
 Fox et al. "Experience With Top Gun Wingman: A Proxy-Based Graphical Web Browser for the 3Com PalmPilot" Proceedings of Middleware '98, 1998, entirety (pages not numbered).
 "Tadpole Technology: ILOG technology within Tadpole's Cartesia Redline system making repairs simpler," M2 Presswire, Jan. 1999, entirety (pages not numbered).
 Thryft, "Wireless link ramped for 'always on' Internet," EETimes, Mar. 23, 1999, pp. 1-4.
 Bharadvaj et al., "An Active Transcoding Proxy to Support Mobile Web Access," Proc. Of the IEEE Symposium on Reliable Distributed Systems, 1998, entirety (pages not numbered).
 Björk, et al., "WEST: A Web Browser for Small Terminals," Proceedings of the 12 ACM symposium on User Interface Software and Technology (1999), entirety (pages not numbered).
 Holmquist, "The Zoom Browser," Human IT Tridskrift för studier av IT ur ett humanvetenskapligt perspektive, Mar., 1998, pp. 1-18.
 Chapple, "Rethinking the role of an embedded Internet client in digital set-top boxes," WebTV Workshop Submission Paper, Jun. 1998, <http://www.w3.org/Architecture/1998/06/Workshop/paper11/>, pp. 1-4.
 Graham, "Mobile SVG at BitFlash Inc.," May 2001, <http://www.w3.org/Talks/2002/1007-Di-Helsinki/bitflash/index.html>, pp. 1-4.
 Koncz, "Working with Autodesk Mapguide," Nov. 2000, http://www.directionsmag.com/article.php?article_id=77&trv=1, pp. 1-6.
 Schaffer et al. "Navigating Hierarchically Cluster Networks through Fisheye and Full-Zoom Methods," Mar. 24, 1988—pp. 1-20.
 Holmquist "Flip Zooming: Focus+Context Visualization of Linearly Ordered Discrete Visual Structures," Published 2000, pp. 27-53.
 Scalable Vector Graphics (SVG) Specification, W3C Working Draft Feb. 11, 1999 WD-SVG-19990211, <http://www.w3.org/TR/1999/WD-SVG-19990211/> (HTML format—initial page).
 Scalable Vector Graphics (SVG) Specification, W3C Working Draft Apr. 12, 1999 WD-SVG-19990412, <http://www.w3.org/TR/1999/WD-SVG-19990412/> (HTML format—initial page).
 Jones et al. "From the large screen to the small screen—retaining the designer's design for effective user interaction," IEEE Colloquium on Issues for Networked Interpersonal Communicators, May 1997, pp. 3/1-3/4.
 Pacchiano, "Adobe's New View," Computer Shopper, vol. 19, No. 4, Apr. 1999, p. 346.
 Official USPTO Actions for U.S. Appl. No. 09/878,097. (Entirety—pages not numbered).
 Official USPTO Actions for U.S. Appl. No. 11/045,649. (Entirety—pages not numbered).
 Official USPTO Actions for U.S. Appl. No. 11/045,757. (Entirety—pages not numbered).
 Official USPTO Actions for U.S. Appl. No. 11/738,932. (Entirety—pages not numbered).
 Official USPTO Action for U.S. Appl. No. 11/868,124 Mailed Aug. 24, 2010 pp.1-15.



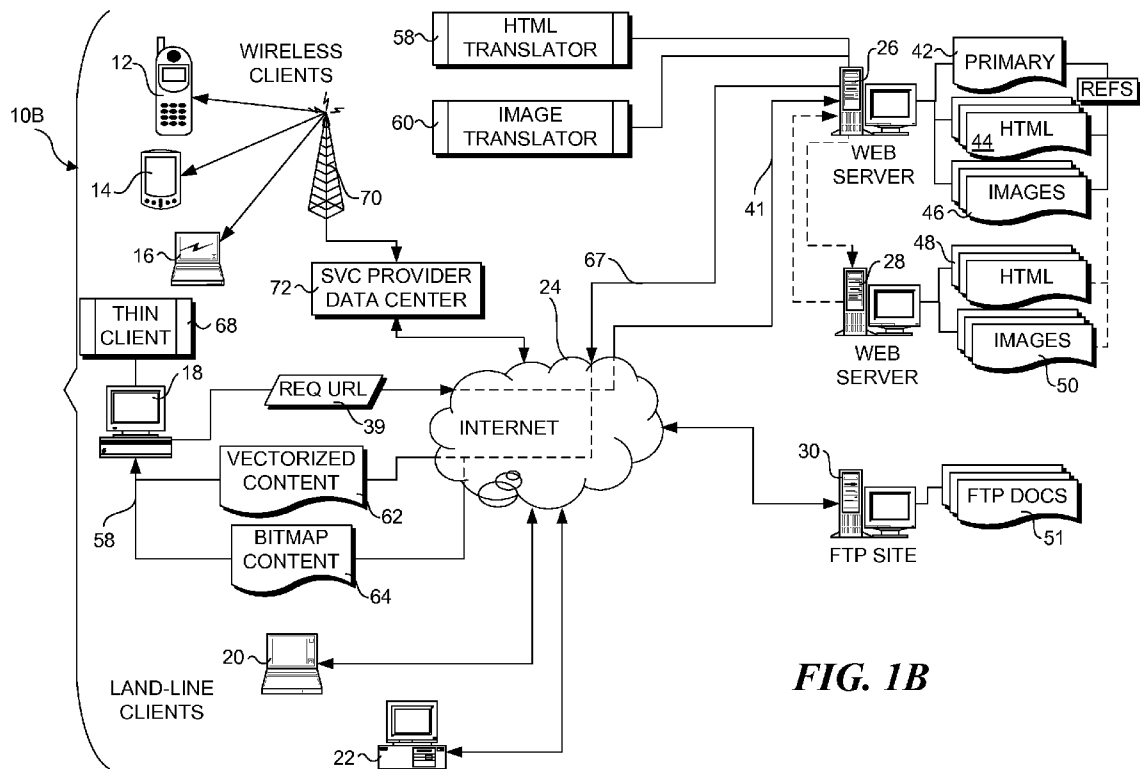


FIG. 1B

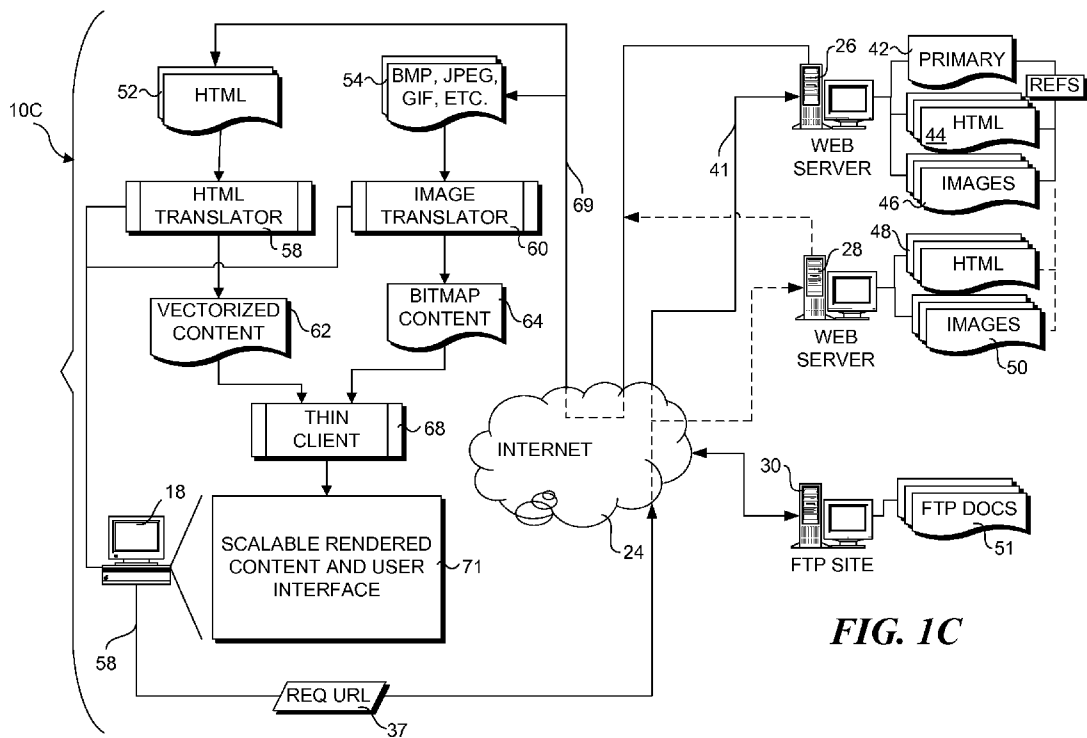


FIG. 1C

FIG. 2A

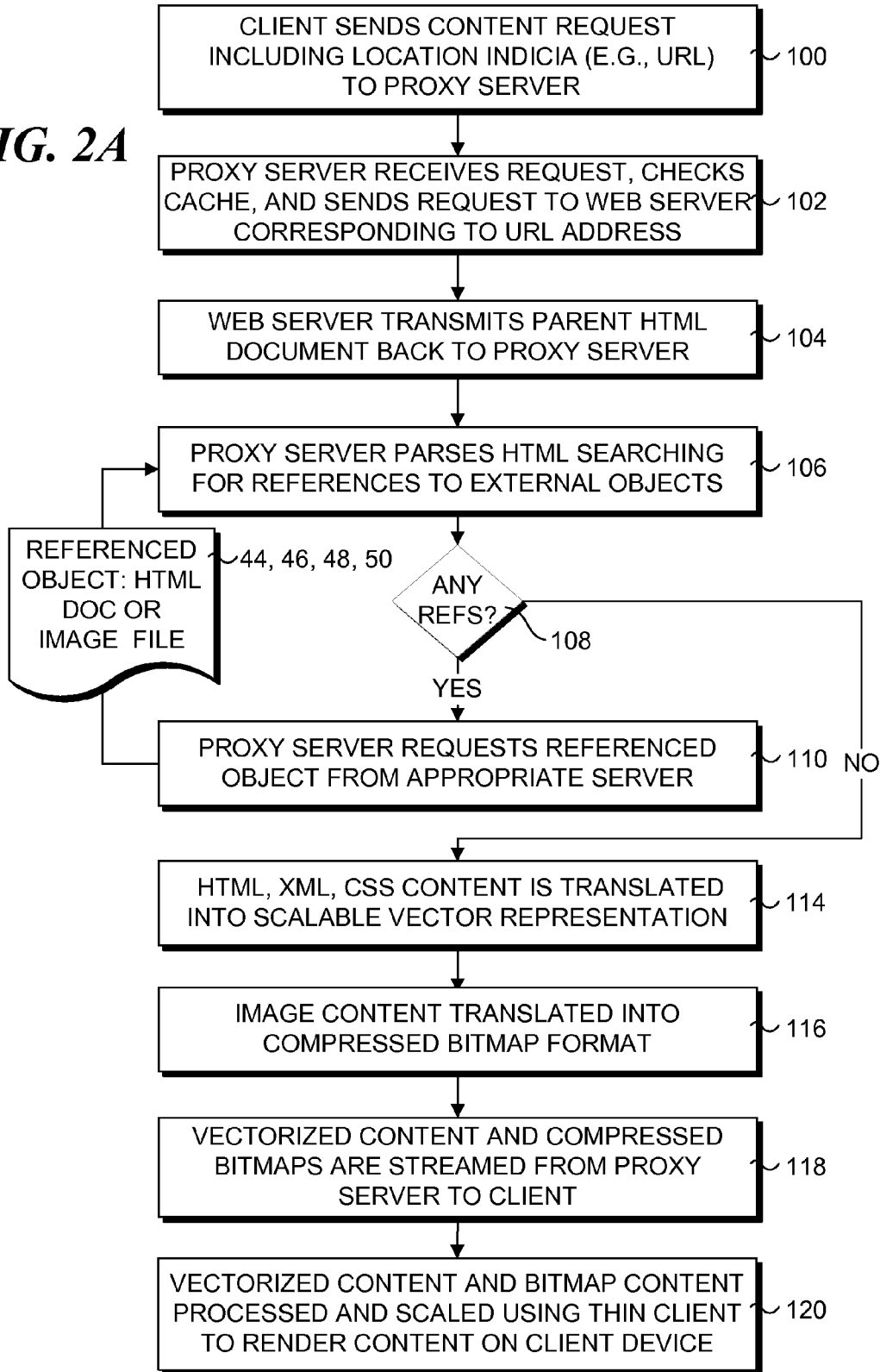


FIG. 2B

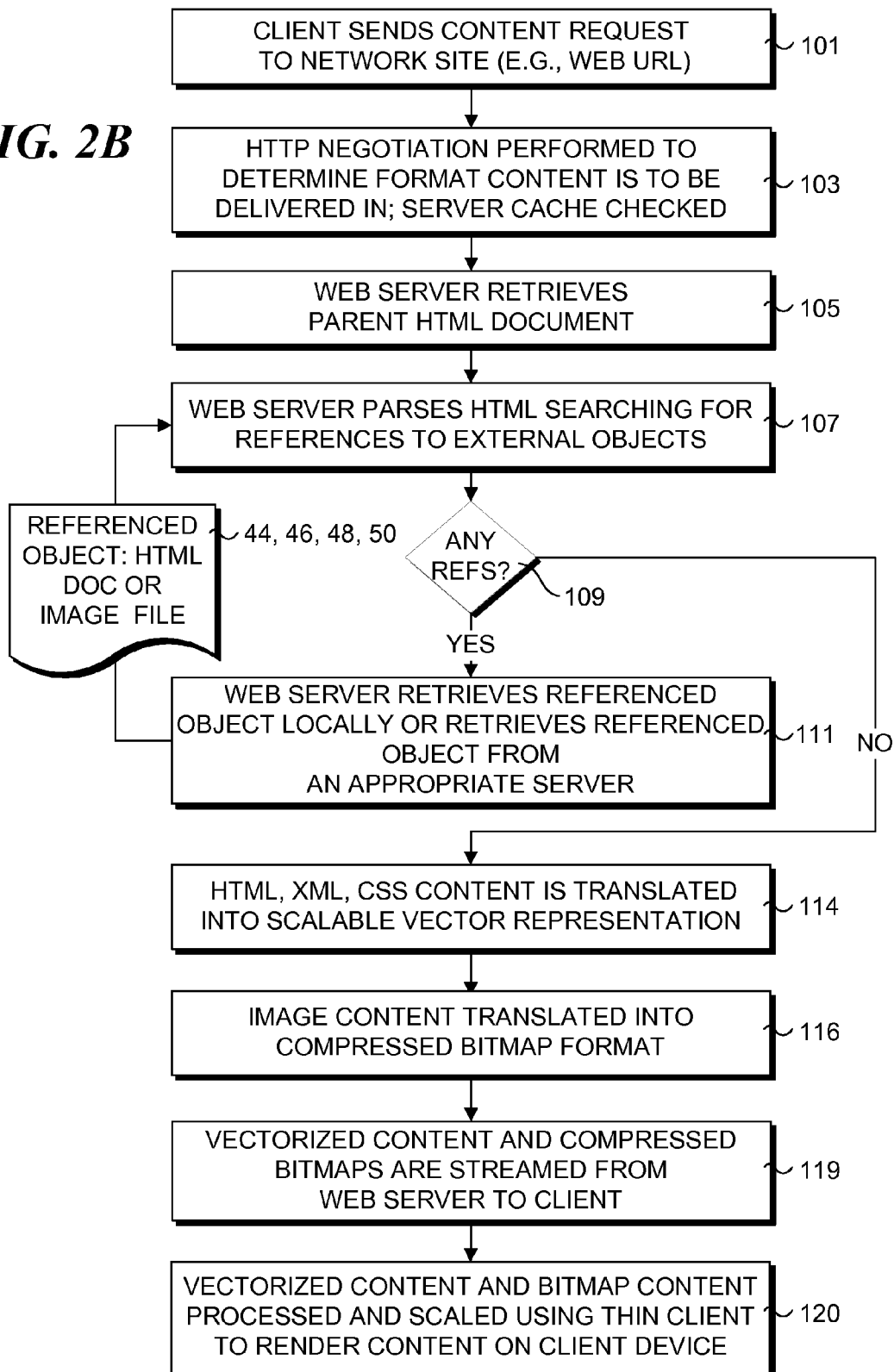
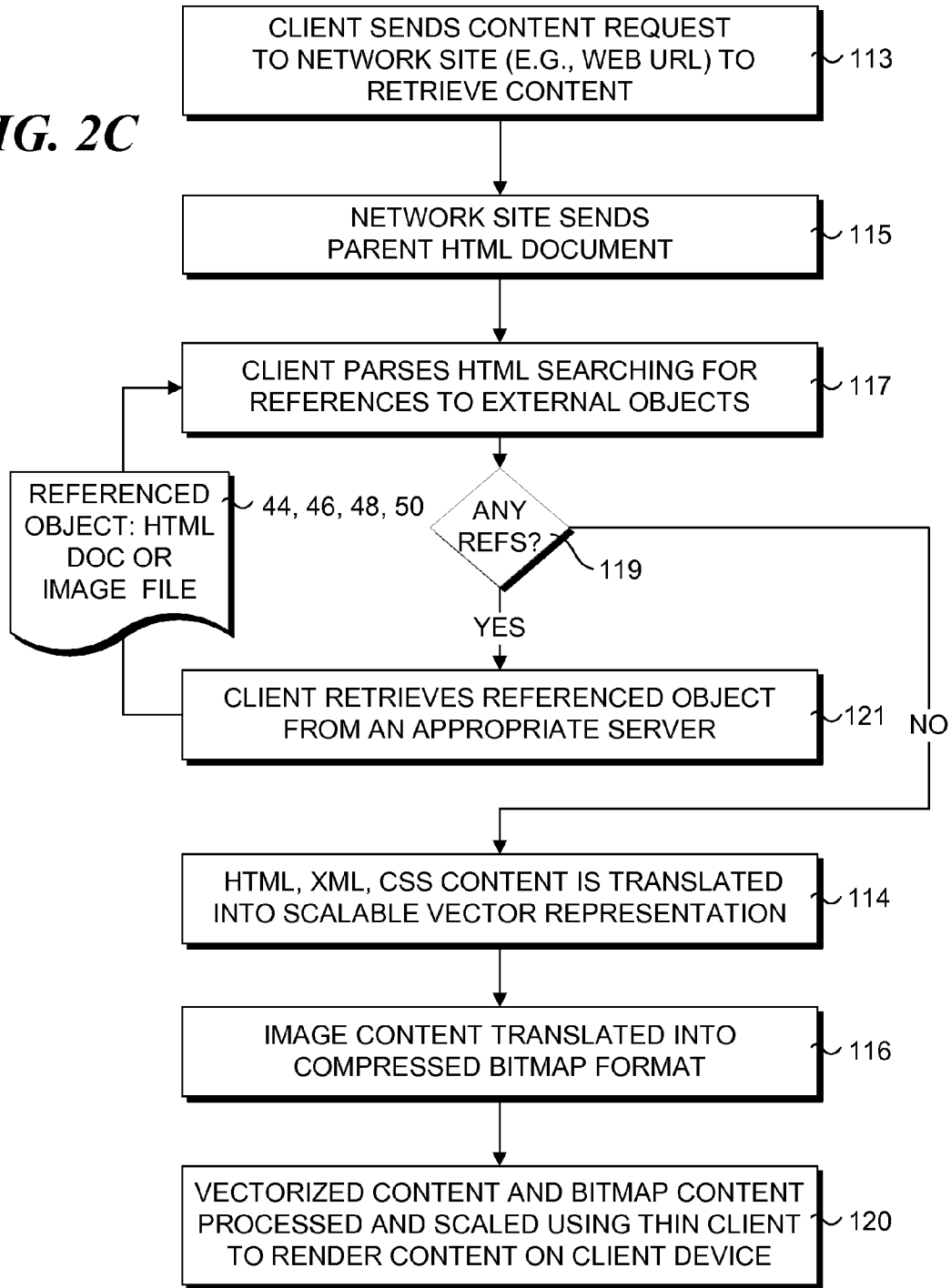


FIG. 2C



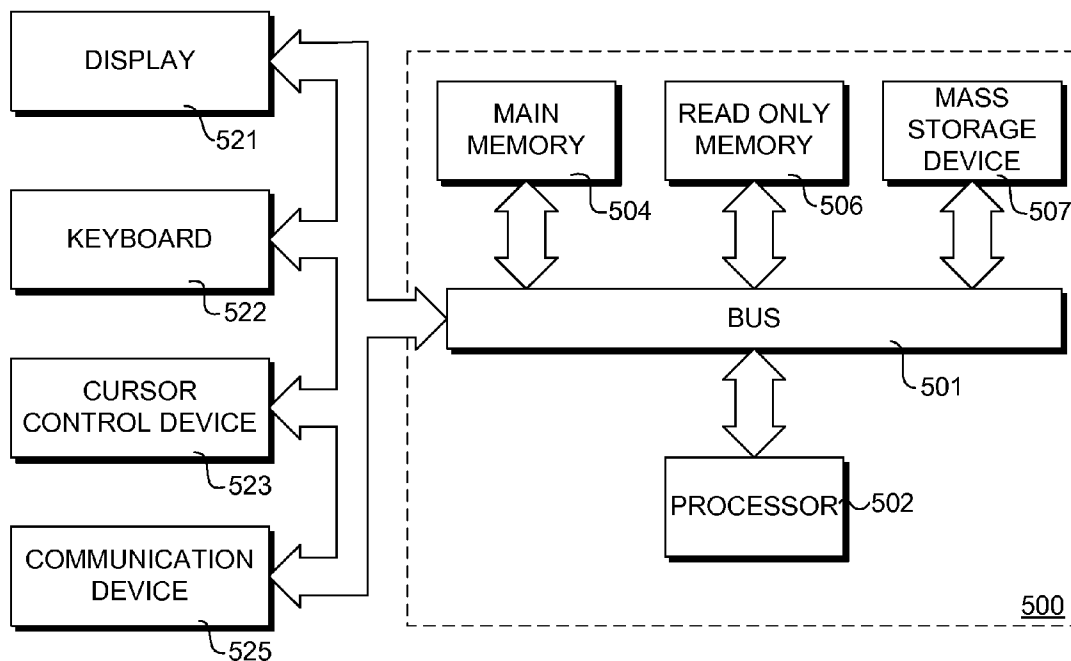
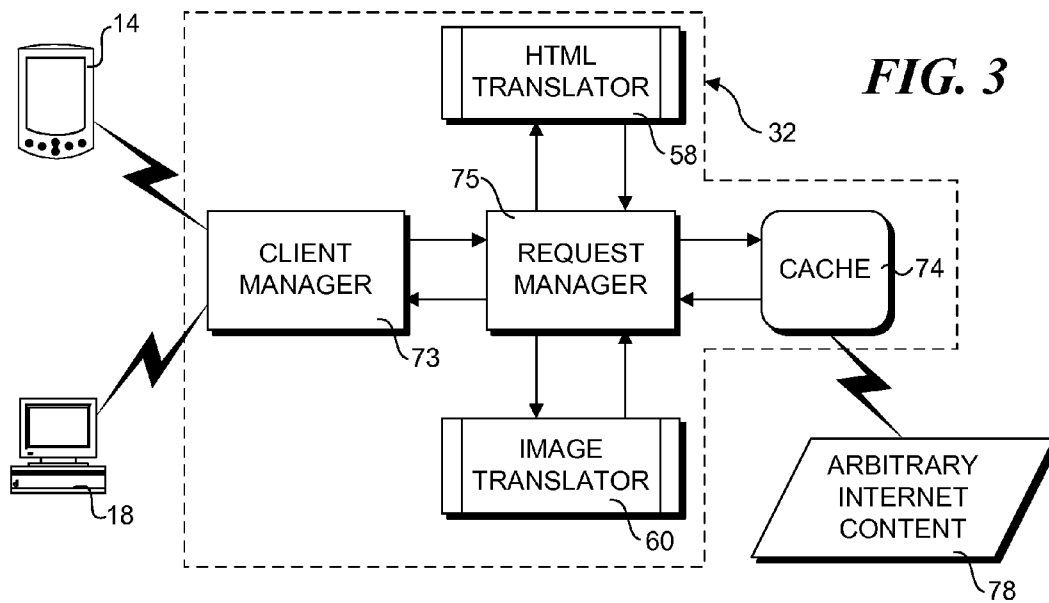
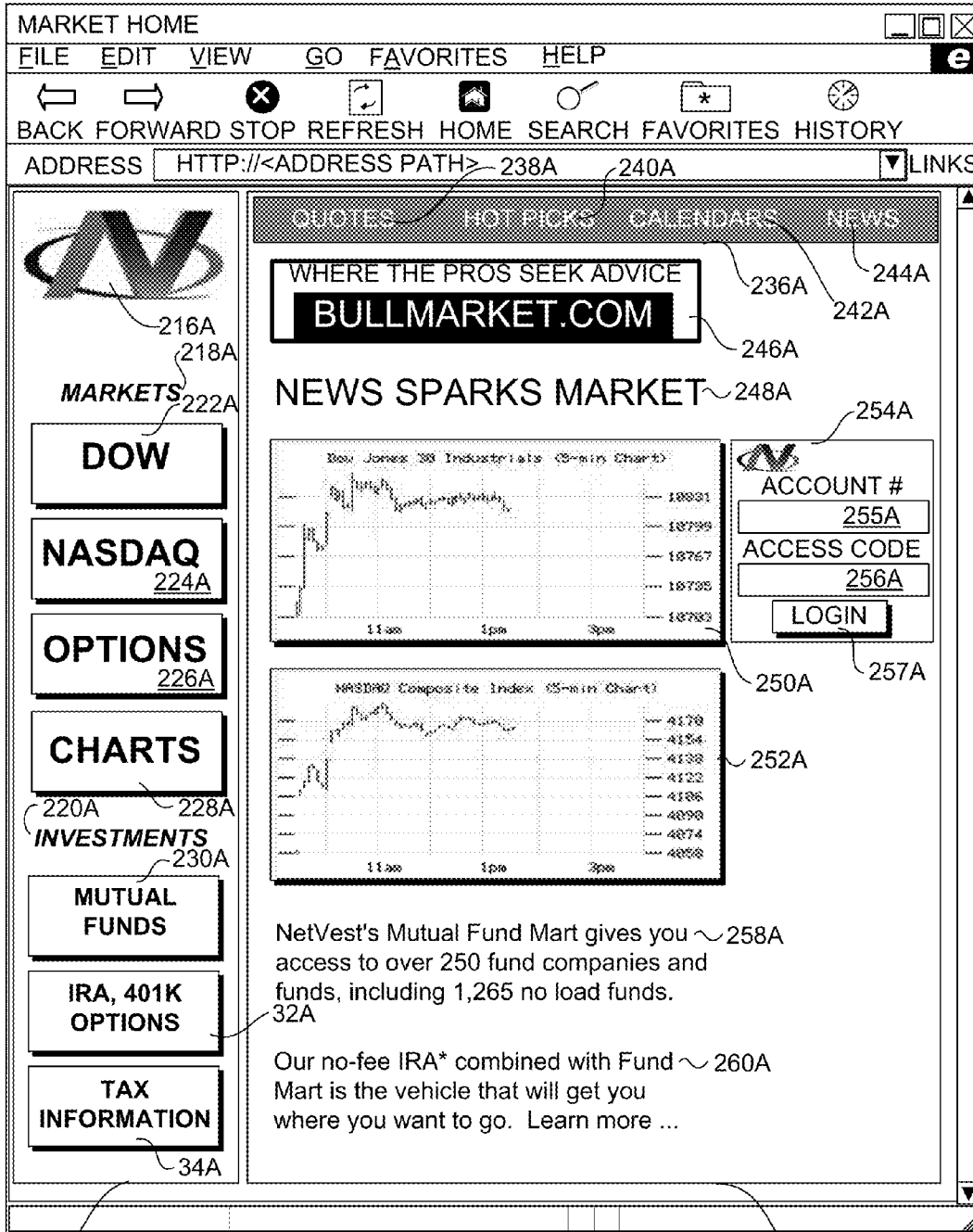


FIG. 10



212

210A

FIG. 4A

214

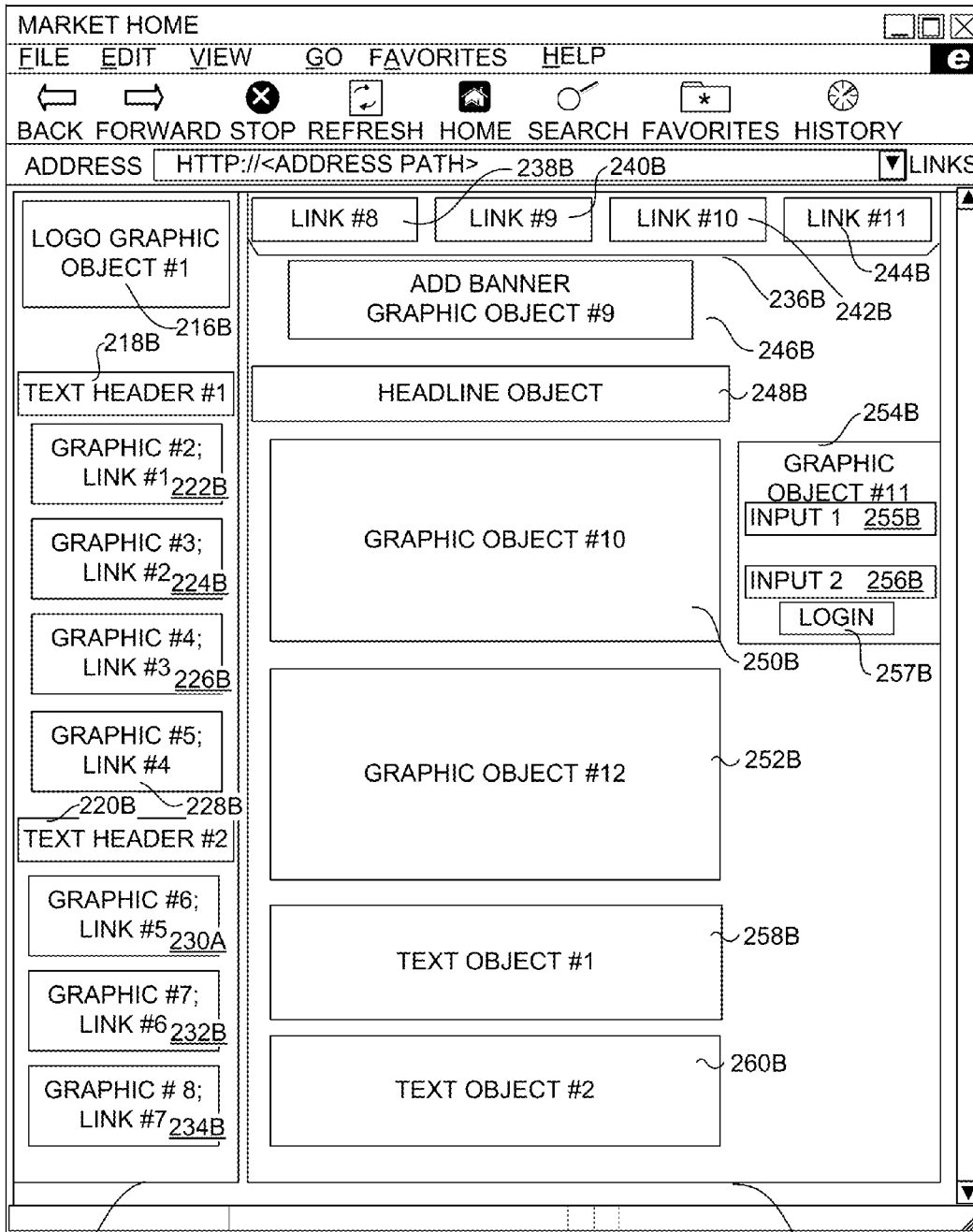
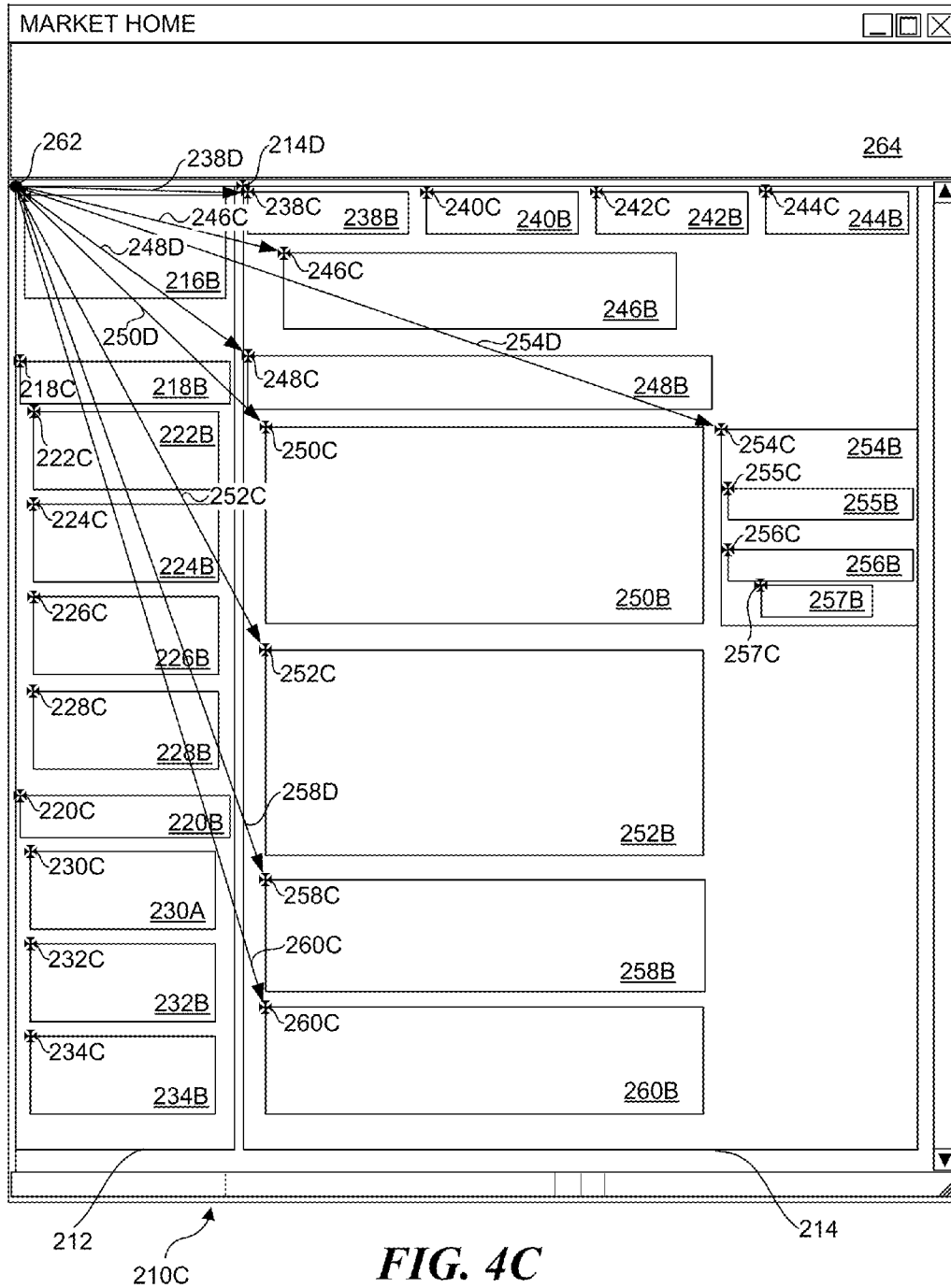


FIG. 4B



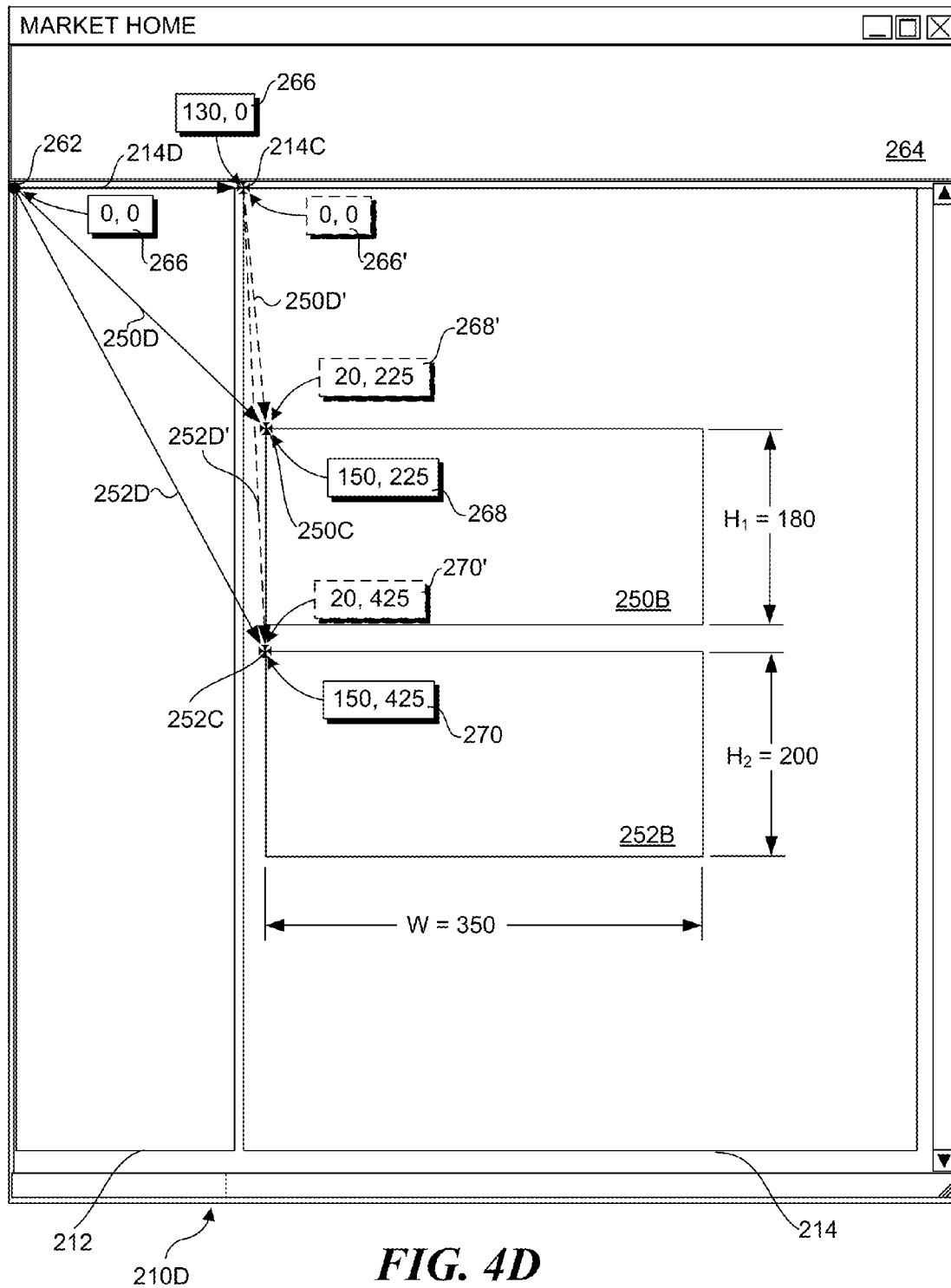
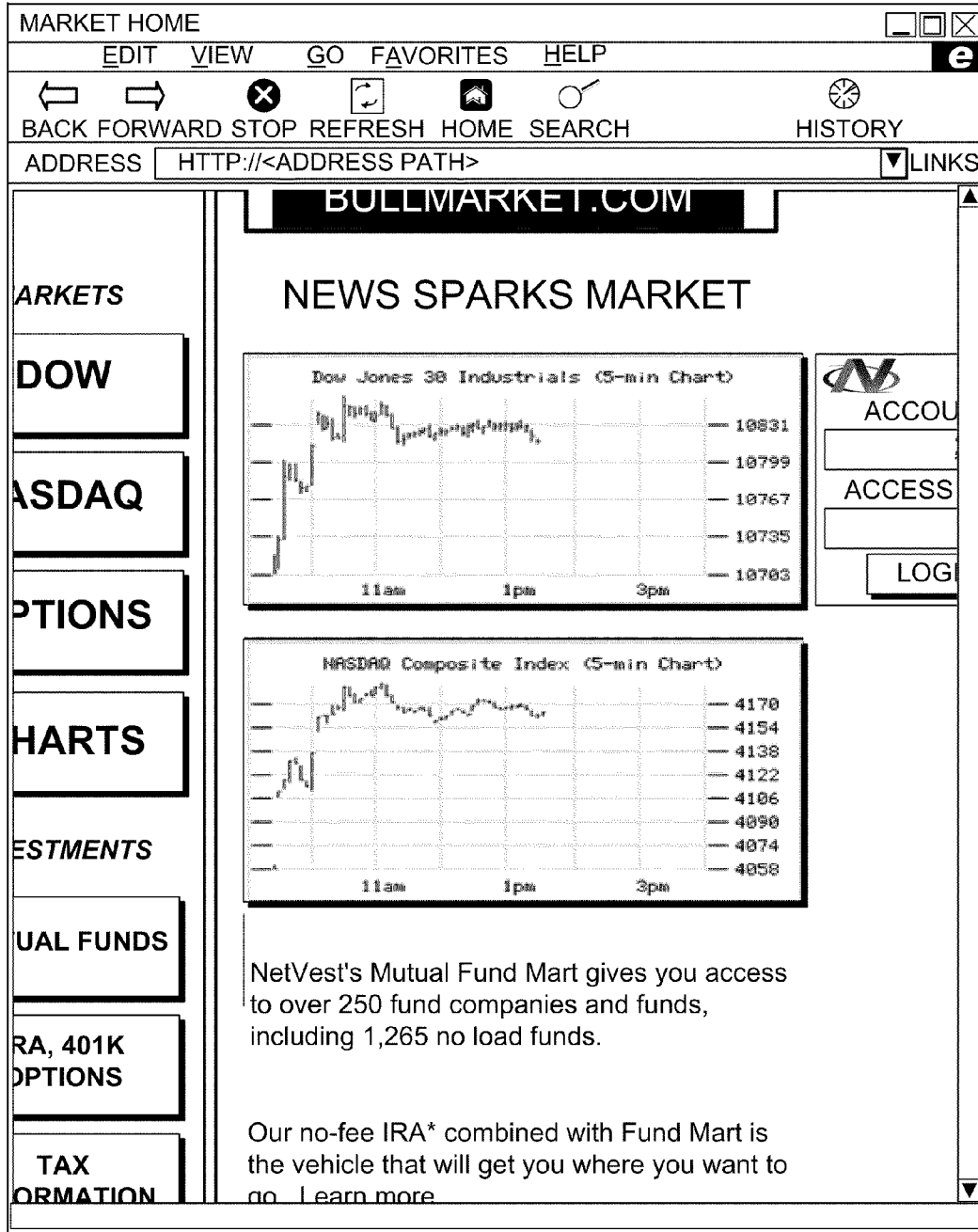


FIG. 4D



210E

FIG. 4E

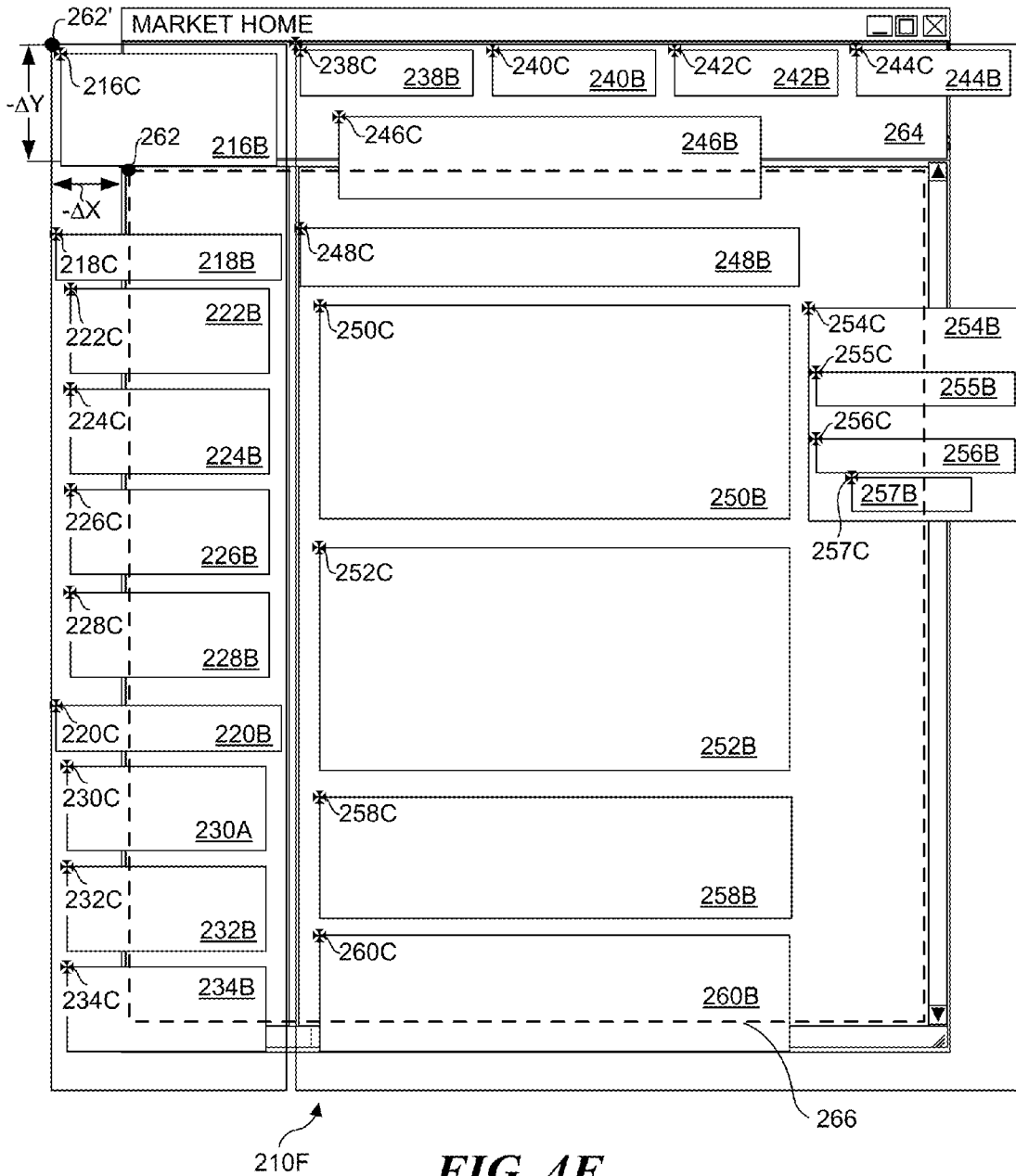


FIG. 4F

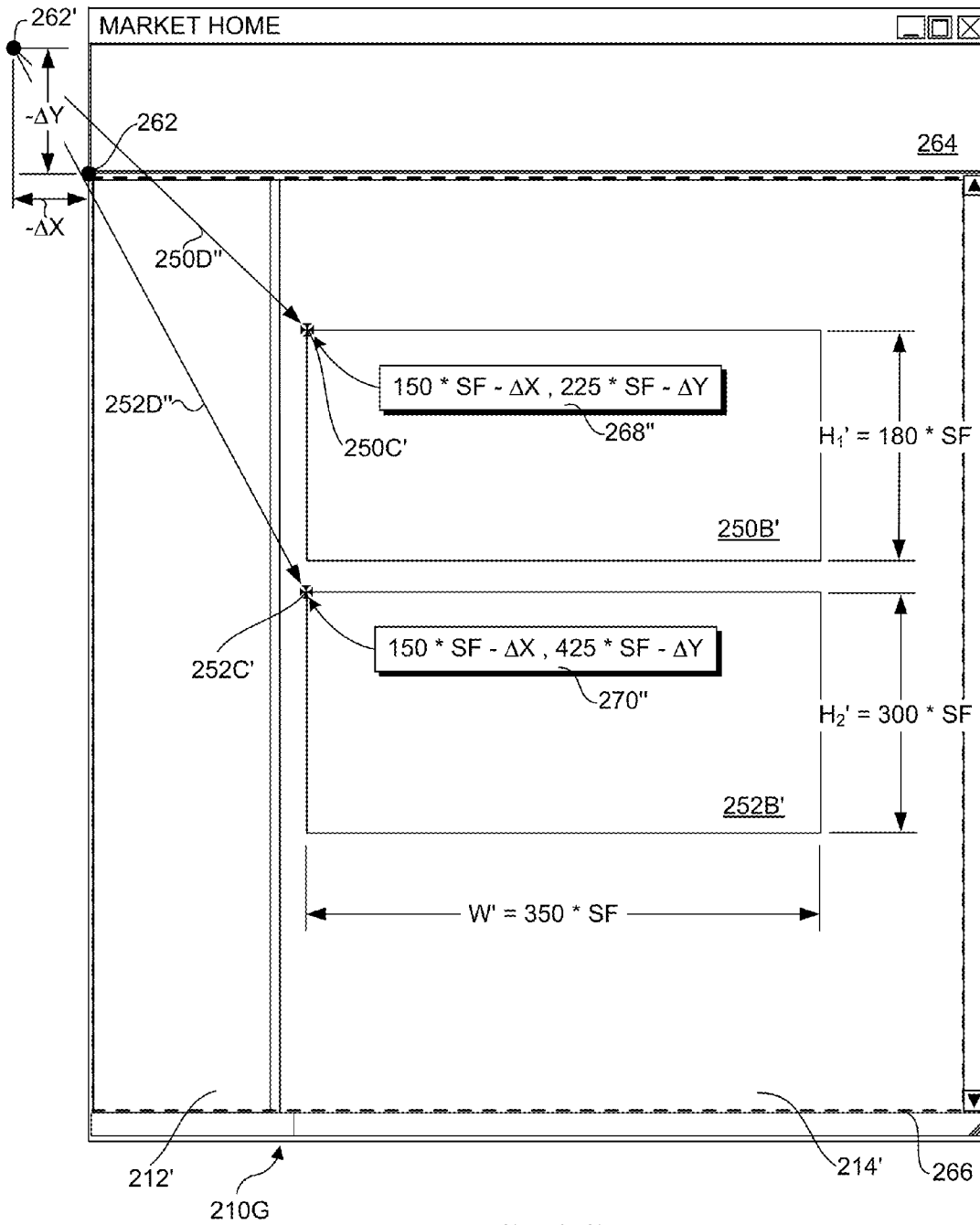
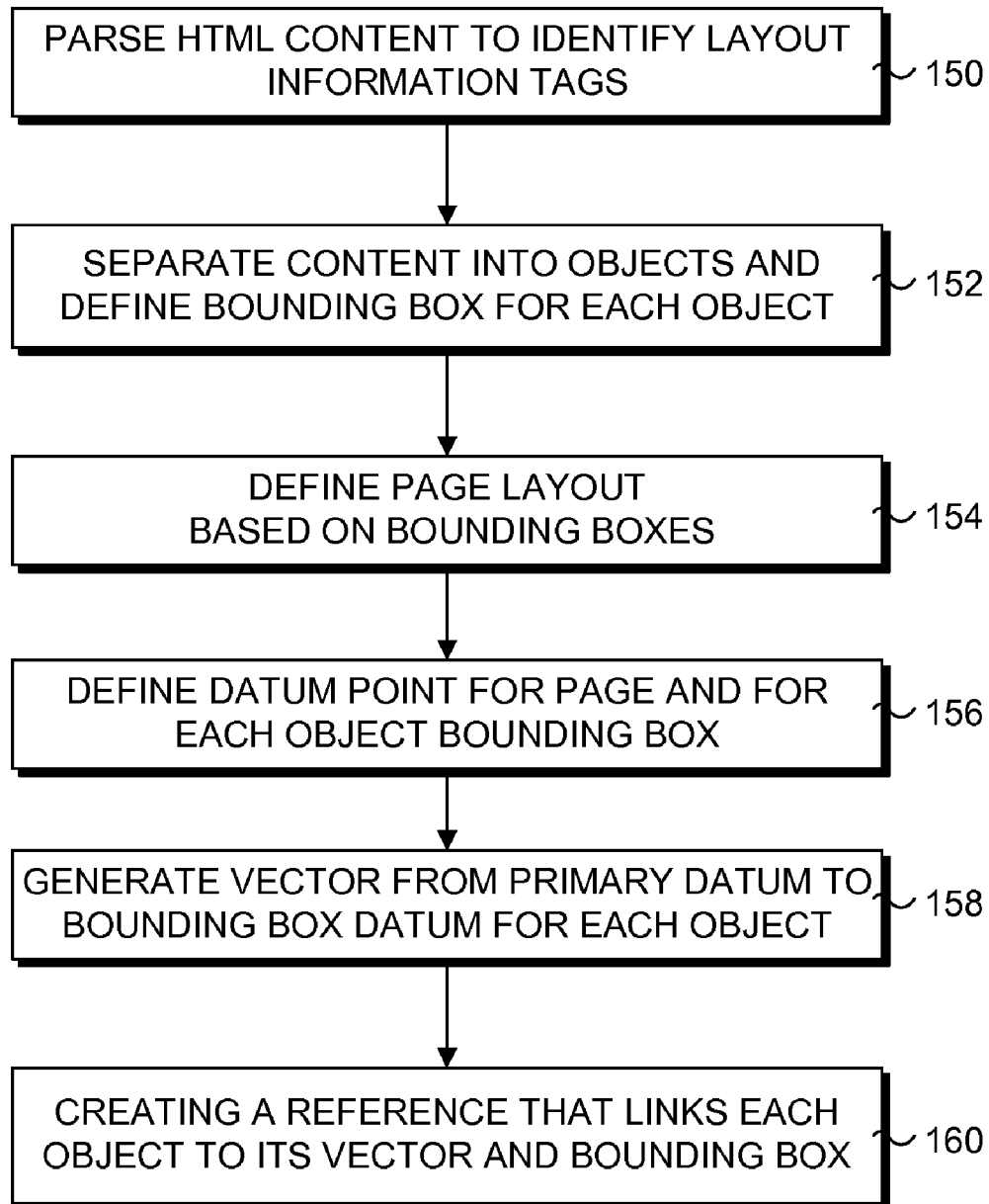


FIG. 4G

**FIG. 5**

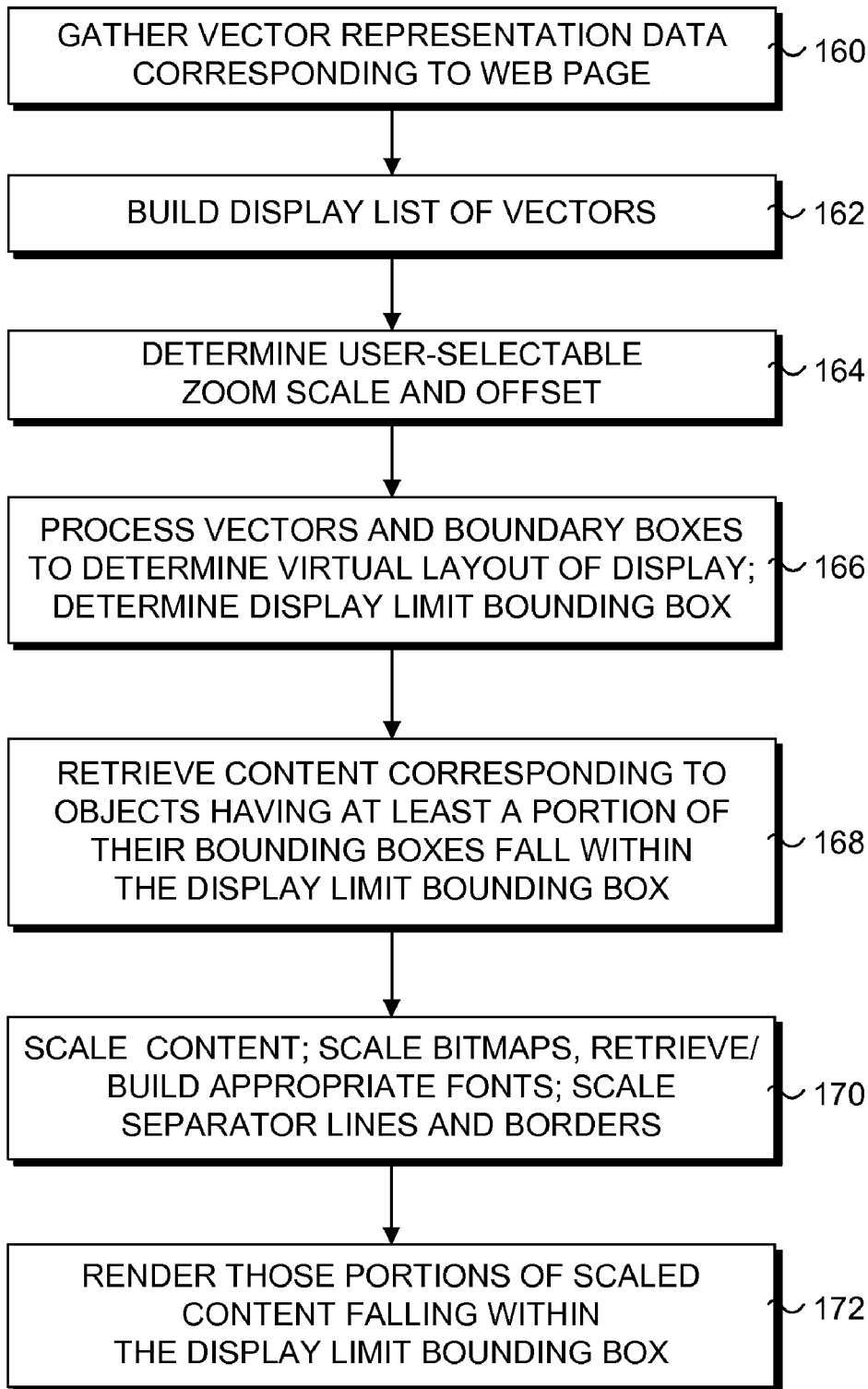


FIG. 6

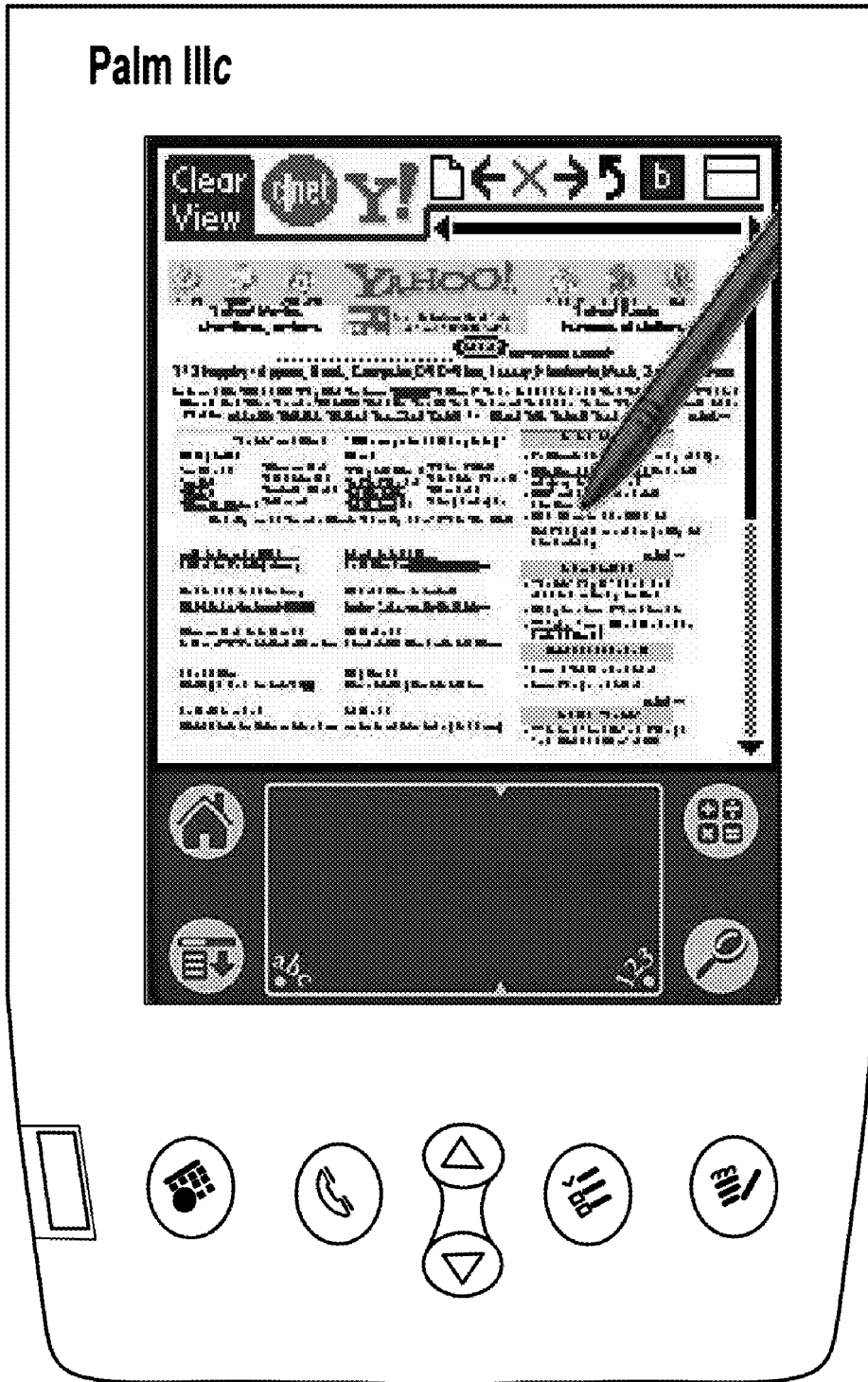


FIG. 7A

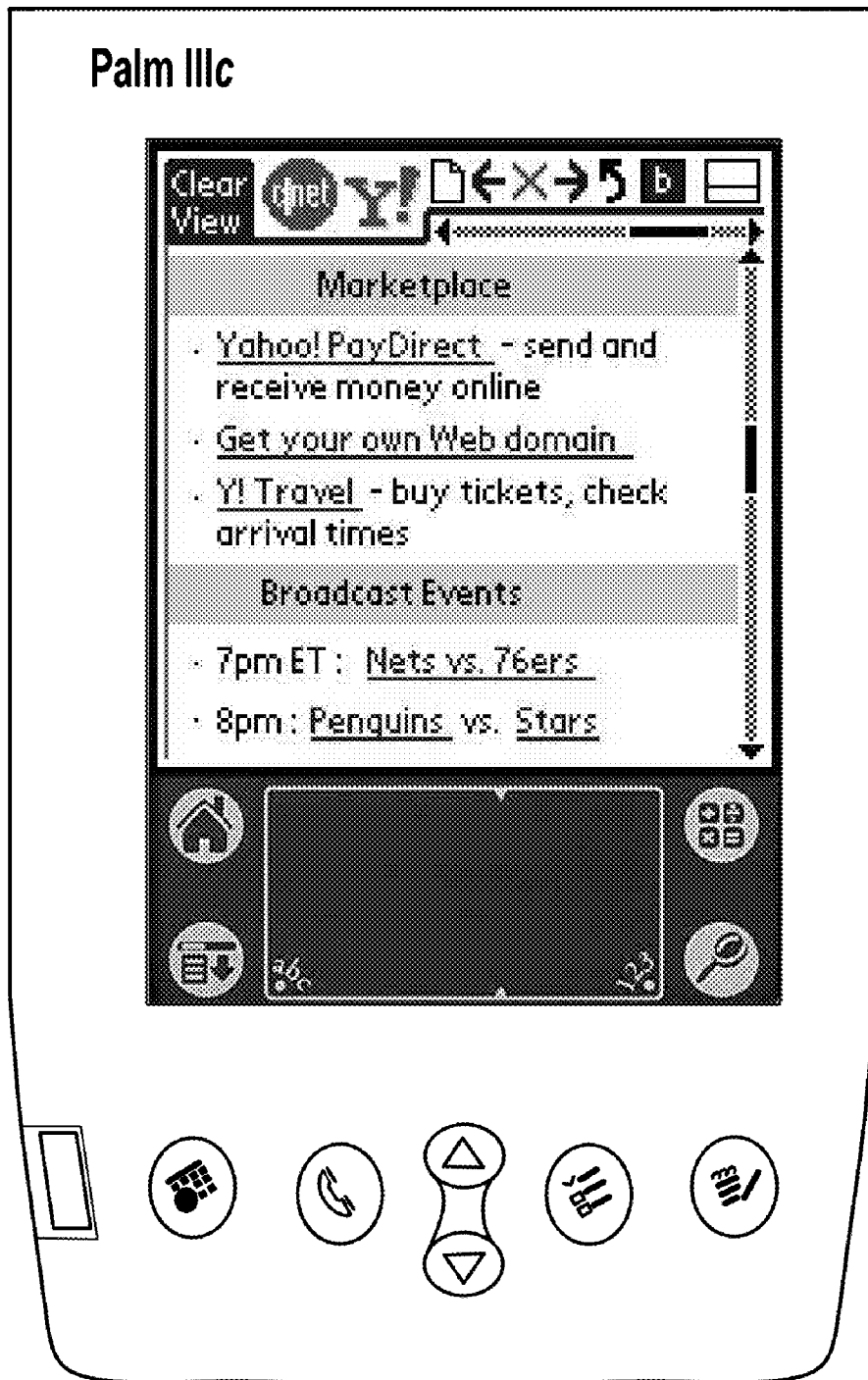


FIG. 7B

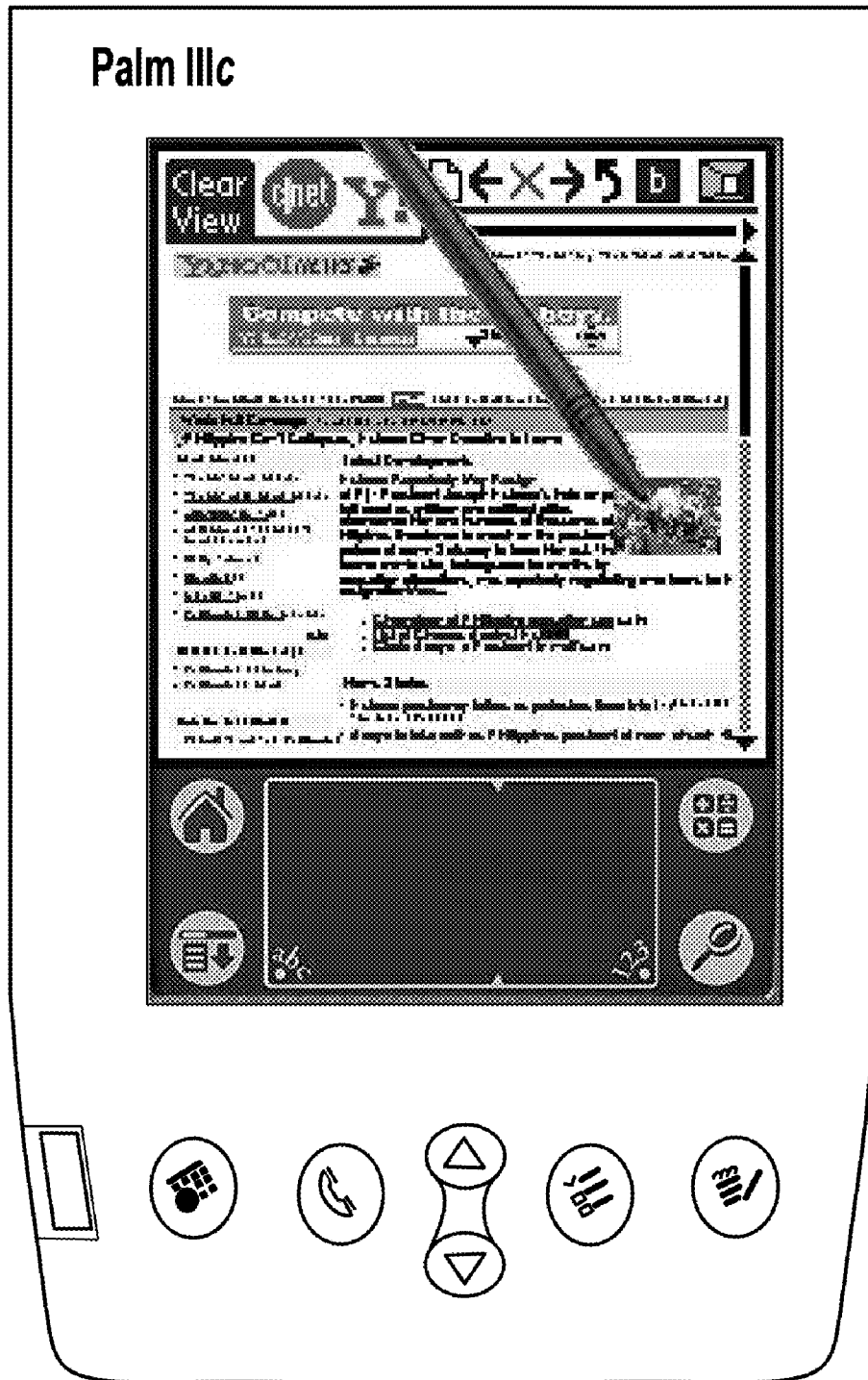


FIG. 8A

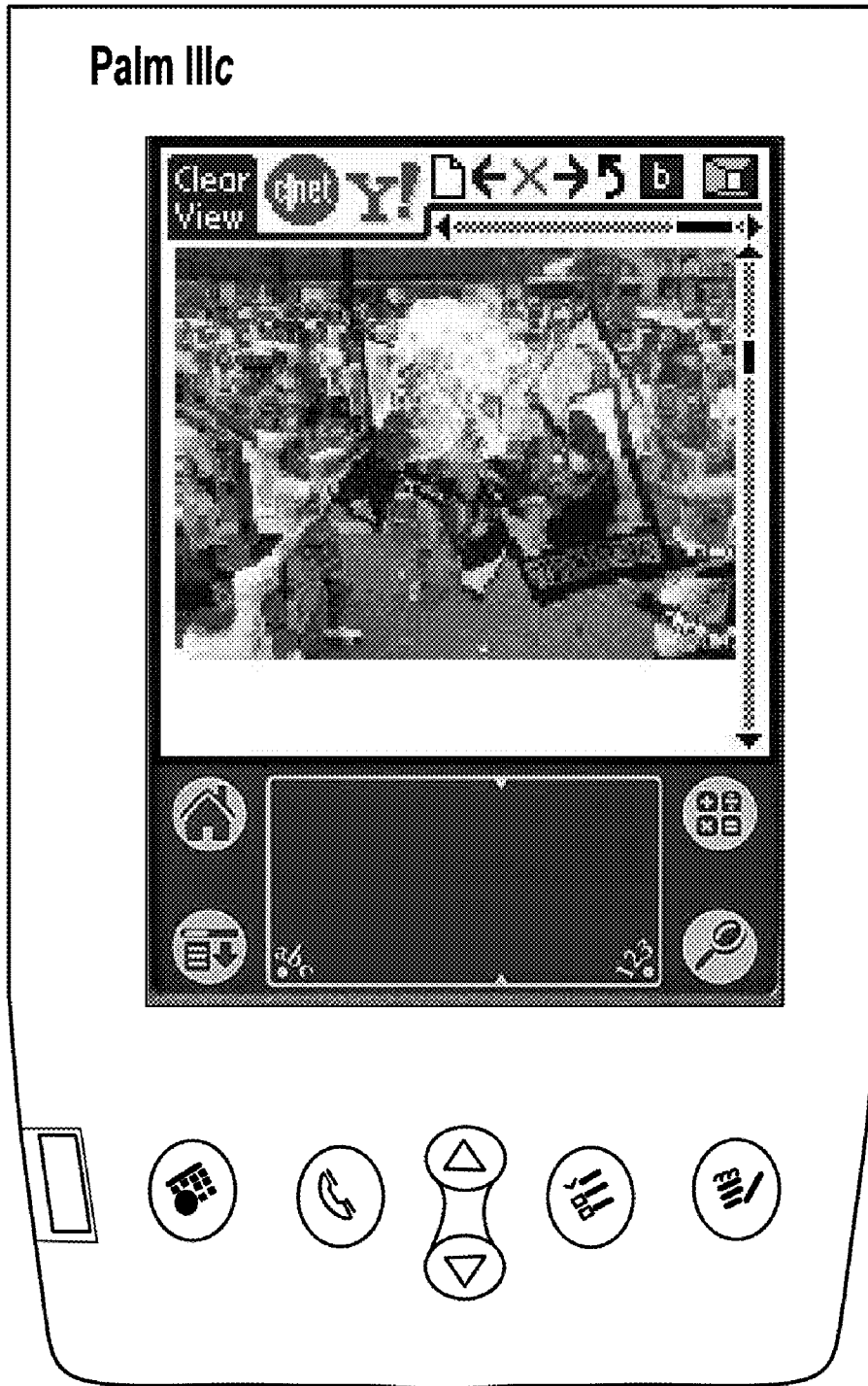


FIG. 8B

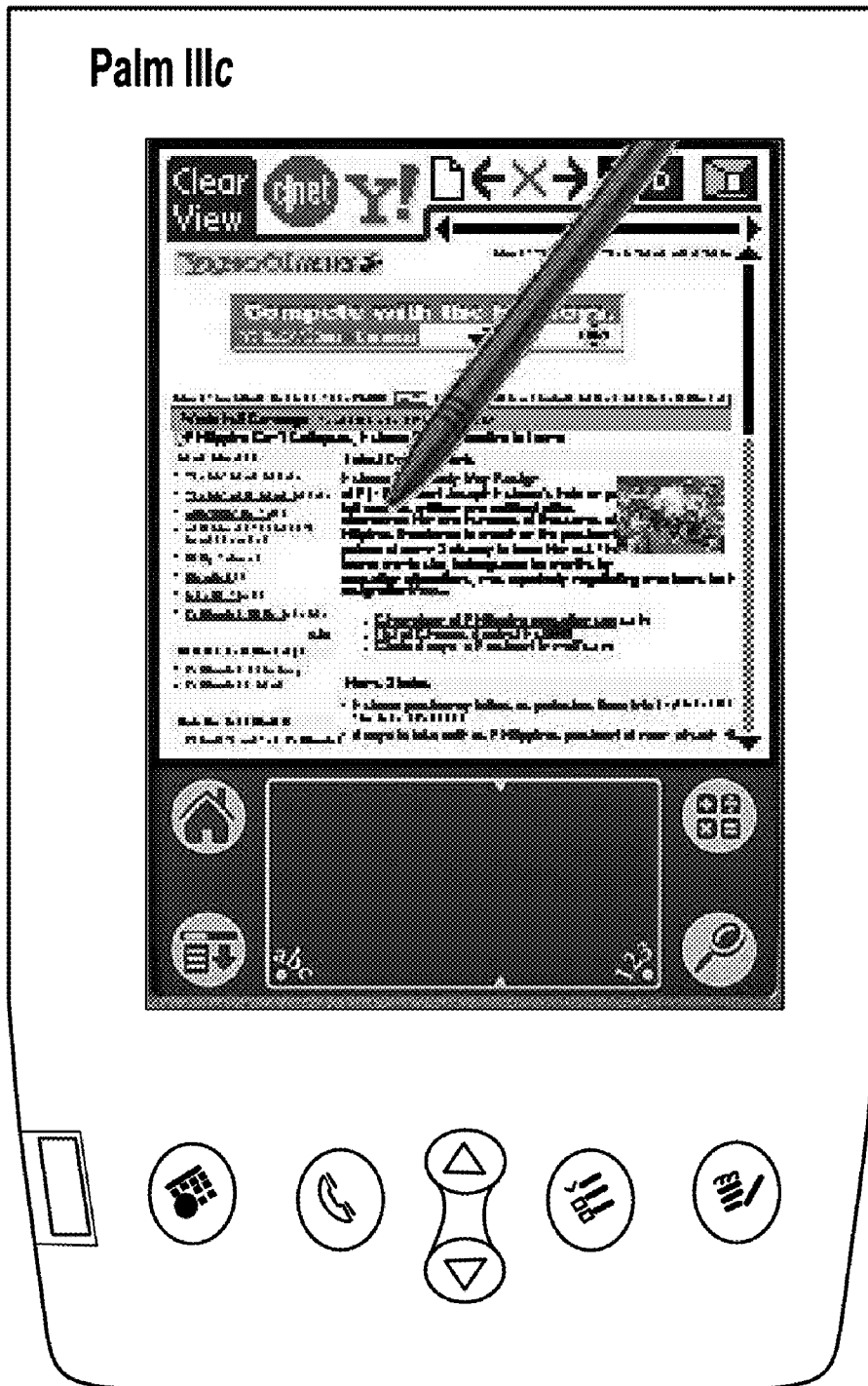


FIG. 9A

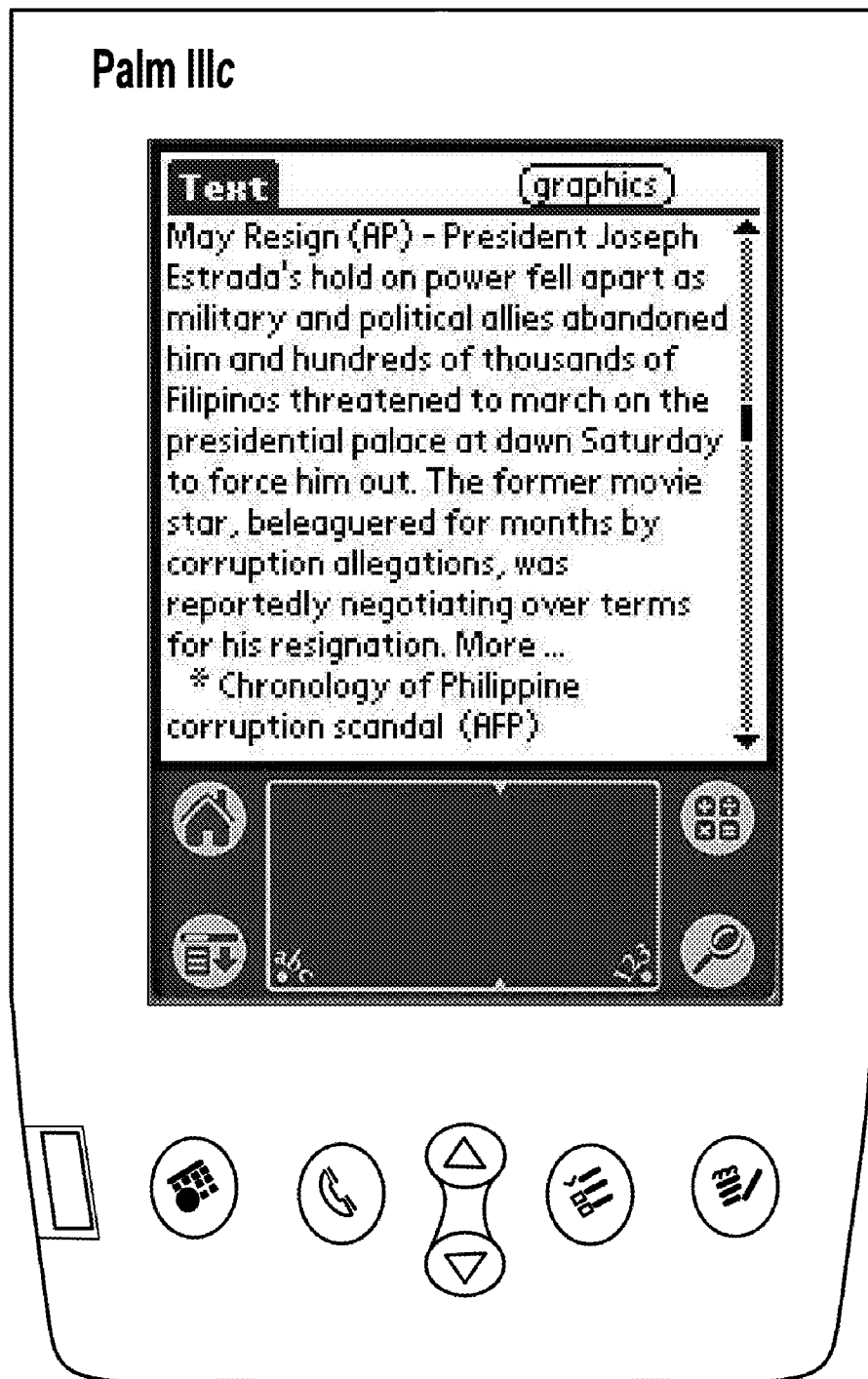


FIG. 9B

SCALABLE DISPLAY OF INTERNET CONTENT ON MOBILE DEVICES

RELATED APPLICATIONS

This application is a Continuation of U.S. Non-provisional application Ser. No. 09/878,097, filed Jun. 8, 2001, (issued as U.S. Pat. No. 7,210,099) entitled "RESOLUTION INDEPENDENT VECTOR DISPLAY OF INTERNET CONTENT," which is a Continuation-in-Part of U.S. Non-provisional application Ser. No. 09/825,511, filed Apr. 7, 2001, (Abandoned) entitled "RESOLUTION INDEPENDENT VECTOR DISPLAY OF INTERNET CONTENT;" the benefit of the filing dates of which is claimed under 35 U.S.C. §120. U.S. Non-provisional application Ser. No. 09/878,097 further claims the benefit of the filing dates of U.S. Provisional Application No. 60/211,019, filed Jun. 12, 2000, entitled "METHOD AND SYSTEM FOR RESOLUTION INDEPENDENT DISPLAY OF HTML AND XML CONTENT" and U.S. Provisional Application No. 60/217,345, filed Jul. 11, 2000, entitled "METHOD AND SYSTEM FOR SELECTION, RETRIEVAL, AND CONVERSION OF COMPUTER CONTENT TO VECTOR FORMAT FOR RESOLUTION INDEPENDENT DISPLAY," under 35 U.S.C. §119(e). The disclosure of each of the foregoing applications is incorporated by reference in its entirety herein for all purposes.

This application also contains subject matter related to Divisionals (of Ser. No. 09/878,097) U.S. Non-provisional application Ser. Nos. 11/045,649 (issued as U.S. Pat. No. 7,584,423) entitled METHOD, PROXY AND SYSTEM TO SUPPORT FULL-PAGE WEB BROWSING ON HANDHELD DEVICES, and 11/045,757 (issued as U.S. Pat. No. 7,461,353) entitled SCALABLE DISPLAY OF INTERNET CONTENT ON MOBILE DEVICES, both filed Jan. 28, 2005. This application also contains subject matter related to U.S. Non-provisional application Ser. Nos. 11/735,477 and 11/735,482, both filed on Apr. 15, 2007, 11/738,932 filed on Apr. 23, 2007, 11/868,124 filed on Oct. 5, 2007, and 12/326,092 filed on Dec. 1, 2008.

COPYRIGHT NOTICE

Contained herein is material that is subject to copyright protection. The copyright owner has no objection to the facsimile reproduction of the patent disclosure by any person as it appears in the Patent and Trademark Office patent files or records, but otherwise reserves all rights to the copyright whatsoever.

BACKGROUND OF THE INVENTION

1. Field of the Invention

The invention relates generally to viewing of Internet content on mobile devices, and more particularly concerns to novel processing of Internet and World Wide Web content to scalable forms for resolution-independent rendering and zoom- and pan-enabling the display of content on mobile devices.

2. Description of the Related Art

Text only Internet information browsers began as a project at the CERN, European Organization for Nuclear Research, facility in Geneva Switzerland. From its inception the intent was to provide a mesh or web of access to data with a common user interface. Browsers moved from the academic environment when NCSA, the National Center for Supercomputing

Applications at the University of Illinois in Urbana-Champaign developed Mosaic, an Internet information browser and World Wide Web client.

Internet content is stored in multiple file formats. These formats include HTML (Hyper Text Markup Language) and XML (eXtended Markup Language) as well as graphic file format GIF (Graphics Interchange Format) and JPEG (Joint Photographic Experts Group). These four file formats constitute the majority of Internet content. Font size and resizing display area for content can alter the size of the display of Internet content in existing browsers. The majority of Internet content displays as a flat single resolution with no browser support for zoom.

Much of the Internet content has been designed for display on desktop computers with a single target resolution. Even though HTML has the ability to adapt to changes in screen resolution, major Internet content providers have chosen to create their Web pages using fixed resolution structures, such as tables. This gives them the ability to control the look and feel of their Web sites. This fixed resolution approach has evolved to the point that the fixed resolution layout of Web pages has become the most common method to brand or uniquely identify Web sites. While this fixed resolution approach is good for site branding and product differentiation it does present a daunting technical problem for display of Internet content (designed for desktop computers) on small screen, low resolution, or different aspect ratio devices, such as cell phones and hand held computers.

BRIEF SUMMARY OF THE INVENTION

In accordance with aspects of the invention, mobile devices enabled to support resolution-independent scalable display of Internet (Web) content to allow Web pages to be scaled (zoomed) and panned for better viewing on smaller screen sizes are disclosed. The mobile devices employ novel processing of original Web content, including HTML-based content, XML, cascade style sheets, etc. to generate scalable content. The scalable content and/or data derived therefrom are then employed to enable the Web content to be rapidly rendered, zoomed, and panned. Display lists may also be employed to provide further enhancements in rendering speed.

According to further aspects, the mobile devices employ touch-sensitive display screens that enable users to provide various inputs to control display of content within Web pages. Exemplary user inputs include tap-based inputs to selectively zoom in on columns, images, and paragraphs. Users can also define a window to zoom in on via the touch-sensitive display.

Other features of the present invention will be apparent from the accompanying drawings and from the detailed description that follows.

BRIEF DESCRIPTION OF THE DRAWINGS

The appended claims set forth the features of the invention with particularity. The invention, together with its advantages, may be best understood from the following detailed description taken in conjunction with the accompanying drawings of which:

FIG. 1A is a block schematic diagram illustrating a first exemplary system infrastructure in accordance with the present invention in which content translation services are performed by a third-party proxy service that translates content requested from a client that is retrieved from one or more network resources into a scalable vector representation and delivers the translated content to the client;

FIG. 1B is a block schematic diagram illustrating a second exemplary system infrastructure in which the translation of content is performed at a content provider's web site and delivered directly to the requesting client;

FIG. 1C is a block schematic diagram illustrating a third exemplary system infrastructure in which content received from one or more network sources is translated into a scalable vector representation at the client;

FIG. 2A is a flowchart illustrating how data is retrieved, processed and transferred in accordance with the system infrastructure of FIG. 1A;

FIG. 2B is a flowchart illustrating how data is retrieved, processed and transferred in accordance with the system infrastructure of FIG. 1B;

FIG. 2C is a flowchart illustrating how data is retrieved, processed and transferred in accordance with the system infrastructure of FIG. 1C;

FIG. 3 is a block schematic diagram illustrating an exemplary architecture corresponding to the proxy server of FIG. 1A;

FIG. 4A is a representation of an exemplary web page as displayed on a conventional browser;

FIG. 4B is a schematic diagram illustrating various objects that are generated based on the HTML code of the web page of FIG. 4A;

FIG. 4C is a schematic diagram illustrating a set of vectors and bounding boxes corresponding to the objects generated in FIG. 4B;

FIG. 4D is a schematic diagram illustrating how various vectors and bounding boxes may be defined in accordance with the invention;

FIG. 4E is a representation of the web page of FIG. 4A after it has been offset and scaled in accordance with the invention;

FIG. 4F is a schematic diagram illustrating new datum points and bounding boxes corresponding to the scaled and offset web page;

FIG. 4G is a schematic diagram illustrating new vectors and bounding box parameters for a pair of objects in the scaled and offset web page;

FIG. 5 is a flowchart illustrating the logic used by the invention when translating content into a scalable vector representation of that content;

FIG. 6 is a flowchart illustrating client-side operations that are performed to create a rendered display page based on the translated content the client receives and user-input;

FIGS. 7A and 7B are representations of a nominal and a zoomed in column view of an exemplary web page as they might appear on a Palm device;

FIGS. 8A and 8B are representation of nominal and zoomed in view of an exemplary graphic image as they might appear on the Palm device;

FIGS. 9A and 9B are representations of a nominal and zoomed in view of a text portion of a web page as they might appear on the Palm device; and

FIG. 10 illustrates an exemplary computer system that may be used for implementing various aspects of embodiments of the invention.

DETAILED DESCRIPTION OF THE INVENTION

Apparatus and methods are described for creating resolution independent vector display of Internet content to allow it to be scaled (zoomed) larger and smaller for better viewing or to fit any resolution or screen size. In addition, infrastructure and methods are provided for delivering such content to clients.

In the following description, for the purposes of explanation, numerous specific details are set forth in order to provide a thorough understanding of the present invention. It will be apparent, however, to one skilled in the art that the present invention may be practiced without some of these specific details. In other instances, well-known structures and devices are shown in block diagram form.

The present invention includes various operations, which will be described below. The operations of the present invention may be performed by hardware components or may be embodied in machine-executable instructions, which may be used to cause a general-purpose or special-purpose processor or logic circuits programmed with the instructions to perform the operations. Alternatively, the operations may be performed by a combination of hardware and software.

The present invention may be provided as a computer program product that may include one or more machine-readable mediums having stored thereon instructions, which may be used to program a computer (or other electronic devices) to perform a process according to the present invention. The machine-readable medium may include, but is not limited to, floppy diskettes, optical disks, CD-ROMs, and magneto-optical disks, ROMs, RAMs, EPROMs, EEPROMs, magnetic or optical cards, flash memory, or other type of media/machine-readable medium suitable for storing electronic instructions. Moreover, the present invention may also be downloaded as a computer program product, wherein the program may be transferred from a remote computer (e.g., a server) to a requesting computer (e.g., a client) by way of data signals embodied in a carrier wave or other propagation medium via a communication link (e.g., a modem or network connection). Accordingly, herein, a carrier wave shall be regarded as comprising a machine-readable medium.

Client Overview

According to one embodiment, an ultra-thin client-side viewer provides the graphics, linking, caching, and function handling capabilities necessary for extending the web to almost any platform. It is designed as a lightweight browser (micro-browser) running directly on device operating systems. In alternative embodiments, the client-side viewer may be deployed as a standard browser plug-in, or Java applet for extending browser functionality. In one embodiment, the client-side viewer attains its small size and efficiency by taking advantage of the power of SVF (Simple Vector Format) to describe almost any current web content. SVF files can be handled with a tiny fraction of the client code required by normal web browsers because current browsers must interpret a large and growing number of file types and their idiosyncrasies. SVF was originally designed to handle a superset of the most commonly used file formats in the complex world of CAD. It can accommodate not only new graphical functions, but the storage and transfer of almost any foreseeable new functional capability. SVF has been under consideration by the W3C (World Wide Web Consortium) for adoption as a standard for vector content on the World Wide Web.

By working tightly with a server-side content translator, web content and functionality can be passed seamlessly to the end user platform without any degradation in the look or feel of the output. In addition, because the resulting file graphics are handled as vectors, the end user can control real time changes in the size of text and graphics as well as what portion of the file is viewable in the display. This "zoom and pan" capability, familiar to CAD and other vector content software users, adds dramatically to the usability of non-standard display sizes. For very small displays, real time zooming and panning allows the user to see graphics and text at sizes that make them easily readable, and then "back up" to view an

entire page for context or pan in any direction for navigation. Because the client-side viewer manipulates vectors, there is no loss in quality as the display is zoomed. The graphics rendering engine within the client is so efficient that file manipulation happens in a fraction of a second. There is no perceptible wait for the user as the file is resized, or the window is repositioned. Content created for one display resolution now can be sized, real time, for any other display without degradation. Besides making small displays eminently usable, this technology extends web content into some surprising new arenas. For example, it enables normal desktop displays to be effective for individuals with visual impairment, or for content designed for 640×480 standard PC monitors to be shown without degradation on web billboards now appearing in cities like Seattle and San Francisco.

With a client of such extraordinary power packed in a tiny footprint, end user device manufacturers can free up valuable memory space for pre-fetching, caching and pre-loading content, dramatically improving performance for use in low bandwidth and portable applications. In the example of a wireless handheld device where expensive flash memory must be used instead of more cost effective bulk storage technology, the difference between consuming 10's of megabytes of flash memory with a standard browser versus running the client-side viewer described herein is dramatic.

Those "saved" megabytes of memory are now available for impressive interfaces, caching of often used content, and pre-fetching of intelligently selected linked files or pre-loading of content for targeted applications. For example, in a mapping application, the map tiles surrounding the viewed map could be downloaded and stored while the user was working with the initial tile, enabling an experience remarkably free from the current frustrations of waiting for a new map to be transferred for even the smallest change in magnification or coverage. If the user knows ahead of time what city they will visit on a business trip, maps and additional travel information in great detail could also be pre-loaded using a high bandwidth connection at home or in the office before heading out to shop or conduct business in the city. Additionally, SVF is a more efficient way to store web content. Resulting content files are reduced in size by anywhere from 20 to 80 percent over their source. SVF is also very compressible. With target file size reduction in the range of 90%, SVF files can take up as little as 1/10th the space of the web files in current use. This means that pre-translated content can be moved up to 10 times the rate of current web pages, and as much as 10 times as many pages, maps, stock charts, etc. can be stored for instant retrieval on the hand held platform as can be handled with current web technology.

When used on content created natively in SVF, additional capability can be extended to the client-side viewer.

Graphing the performance of stocks over time is only one use of SVF's ability to handle streams of data. Handling the output from financial systems, transactional systems, ERP packages, and CRM systems becomes easier and more flexible. Of course, systems integrators don't have to use these powerful capabilities to start with. If the target system provides web interfaces, these can be viewed, as designed, with no additional software to write, and no changes to the design or layout of the interface.

Server Overview

Enabling the client-side viewer to be so small and powerful is the server-side content translator. The server-side content translator rapidly translates Web content to SVF, compresses and encrypts the SVF results if desired, and transfers the vector formatted results to the client-side viewer. Alternatively, SVF files can be cached or stored in a file system for

fetching and transfer at a later time. Pre-translated or cached content transfers are significantly faster as no conversion overhead is incurred, and file sizes are reduced using the more efficient SVF. Combine that with standard compression algorithms selectable for use with the client-side viewer for additional performance improvements.

During the translation process, and in the process of serving cached, pre-translated, or native SVF content, output files are "streamed" to the client-side viewer. Although this does not decrease the total time for file transfer, it can significantly improve the effective system performance for the end user. Content can be selectively streamed, with text and links coming through first, followed by graphic images and other content, for example. Should the user be accessing a link, rather than having interest in the entire file served, links can be selected early in the transfer and the next file download started immediately. In addition to streaming, the server-side content converter may also layer the content by type. This means that text can be put in one layer, links in another, GIF images in another, Javascript in another and so on. Layers can be turned on or off depending upon client capabilities, making files for less capable clients, or for users interested in a reduced functionality, higher transfer performance mode to be handled automatically.

All operational modes may be controlled through an administrative interface or accessible through a straightforward API (Application Program Interface). Furthermore, the system works with existing firewalls and within standard security protocols. In more secure modes, the server-side content converter and the client-side viewer may operate using Public/Private key authentication and encryption.

Exemplary System Infrastructures

In the following paragraphs, a description of three exemplary system infrastructures is provided. Schematic illustrations of these system infrastructures are shown in FIGS. 1A, 1B, and 1C. It is noted that like-numbered components in these Figures perform substantially the same function. Therefore, any discussion of the functions of a component with reference to one or more of the infrastructures generally may apply to the other infrastructures as well, unless specifically noted otherwise.

A first of exemplary system infrastructure 10A for implementing the invention is shown in FIG. 1A. Infrastructure 10A enables various clients, including wireless devices such as a cellular phone 12, a wireless-enabled PDA 14, and a wireless-enabled laptop computer 16, as well as landline computers 18, 20, and 22, to request content that is accessible via a network such as the Internet 24 to be retrieved from selected network resources, including web servers 26 and 28 and an FTP site 30, wherein the content is translated into a scalable vector representation (e.g., SVF, also referred to herein as "vectorized content") through use of a proxy server 32 and sent to the requesting client. Upon being received by the client, the vectorized content is processed and rendered using a thin client to enable a user to view the content on the client device.

With reference to the flowchart of FIG. 2A, the foregoing process is initiated by a client in a block 100, wherein the client submits a request to proxy server 32 to retrieve and convert selected content. As depicted by a transfer path 34, this comprises sending data 36, which includes content network location indicia from which the content can be retrieved and proxy server network location information by which the content request may be delivered to over Internet 24 to proxy server 32. Typically, it will be desired to retrieve a particular web page. Accordingly, the content network location indicia will comprise a URL (uniform resource locator) for the web

page. Similarly, the proxy server network location information may also comprise a URL corresponding to a network access point for the proxy server. Optionally, the location information may comprise a network IP address for one or both of the content location and the proxy server location. If the content is to be retrieved from an Internet resource, the request will typically be sent using the HyperText Transfer Protocol (HTTP) over the TCP/IP transport.

Next, in a block **102**, the request is received by the proxy server and the proxy server checks its cache to see if it already has the request content in its cache. If it does, it sends this cached content back to the client. If it does not have the requested content cached, the proxy server sends out a request to retrieve the content from the network resource. For illustrative purposes, it will be assumed for the present example that the desired content comprises a web page that is stored on web server **26**. Typically, when the requested content comprises a web page, the content may be retrieved using conventional web content retrieval techniques, such as that employed by various modern browser clients, including Netscape Navigator and Internet Explorer. This generally comprises providing routing information, such as the URL for the web page (URL **38**) to routing services provided by Internet **24**, which routes the request to an appropriate network resource (e.g., web server **26**), as depicted by a transfer path **40**.

Typically, the URL will correspond to a web page whose content is stored by the web server in an HTML (HyperText Markup Language) document comprising HTML code and embedded text content, in addition to other optional content languages, that may contain references to other objects (e.g., HTML documents and graphic image files) stored locally to the server or stored on a remote server. For example, the HTML content corresponding to a single-frame web page is often stored in a single file, while multiple-frame web pages may comprise content that is stored in a single file or in multiple files. These files may be stored locally on the web server (e.g., on one of the server's hard disks), or on a local storage device connected to the web server via a local area network (LAN), such as a network attached storage (NAS) filer. Optionally, some of the web page's content may comprise one or more documents that are stored at remote locations that may be accessed via a WAN (wide area network) or the Internet.

HTML is a standardized language that describes the layout of content on a web page, and attributes of that content. This layout and attribute information is defined by sets of tags contained in HTML code corresponding to the page. The tags define various HTML layout and display information, including tables, paragraph boundaries, graphic image positions and bounding box sizes, typeface styles, sizes, and colors, borders, and other presentation attributes. A portion or all of a web page's text content may be contained in the parent HTML document corresponding to the URL. In addition to basic HTML, web page documents may contain XML (eXtensible markup language) code, as well as scripting language code, such as javascript. However, for simplicity, any documents containing web page content other than only graphic content that are discussed herein will be referred to as HTML documents.

In addition to HTML and other markup and scripting language content, it is very common for web pages to include graphical content. In general, graphical content is usually stored in an image file or files that are external from the parent HTML document for the web page. For example, the parent HTML document may contain one or more embedded image tags that reference the location where those images are stored.

As before, the graphic images may be stored locally, or may be stored on remote servers that are accessed by the web server via a WAN, or the Internet. These files will typically comprise data stored in one of several well-known graphic formats, including bitmap files (BMP), GIF (Graphics Interchange Format) files, and JPEG (Joint Photographic Experts Group) files.

In response to receiving the request for content, web server **26** begins sending a parent HTML document **42** back to proxy server **32** in a block **104**. In a block **106**, the HTML content of the parent HTML document is parsed to search for references to external objects such as HTML frames and graphics. In a decision block **108**, a determination is made to whether any references are found. For each reference to an external object that is found, proxy server **32** requests to have the object retrieved from an appropriate network resource (e.g., a web server) in a block **110**, and data corresponding to the object is transmitted back to the proxy server, as depicted by locally accessible HTML documents **44** and graphic images **46**, as well as remotely accessible HTML documents **48** and graphic images **50**, which may be accessed via web server **28**. If the external object is a graphic image, there is no further processing of the object at this point. If the object is an HTML document, the functions provided by blocks **106** and **108** are repeated. Generally, this set of processing functions is repeated iteratively until all of the external objects are retrieved. However, as described below, there will be some instances in which certain objects will be retrieved at a later point in time. In addition to content stored on web servers that are accessed using HTTP, content may also be retrieved from various network sites using the File Transfer Protocol (FTP), such as FTP documents **51**, which are accessed via FTP server **30**.

In general, HTML documents and graphic files will be sent as packetized data streams using HTTP over one or more TCP/IP network connections, wherein the data streams will usually be asynchronous. Retrieval of HTML documents and graphic files corresponding to the embedded references will usually require additional transfer time. Furthermore, graphic content oftentimes comprises significantly larger file sizes than HTML content, leading to significant transfer times in some instances. For simplicity, the transfer of the various HTML documents and graphic files for the content request are depicted by HTML documents **52** and graphic documents **54**, which are transferred over a transfer path **56**.

When the HTML documents and graphic content are received by proxy server **32**, a scalable vector representation of the web page is generated in a block **114** by an HTML translator **58**. In brief, HTML translator **58** translates HTML, XML, and cascaded style sheet (CSS) layout content into a scalable vector representation, such as SVF. Details of the HTML translation process are contained below. In addition, the graphic images are converted into a compressed bitmap format in a block **116** by a graphics translator **60**. The vectorized content **62** and compressed bitmaps **64** are then streamed back to the client (i.e., computer **18**) in a block **118**, as depicted by a transfer path **66**. In one embodiment, the content portions are sent in separate streams using multiple connections. In another embodiment, the content portions are sent via a multiplexed stream using a single connection. As the vectorized content and compressed bitmap data are received by the client device, they are processed by a thin client **68** running on the client device, whereby a representation of the original web page content may be rendered on the client device's display screen at various user-selectable scaled resolutions and pan offsets in a block **120**, thereby

enabling a user to more clearly see an overview or details in the web page. Further details of the client side processing are provided below.

As discussed above, wireless clients may also access the vectorized network (e.g., web site) content provided via proxy server 24. The majority of this process is identical to that described above for land-line clients (e.g., computers 18, 20, and 22), except for provisions required for sending data to and receiving data from wireless devices. In general, most wireless devices will access the Internet via a wireless service provider (i.e., a wireless telecommunications carrier) that is particular to that wireless device. Accordingly, a portion of the transmission path to and from proxy server 24 will comprise infrastructure provided by that service provider and/or shared with other service providers. For simplicity, this infrastructure is shown as a cellular tower 70 and a service provider data center 72, although it will be understood by those skilled in the art that the connection path may comprise additional infrastructure components, including appropriate gateways and routers, that enable wireless devices to access proxy server 24.

In some implementations, there will be no special formatting/protocol services that need to be performed by proxy service 24—from the viewpoint of the proxy service, it will be immaterial whether the client is a land-based or wireless client; the special handling provisions for wireless devices will be handled entirely by the service providers infrastructure transparently at both ends of the communications path. In other instances, it may be desired or necessary to reformat the data content delivered to the wireless device at the proxy service. This will generally be dependent on the particular wireless protocol used, and what services are provided by the service provider for the wireless client.

Currently, in the United States, wireless clients generally access Internet 24 by using the Wireless Application Protocol (WAP). In Japan, the most popular access means is NTT DoCoMo's i-Mode wireless protocol. In addition to these wireless standards, new standards are anticipated to be in force in the near future, including NTT DoCoMo's FOMA (Freedom of Mobile Multimedia Access), which is transported over W-CDMA (Wideband Code Division Multiple Access), and CDMA-2000. For the purposes of the invention herein, it will be understood that those skilled in the mobile telecommunications arts will be knowledgeable about any particular format and/or transport protocol requirements that pertain to the particular protocol that is to be used.

A second exemplary system infrastructure 10B for implementing the invention is shown in FIG. 1B. As will be readily recognized, much of infrastructure 10B is similar to infrastructure 10A; however, rather than have a separate proxy server perform the proxy functions (retrieve and translate content), these functions are performed on machines operated by the web site in infrastructure 10B.

The logic implemented by the invention when providing content to a client using infrastructure 10B is illustrated in the flowchart of FIG. 2B, wherein the process begins in a block 101 in which the client sends a content request 39 directly to the network site (e.g., web server 26), as depicted by a transfer path 41. In a block 103, HTTP negotiations are performed to determine the format the content is to be delivered in. For example, the request may contain indicia identifying the type of content requested, such as an SVF MIME type (e.g., image/vnd.svf). This is to inform the web server that the request is for specially-formatted content rather than conventional content. The server first checks to see if it already has cached the requested content. If it has, it sends the content to the requesting client; otherwise, it retrieves the parent HTML document

in a block 107. It then performs processing steps in blocks 107, 109, and 111 to retrieve content referenced through embedded tags in a manner substantially similar to that discussed above with reference to respective blocks 106, 108, and 110. The primary difference in this instance is that the web server does not receive requests from or send documents to a proxy server—rather, the content is retrieved and processed at the web server, wherein the retrieved content may be stored local to the web server or retrieved from a remote server in a manner similar to that described above.

As before, the retrieved HTML documents are translated into scalable vector representations by HTML translator 58 in a block 114, while the graphic images are translated into a compressed bitmap format by image translator 60 in a block 116, as depicted by vectorized content 62 and bitmap content 64. The vectorized content and bitmap content are then streamed from the web server to the client in a block 119, as depicted by a transfer path 67. Upon arriving at the client, the vectorized content and bitmap content are processed, scaled, and rendered on the client in a block 120.

A third exemplary system infrastructure 10C for implementing the invention is shown in FIG. 1C. In this configuration, the proxy functions are performed at the client. As shown by a block 113 in FIG. 2C, the process for providing vectorized content to a client in accordance with infrastructure 10C begins in a block 113, in which the client sends a content request 37 to a network site, such as web server 26, via Internet 24. In response, the network site retrieves the parent HTML document and sends it to the requesting client in a block 115. In a manner similar to that discussed above with reference to blocks 106, 108, and 110 of FIG. 1A, the client first parses the parent HTML document searching for embedded references to external objects and retrieves these objects, whereupon the embedded reference search is performed on the newly retrieved document until all of the content corresponding to the original content request has been retrieved. This content is depicted by HTML documents 52 and image files 54, which are sent from the network site to the client via a transfer path 69. At this point, the client performs translations on the HTML content and the graphic image content that are substantially similar to that performed by the proxy server in FIG. 1A or at the web site in FIG. 1B, as provided by blocks 114 and 116. The vectorized and image content is then processed and scaled by thin client 68 in a block 120, as depicted by device output 71.

Attention now is focused on the functionality provided by proxy server 24 in system infrastructure 10A of FIG. 1. Fundamentally, the proxy server functions as a proxy. It accepts requests for content from client devices as full URLs using standard HTTP mechanisms carried over a multiplexed TCP connection. Standard HTTP content negotiations features specify the formats in which content is to be delivered (SVF, bitmap, and possibly others, which can be handed off to cooperating client-side display software). As described in further details below, in some embodiments the proxy server appears for the client as a normal proxy (that is, the client knows it is retrieving content via the proxy), while in other embodiments the proxy is transparent to the client.

The proxy server responds to client content requests by delivering content in one of the requested formats, by retrieving the content in an appropriate format from its cache, or from an upstream content source (again using standard HTTP content negotiation features), or by translating upstream content from a supported original format to SVF or the client bitmap format.

Requests from the server installation to its cache and from the cache to upstream content sources are made in HTTP

carried over TCP using simple straightforward Web content requests. For example, requests from clients to the proxy server comprise HTTP proxy requests (e.g., "GET http://www.xyz.com/some_page.html HTTP/1.0 . . .") carried over TCP or over a lightweight multiplexing protocol over TCP. The multiplexing protocol allows the server to push image thumbnails to the client before the SVF stream is available, as well as offering a channel for control and status information, more simultaneous channels than the client operating system may support, and a mechanism for prioritizing information flow from server to client under loose client control. In addition to HTTP requests, the proxy server architecture supports other user-level protocols, such as FTP and Gopher.

Details of some of the primary components of the proxy server architecture are shown in FIG. 3. Internally, the proxy server comprises a suite of coordinated processes connecting to upstream content through an HTTP cache 74. In one embodiment all functions except caching are performed in a single process, wherein multiple threads are used to effect asynchronous I/O. Separate processes communicated via persistent multiplexed connections carried over the most efficient reliable transport available (e.g., Unix sockets over single processor and symmetric multiprocessor (SMP) computers; TCP sockets between separate computers). All processes are capable of servicing multiple requests simultaneously. No process maintains client state outside the context of a single request, so all components can be repeated and load balanced across multiple CPU's of an SMP computer or across separate computers on a LAN.

The various content translators used by the proxy server accept (via HTTP PUT) or request (driven by HTTP proxy GET/POST) content in supported, but client-unsupported, formats; and return (via HTTP PUT or GET/POST response) one or more representations of that content in a client-supported format. In the embodiments illustrated in FIGS. 1A-C, two translators are used: HTML translator 58 and image translator 60. Future content types may be accommodated by new translators, by extending existing translators to cover the new content types, or by extending the client's capabilities. Standard HTTP content negotiation mechanisms are used to inform the proxy server of the client's capabilities and expectations on each request.

Managers at the proxy server coordinate the operations of other components. Two managers are presently defined; a client manager 73 that handles client proxy requests, and a request manager 75 that handles unproxied HTTP requests from other services. The managers accept requests, attempt to service them from HTTP cache 74, and drive HTML translator 58 and image translator 60 when content does not match

the clients' requirements. Managers also handle translator requests for inline content (e.g., image dimensions for page layout), and push translated content into HTTP cache 74. Additionally, the client manager coordinates delivery of primary and inlined content, and provides process and status information to the clients.

As discussed above, HTML translator 58 creates a scalable vector representation of the original HTML content of a requested web page. In order to better explain how translation of HTML content is performed, one embodiment of a translation process is described below as applied to an exemplary web page. In addition, details of conventional web page client and server-side processing are provided so as to clarify how web content is laid out during a pre-rendering process on the client.

FIG. 4 shows a representation of a web page 210 served from an exemplary stock brokerage Internet web site as it would appear when rendered on a modern Internet browser, such as Microsoft's Internet Explorer or Netscape's Navigator. Web page 210 is exemplary of many web pages that implement frames, and includes two adjacent frames 212 and 214. A logo graphic object 216A is displayed at the top of frame 212, which additionally includes a "MARKETS" text header 218A, an "INVESTMENTS" text header 220A, and a plurality of links with overlaying graphic objects, including a "DOW" link 222A, a "NASDAQ" link 224A, an "OPTIONS" link 226A, a "CHARTS" link 228A, a "MUTUAL FUNDS" link 230A, a "IRA, 401K OPTIONS" link 232A, and a "TAX INFORMATION" link 234.

A horizontal group of links 236 is disposed at the top of frame 214, and includes a "QUOTES" link 238A, a "HOT PICKS" link 240A, a "CALENDARS" link 242A, and a "NEWS" link 244A. An advertisement banner 246A is displayed just below the horizontal group of links and just above a "NEWS SPARKS MARKET" headline 248A. Frame 214 also includes a pair of graphic image objects, including a DOW chart 250A and a NASDAQ chart 252A. A set of user input objects is disposed adjacent to DOW chart 250A within a graphic object 254A, including an "ACCOUNT #" input box 255A, an "ACCESS CODE" input box 256A, and a "LOGIN" button 257A. In addition to the foregoing objects, frame 214 also includes text objects 258A and 260A.

An HTML listing corresponding to web page 210 is presented below as LISTING 1. Note that LISTING 1 sometimes refers to object descriptions and link paths rather than the text or path location of actual objects for simplicity, and that other elements commonly found in HTML pages, such as META entries, are omitted for clarity.

LISTING 1

```

1. <html>
2. <head><title>"MARKET HOME"</title></head>
3.
4. <body bgcolor="#FFFFFF" link="#0033CC" vlink="#0033CC">
5.
6. <frameset cols="25%,75% frameborder=0 border=0>
7. <frame>
8. <align=left><align=top>
9. 
10. <br><br>
11. <t3>TEXT HEADER #1 align=left</t3><br>
12.
13. <table width="90%" border=0 cellpadding=10 cellspacing=10 bgcolor="#000000"
14. align=center>
15. <tr>
```


-continued

LISTING 1

```

16.      <a href="URL or path for LINK #5" </a>
18.      <tr>
19.      <a href="URL or path for LINK #6" </a>
21.      <tr>
22.      <a href="URL or path for LINK #7" </a>
24.      <tr>
25.      <a href="URL or path for LINK #8" </a>
27. </table>
28.      <br>
29.      <t3>TEXT HEADER #1 align=left</t3>
30.      <br>
31. <table width="90%" border=0 cellspacing=10 cellpadding=0 bgcolor="#000000"
32. align=center>
33.      <tr>
34.      <a href="URL or path for LINK #9" </a>
36.      <tr>
37.      <a href="URL or path for LINK #10" </a>
39.      <tr>
40.      <a href="URL or path for LINK #11" </a>
42. </table>
43. </frame>
44. <frame>
45. <table>
46.      <tr>
47.      <table width="100%" border=0 cellspacing=15 cellpadding=0
48.      bgcolor="#000000" align=center>
49.      <tr>
50.      <td><a href="URL or path for link#1"> alt="QUOTES"</a>
51.      <td><a href="URL or path for link#2"> alt="HOT PICKS"</a>
52.      <td><a href="URL or path for link#3"> alt="CALENDERS"</a>
53.      <td><a href="URL or path for link#4"> alt="NEWS"</a>
54. </table><br>
55.      <br>
56.      
58. <br><t1>HEADLINE TEXT</t1>
59. <table>
60.      <Colgroup span="2">
61.      <Col width = "400" align="center">
62.      <Col width = "200" align="center">
63.      <tr><td>
64.      
66.      <td>
67. /* INPUT FOR ACCOUNT NUMBER AND ACCESS CODE */
68. <SCRIPT LANGUAGE = "Javascript">
69. <!--
70.      [Javascript variable declarations]
71.      [Javascript functions to enable login] ---!>
72. </SCRIPT>
73.      <table>
74.      <td>
75.      
76.      <table width="150" height="25">
77.      <tr>
78.      <td>
79.      <font size=-2 face="arial, helvetica, verdana">Account #</font>
80.      <tr><input type="text" name="USERID" maxlength=9 size=20>
81.      <tr><font size=-2 face="arial, helvetica">Access Code:</font>
82.      <tr><input type="password" name="PASSWORD" maxlength=10 size=20
83.      onKeyDown="SuppressEnterBell(event)"
84.      onKeyPress="SuppressEnterBell(event)"
85.      onKeyUp="SubmitOnEnter(event)">
86.      <br>&nbsp;
87.      <br><input type="button" value="Login"
88.      OnClick="ProcessForm( )">&nbsp;&nbsp;&nbsp;<input type="reset">
89.
90.

```

-continued

LISTING 1

```

91.          <br>&nbsp;
92.          </td>
93.        </table>
94.      </table>
95.    <tr>
96.      
98.    </tr>
99.      <p>TEXT FOR TEXT OBJECT #1</p><br>
100.     <p>TEXT FOR TEXT OBJECT #2</p>
101.   </table>
102. </frame>
103. </frameset>
104. </html>

```

Web page documents comprise HTML code that is parsed, interpreted, and rendered by a browser. An HTML document comprises a plurality of HTML “markup” elements (tags) with corresponding attributes, that are used to describe the layout and formatting of various objects, including plain text and graphic objects, embedded between tag pairs. Exemplary elements include text tags (e.g., `` for bolding text), links (e.g., ``), formatting (e.g., `<p></p>` for creating a new paragraph, graphical (e.g., ``), wherein “name” defines an absolute or relative location at where an image is stored, tables (e.g., `<table></table>`) creates a table, and forms (e.g., `<form></form>` creates all forms).

As of Netscape Navigator 3.0 (and other later browsers), web pages could include frames. When using frames, the display page is divided into multiple framed areas. Framing enables a single display page to include source code from several HTML documents (one for each frame) or optionally, enables a single document to include more complicated grouping of contents whereby different content groups are contained in separate frames. Frames are commonly found on the web pages at sites that display a great deal of text and graphical content, such as MSN.com, ESPN.com, and USA-Today.com.

With reference to the flowchart of FIG. 5, the process for translating the HTML content into a scalable vector representation proceeds as follows. The process is initiated when the proxy server receives the HTML corresponding to the parent document (and frame documents, if appropriate), whereupon a pre-rendering parsing of the HTML is performed to determine where to place the various objects on the display page in a block 150. For example, elements such as tables, column definitions, graphic images, paragraphs and line breaks are identified. If frames are included, each frame is examined in the sequential order it appears in the HTML document, or the order in which the HTML documents corresponding to the frames in a frameset are downloaded to the browser. During further processing, the actual objects are rendered in their respective positions. Some of these objects are rendered almost immediately, such as plain text, while other objects, such as graphic objects, must first be retrieved prior to being fully-rendered. With respect to tables, there are some instances in which all of the objects corresponding to the cells in the table must be retrieved prior to rendering any of the table, while a well-designed table can be rendered incrementally. For example, by using Column grouping, the format of the corresponding table can be quickly determined by the browser. In some instances, one or more bitmaps may actually need to be fetched before the page layout can be determined.

Next, in a block 152, the content is separated into objects based on logical groupings of content portions and a page layout is built using bounding boxes that are produced for each object. As the primary HTML document is parsed, logical groupings of content will emerge. For instance, text content contained within paragraph tags `<p></p>` forms a logical grouping of text content. In essence, a logical grouping means the content should appear together as a logical group, such as within a substantially rectangular outline, in the rendered page. Other logical groupings include frames, table content, row content, single line entries such as headlines and headers, and user-interface objects, as well as graphic layout objects, such as separator bars, and graphic images. In addition to logically grouping content into objects, a “bounding box” is defined for each object. In general, the bounding box defines an outlined shape within which the content (text or graphic image) will appear. In most instances, the bounding box will be substantially rectangular in shape. However, bounding boxes comprising more complex shapes may also be produced.

In further detail, the following explains how objects corresponding to graphic images are produced. In HTML, objects comprising graphic content are identified by an `` or `<object>` or other tags. In the foregoing tag, local graphic images are typically stored on the same server as the web page, or another computer that is local to the site’s server, and generally are located through a local directory path (absolute or relative to the location of the present page) that points to the graphic image file. Remote images are those images that are stored on servers at sites that are remote to the web server. For example, with reference to LISTING 1, when the parser encounters line 9, the browser identifies that data comprising a graphic image corresponding to logo graphic object 1 will be arriving (or may have already been received), and the displayed image is to have a height of 80 pixels and a width of 100 pixels. The location of each object on a display page will be dependent on previous HTML layout elements, such as tables, paragraphs, line breaks, and other graphic objects. The size and location of the other graphic objects (i.e., graphic objects #2-12) on the page are determined in a similar manner. The HTML code for these objects are shown in lines 16, 19, 22, 25, 34, 37, 40, 59, 67, 78 and 96, respectively. As identified in the HTML code, data corresponding to graphic objects #9 (advertisement banner 46A) is forwarded to the browser from an external site (as indicated by the URL to GRAPHIC #9), while graphic objects 1-8 and 10-12 are sent from the web site the parent HTML document is sent from.

In a similar manner, the foregoing technique is applied to the HTML code in the primary document to identify other types of objects as well. In addition to parsing the primary HTML document, similar processing is performed on referenced documents, such as documents that include frame content that is defined and stored separate from the primary HTML document.

A representation of the results of the functions performed in block 152 are shown in FIG. 4B. In the Figure, objects corresponding to the original content of FIG. 4A are shown with an appended "B" that is added to each object's root reference number, wherein the root reference number for an object is that same as the logically grouped content in FIG. 4A that it corresponds to, e.g., an object 248B is generated for "NEWS SPARKS MARKET" headline 248A, etc.

Next, in a block 154, the page layout is defined based on the bounding boxes. In actuality, generation of the page layout information is performed in conjunction with defining the boundary boxes for the objects, wherein the location of a given object is based on the location of other related (e.g., if within a table) or non-related objects corresponding to HTML content that have been previously parsed. For example, the location of a given paragraph will depend on the other content for the page that are listed prior to the definition for the paragraph in the primary HTML document or referenced document, if applicable. As the HTML content of the primary and any referenced HTML documents are parsed, the page layout is generated based on the various HTML tags and the content embedded between tag pairs and/or referenced by a tag pair statement (e.g., graphic images).

As will be recognized by those skilled in the art, the functions performed in blocks 150, 152, and 154 are commonly performed by conventional browsers during a pre-rendering process. In some browsers, these functions are performed by the Mozilla rendering engine, which comprises open source software that is readily available for use by developers. At present, the software for the Mozilla rendering engine may be accessed via the Internet at www.mozilla.org. Accordingly, in one embodiment, the present invention uses core functionality provided by the Mozilla rendering engine source code to perform the functions of block 150, 152, and 154.

At this point, the present invention deviates substantially from the prior art by using the various object layout data generated during the pre-rendering process to generate a scalable vector representation of the original page content. First, in a block 156, a datum point is defined for the page and the bounding box for each object. For example, as shown in FIG. 4C, a rendered page datum 262 is defined to be coincident with the upper left hand corner of the display frame of the rendered page for the web page. Generally, any point on the page may be used as the page datum—the only requirement is that the page datum that is selected is used consistently throughout the process. The use of the upper left hand corner of the display frame is advantageous since the location of the first object encountered in the HTML code for a page is located relative to this corner.

In general, the datum points for each object may also be located any place on the object, as long as the object datum points are used in a predictable manner. For example, as depicted in FIG. 4C, various datum points for corresponding objects are defined to be coincident with the upper left hand corner of the bounding box for that object, wherein the object's datum point shares the root reference number of the object with an appended "C."

Once the page's datum point and an object's datum point are known, a vector between these points is generated for each object in a block 158. With reference to FIG. 4D, in one

embodiment, wherein the page datum point corresponds to the upper left and corner of the display frame and is assigned an XY value 266 of 0,0, the vector for a given object may be stored as the XY value of the datum point of that object relative to 0,0, such as a value of 150, 225 (ref. num. 268) for a vector 250D pointing to an object datum 250C, and a value of 150, 425 (ref. num. 270) for a vector 252D pointing to an object datum 252C. In another embodiment, each vector may be stored as XY data relative to a 0,0 datum point corresponding to the upper left hand corner of the frame the object belongs to. For example, a vector 250D' from a frame datum 214D to object datum 250C is stored as 20, 200 (ref. num. 268'), while a vector 252D from frame datum 214D to object datum 252C is stored as 20, 425. In this embodiment, offset information for each frame relative to a known datum will also be stored, as depicted by a vector 214D.

The scalable vector representation is completed in a block 160, wherein a reference is created for each object that includes or links an object's content and attributes, such as object type (e.g., text, image), object typeface, and boundary box parameters, to the object's vector. For example, object 250B is a graphic image having a vector 250D and a bounding box that is 180 pixels high and 350 pixels wide, while object 252B is a graphic image having a vector 252D and a bounding box that includes a height of 200 pixels and a width of 350 pixels. This enables client-side operations to be performed that only initially consider the vectors, wherein if it is determined that a vector's endpoint (and/or the bounding box corresponding to the object the vector points to) would appear off of a display, there is no need to retrieve the content and attribute data linked to the vector. This concept is explained in further detail in the following section.

It is noted that a portion of the display content produced on a client device will never contain any rendered content, as this portion is reserved for the browser's user interface. In WINDOWSTM environments, this portion will include the browser's window frame, as well as the pulldown and icon menus provided in the browser's user interface, which are depicted by a box 264 in the Figures herein.

Client-Side Software and Processing

As discussed above, the present invention supports a wide variety of clients, including land-based clients and wireless clients. Each client requires some client-side software that enables the scalable vector content data provided to it to be rendered at a user-selectable scale factor and offset on the client's display, such as a monitor or built-in LCD screen.

By enabling original content from a web site to be displayed in such a resolution-independent manner, users will be able to view content in a manner that did not previously exist, greatly enhancing the user experience. For example, in some implementations the client may be a personal computer (PC). Using a least-common denominator approach, many web pages are designed for a smaller resolution (for example 640×480 pixels, a minimum resolution commonly supported by nearly all PC's, including legacy PC's) than the resolution provided by the video output capabilities available with many of today's PC's, such as 1024×768 pixels, 1280×1024 pixels, and even 1600×1200 pixels. As a result, when these web pages are displayed on a high-resolution display, they occupy only a portion of the display, making portions of the pages, especially those portions containing small text, difficult to read. By enabling users to selectively magnify the entire page, these design flaws are easily overcome. Alternatively, the client may be a small device, such as a hand held computer or a cell phone, which has a smaller display resolution than common Web pages are designed for. As explained below, through use of the invention's scalable vector representation

and client-side processing, users are enabled to view the entire content of billions of existing Web pages using handheld devices in a simple and reasonable way.

In one embodiment, the client software may be a plug-in to a Web browser, such as Netscape Navigator or Microsoft Internet Explorer. Such a plug-in might have the browser download the data and display it in a sub-window of the browser. Alternatively, the client software may be a Java applet running in a browser. As another option, the client software may be a stand-alone program that interfaces with the proxy server or proxy software directly. The client software may bypass the proxy when requesting information that won't be translated to vectors, such as bitmaps.

With reference to FIG. 6, client-side processing proceeds in the following manner. In a block 160, the vector representation data (i.e., vectorized HTML content and compressed bitmap content) for the web page is gathered at the client. Typically, this data will be stored in a cache at the client as it is being received, and the client simply retrieved the data from the cache. In a block 162, a display list of vectors is built. This process is well known in the CAD arts, and is enabling rapid zooming of vector-based objects. In a block 164, user selectable scale and offset (pan) values are determined. Based on various user interactions with the user-interface of the client, the user is enabled to control the zoom (size) and offset of the rendered page. For example, suppose the user provides zoom and offset inputs to produce a rendered page 210E, as shown in FIG. 4E. In this rendered page, the original origin is now off of the screen (the page image is shifted upward and toward the left—see FIG. 4F), and the view has been scaled approximately 1.3 times.

Next, in a block 166, the vectors and boundary boxes are processed based on the scale and offset, and a bounding box defining the limits of the display content is determined. The results of this step are shown in FIG. 4F, while FIG. 4G shows specific details on how the vectors and bounding boxes corresponding to image objects 250B and 252B (now 250B' and 252B', respectively) are processed. Logically, there are generally two ways to scale and offset the rendered content. In one embodiment, vectors and bounding boxes are mapped to a virtual display area in memory that has much greater resolution (e.g., 100,000×100,000 pixels) than any real display, and a virtual display limit bounding box is scaled and moved around over the virtual display area. Accordingly, during subsequent processing described below, objects falling within the display bounding box are rendered by reducing the scaling of those objects in the virtual display to how the objects will appear on the client device display relative to the virtual display bounding box. In the alternate, a fixed reference frame corresponding to the display resolution of the client device screen is maintained, wherein all vectors and bounding boxes are scaled and offset relative to the fixed reference frame. Each scheme has its advantages and disadvantages. One advantage of the second method is that the display bounding box is always maintained to have a size that matches the resolution of the content display area on the client device.

As shown in FIG. 4G, respective offsets in X and Y, ($-\Delta X$ and $-\Delta Y$ in the Figure) are applied to the starting point of each of the vectors. The vectors are then scaled by a scale factor "SF." The results of the new vectors are depicted by vectors 250D' and 252D'. This produces a new datum for each object's bounding box that is relative to rendered page datum 262, which remains fixed. As discussed above, only a portion of the display screen will actually be used to display content (as defined by a display limit bounding box 266 in this embodiment), while other portions of the screen, including

box 264, will comprise a generally fixed-size user interface. Accordingly, rendered page datum 262 is not located at the upper left hand corner of the display area, although it possibly could be located at this point when either the current user interface is inactive (i.e., the display portion of the user interface is temporary disabled) or the user interface is contained in other portions of the display.

This foregoing process establishes a starting point (the new datum) for where the content in each object's bounding box will be rendered. At this point, each object's bounding box is then drawn from its new datum using the scaling factor. For example, in the original web page 210D (FIG. 4D), bounding box 250B had an X-axis datum of 150 pixels, a Y-axis datum of 225 pixels, and a height and width of 180×350 pixels. In contrast, after being offset and scaled, bounding box 250B' has an X-axis datum of $150*SF-\Delta X$, a Y-axis datum of $225*SF-\Delta Y$, and a height and width of $180*SF\times 350*SF$.

Returning to the flowchart of FIG. 6, once the vectors and bounding boxes are offset and scaled, content corresponding to objects having at least a portion of their bounding boxes falling within the display limit bounding box is retrieved from the client device's display list in a block 168. For examples, as shown in FIG. 4F, content corresponding to all of the objects except for those falling entirely outside of display limit bounding box 266 (objects 216, 238, 240, 242 and 244) is retrieved from the display list. That content is then scaled in a block 170. For image content, this comprises decompressing and scaling the compressed bitmaps corresponding to those images. For text content, this comprises scaling the font (i.e., typeface) that the text content portions of the web page are written in the parent HTML document and any referenced documents. There are various techniques for typeface scaling that may be implemented here, depending on the available resources provided by the operating system of the client device. For example, for WINDOWS™ operating systems, many TRUETYPE™ fonts are available, which use a common scalable definition for each font, enabling those fonts to be scaled to just about any size. In other cases, such as current PDA (e.g., Palm Pilots) operating systems, there is no existing feature that supports scaling fonts. As a result, bitmapped fonts of different font sizes and styles may be used. In addition to scaling image and text content, other types of content, such as separator lines and borders may also be scaled by block 170.

The process is completed in a block 172, wherein those portions of the scaled content falling within the display limit bounding box are rendered on the client device's display.

As discussed above, it is foreseen that the invention will be used with client devices having small, low resolution displays, such as PDAs and pocket PCs. Examples of various views of an exemplary web pages obtained from the YAHOO™ web site are shown in FIGS. 7A-B, 8A-B and 9A-B. For instance, FIG. 7A represents how the YAHOO™ home page might appear on a Palm IIIc color PDA.

In addition to directly scaling and offsetting content, the client user-interface software for PDA's provides additional functionality. For instance, a user may select to view a column (results represented in FIG. 7B by tapping that column with a stylus, a shown in FIG. 7A. Similarly, the user may select to zoom in on an image by tapping the image with the stylus, as shown in FIGS. 8A and 8B, or select to view a paragraph in an article by tapping on the paragraph, as shown in FIGS. 9A and 9B. It is noted that in some instances, the display of the paragraph may be reformatted to fit the characteristics of the display, rather than following the original format in the zoom-out view.

21

It is further noted that that different scaling factors can be applied to the X and Y axis so as to change the aspect ratio of the display. For example, a Web page may be designed to be displayed on a computer having a resolution of 800x600 pixels, or a 4X to 3Y aspect ratio. In this case, the display corresponds to a "landscape" layout, wherein there are more pixels along the X axis than along the Y axis. Conversely, many handheld devices display images having a "portrait" layout, wherein there are more pixels along the Y axis than the X axis. By enabling different scaling factors to be applied to the X and Y axes, the present invention enables the aspect ratio of a rendered display image to be adjusted to better fit the aspect ratio of the client device.

An Exemplary Computer Architecture

An exemplary machine in the form of a computer system 500 in which features of the present invention may be implemented will now be described with reference to FIG. 10. Computer system 500 may represent a workstation, host, server, print server, or printer controller. Computer system 500 comprises a bus or other communication means 501 for communicating information, and a processing means such as processor 502 coupled with bus 501 for processing information. Computer system 500 further comprises a random access memory (RAM) or other dynamic storage device 504 (referred to as main memory), coupled to bus 501 for storing information and instructions to be executed by processor 502. Main memory 504 also may be used for storing temporary variables or other intermediate information during execution of instructions by processor 502. Computer system 500 also comprises a read only memory (ROM) and/or other static storage device 506 coupled to bus 501 for storing static information and instructions for processor 502.

A data storage device 507 such as a magnetic disk or optical disc and its corresponding drive may also be coupled to bus 501 for storing information and instructions. Computer system 500 can also be coupled via bus 501 to a display device 521, such as a cathode ray tube (CRT) or Liquid Crystal Display (LCD), for displaying information to an end user. Typically, an alphanumeric input device 522, including alphanumeric and other keys, may be coupled to bus 501 for communicating information and/or command selections to processor 502. Another type of user input device is cursor control 523, such as a mouse, a trackball, or cursor direction keys for communicating direction information and command selections to processor 502 and for controlling cursor movement on display 521.

A communication device 525 is also coupled to bus 501. Depending upon the particular presentation environment implementation, the communication device 525 may include a modem, a network interface card, or other well-known interface devices, such as those used for coupling to Ethernet, token ring, or other types of physical attachment for purposes of providing a communication link to support a local or wide area network, for example. In any event, in this manner, the computer system 500 may be coupled to a number of clients and/or servers via a conventional network infrastructure, such as a company's Intranet and/or the Internet, for example.

Importantly, the present invention is not limited to having all of the routines located on the same computer system. Rather, individual objects, program elements, or portions thereof may be spread over a distributed network of computer systems. Additionally, it is appreciated that a lesser or more equipped computer system than the example described above may be desirable for certain implementations. Therefore, the configuration of computer system 500 will vary from imple-

22

mentation to implementation depending upon numerous factors, such as price constraints, performance requirements, and/or other circumstances. For example, according to one embodiment of the present invention, a cell phone or a hand held computer may comprise only a processor or a micro controller and a memory, such as a micro code ROM or RAM, for storing static or dynamically loaded instructions and/or data.

In the foregoing specification, the invention has been described with reference to specific embodiments thereof. It will, however, be evident that various modifications and changes may be made thereto without departing from the broader spirit and scope of the invention. The specification and drawings are, accordingly, to be regarded in an illustrative rather than a restrictive sense.

What is claimed is:

1. A mobile device, comprising:

- a processor;
- a wireless communications device operatively coupled to the processor, to facilitate communication with a network via which Web content may be accessed;
- a touch-sensitive display;
- a memory, operatively coupled to the processor; and
- storage means, operatively coupled to the processor, in which a plurality of instructions are stored that when executed by the processor enable the mobile phone to perform operations including,
 - enabling a user to request access to a Web page comprising HTML-based Web content defining an original page layout, functionality, and design of content on the Web page;
 - retrieving HTML-based Web content associated with the Web page;
 - translating the HTML-based Web content to produce scalable vector-based page layout information;
 - employing the scalable vector-based page layout information and/or data derived therefrom to,
 - render a view of at least a portion of the Web page on the touch-sensitive display using a first scale factor; and
 - re-render the Web page in response to associated user inputs to enable a user to iteratively zoom in and out views of the Web page on the display while preserving the original page layout, functionality, and design of the content on the Web page defined by the HTML-based Web content,
 - wherein preservation of the functionality defined by the HTML-based content includes preservation of hyperlink functionality.

2. The mobile device of claim 1, wherein the device comprises a mobile phone.

3. The mobile device of claim 1, wherein the device comprises one of a hand-held device or a palm-held device.

4. The mobile device of claim 1, wherein execution of the instructions performs further operations comprising enabling the user to zoom in on a user-selectable portion of a display of the Web page in response to a user interface input made via the touch-sensitive display.

5. The mobile device of claim 1, wherein the display of the Web page is re-rendered in real-time to effect zooming operations.

6. The mobile device of claim 1, wherein the Web content includes at least one hyperlink, and wherein execution of the instructions performs further operations comprising:

- enabling the user to select the hyperlink via the touch-sensitive display; and, in response thereto,

23

retrieving and translating HTML-based Web content associated with the hyperlink to produce additional scalable vector-based page layout information; and employing the additional scalable vector-based page layout information and/or data derived therefrom to render the Web content associated with the hyperlink on the touch-sensitive display.

7. The mobile device of claim 1, wherein execution of the instructions performs further operations comprising enabling the Web content to be displayed at different resolutions by scaling the scalable vector-based page layout information to resize a view of the Web page on the display in response to associated user inputs.

8. The mobile device of claim 1, wherein execution of the instructions performs further operations comprising returning the display of the Web content to a previous view in response to a corresponding user input made via the touch-sensitive display.

9. The mobile device of claim 1, wherein execution of the instructions performs further operations comprising enabling a user to pan a display of the Web content in response to a corresponding user input made via the touch-sensitive display.

10. The mobile device of claim 9, wherein execution of the instructions performs further operations comprising enabling the display of the Web content to be panned in real-time.

11. The mobile device of claim 1, wherein the page layout of the Web page is defined to have an original aspect ratio, and wherein the scalable vector-based page layout information and/or data derived therefrom is scaled to render a display having a different aspect ratio.

12. The mobile device of claim 1, wherein execution of the instructions performs further operations comprising enabling a user to view a column of the Web content at a higher resolution than a current resolution by tapping on the column via the touch-sensitive display, wherein in response thereto, the display is re-rendered such that content corresponding to the selected column is displayed to fit across the touch-sensitive display.

13. The mobile device of claim 1, wherein the Web content includes at least one image, and wherein execution of the instructions performs further operations comprising enabling a user to view an image at a higher resolution than a current resolution by tapping on the image via the touch-sensitive display, wherein in response thereto, the display is re-rendered such that the image is displayed to fit across a width of a display area of the touch-sensitive display.

14. The mobile device of claim 1, wherein execution of the instructions performs further operations comprising enabling a user to view a paragraph of the Web content at a higher resolution than a current resolution by tapping on the paragraph via the touch-sensitive display, wherein in response thereto, the display is re-rendered such that content corresponding to the selected paragraph is displayed to fit across a width of a display area of the touch-sensitive display.

15. The mobile device of claim 1, wherein execution of the instructions performs further operations comprising: generating a display list derived, at least in part, via use of the vector-based page layout information; and employing the display list to re-render the display of the Web page.

16. The mobile device of claim 1, wherein execution of the instructions performs further operations comprising:

parsing HTML-based code corresponding to the received Web content to logically group content into objects, the objects including a plurality of display objects;

24

defining a primary datum corresponding to a page layout; and,

for each display object,

defining an object datum corresponding to a layout location datum for the object's associated display content; generating a vector from the primary datum to the object datum for the object; and

creating a reference that links the object to its corresponding vector.

17. The mobile device of claim 16, wherein execution of the instructions performs further operations comprising: mapping the object vectors to a virtual display area in memory.

18. The mobile device of claim 1, wherein execution of the instructions performs further operations comprising:

parsing the HTML-based content to logically group content into objects;

generating page layout information including a bounding box for each object, the bounding box defining width and height dimensions for the object; and

storing information that links each object with its corresponding page layout information;

wherein the page layout information further includes information from which a page layout location of each of the bounding boxes can be determined.

19. The mobile device of claim 1, wherein the scalable vector-based content includes scalable text content, and wherein execution of the instructions performs further operations comprising scaling a scalable font to render the scalable text content.

20. The mobile device of claim 1, wherein at least a portion of the instructions comprise Java-based instructions configured to be executed on a Java virtual machine.

21. The mobile device of claim 1, wherein translating the HTML-based Web content to produce scalable vector-based page layout information comprises:

processing the HTML-based Web content with a rendering engine to generate page layout information corresponding to the original page layout as interpreted by the rendering engine; and

employing the page layout information to produce scalable vector-based page layout information.

22. The mobile device of claim 21, wherein the page layout information defines a layout location for a plurality of objects, including text objects, graphic layout objects, and/or image objects included on the Web page, and wherein producing vector-based page layout information comprises:

defining a primary datum corresponding to a page layout; and,

for each object,

defining an object datum corresponding to the layout location for the object on the page layout;

generating a vector from the primary datum to the object datum for the object; and

creating a reference that links the object to its corresponding vector.

23. The mobile device of claim 22, wherein execution of the instructions performs further operations comprising effecting a zoom operation combined with a pan operation by, for each of the plurality of display objects to be included in a panned view of the Web page to be rendered on the display,

scaling page layout information associated with the display object using a scale factor corresponding to a zoom level associated with the zoom operation to determine a scaled datum;

25

determining an offset corresponding to the pan operation and combining the scaled datum with the offset to produce a scaled and offset datum that defines a location where the display object is to be rendered on the panned view of the Web page;

scaling content associated with the display object using the scale factor; and

rendering the scaled content at the location defined by the scaled and offset datum to render the display object on the panned view of the Web page.

24. The mobile device of claim 23, wherein rendering scaled content associated with a text object comprises:

retrieving presentation attributes for the text object, the presentation attributes including a font typeface, size and color;

employing a scalable font associated with the font typeface to render text associated with the text object in a color associated with the color attribute, wherein the text is rendered relative to a location associated with the scaled and offset datum for the text object, and wherein the scale applied to the scalable font is a function of the scale factor and the font size.

25. The mobile device of claim 21, wherein the original format of the Web page defines a width for the Web page, as interpreted by the rendering engine, and wherein execution of the instructions performs further operations comprising:

determining an applicable scale factor to fit the width of the Web page across a display area of the touch-sensitive display; and

employing the scale factor that is determined as the first scale factor.

26. The mobile device of claim 1, wherein zooming operations are effected by applying a mathematical transformation to a plurality of points in a two-dimensional coordinate system comprising X and Y axes, including points comprising datum points having corresponding vectors included in the scalable vector-based page layout information defining page layout locations of corresponding text and image objects mapped to the two-dimensional coordinate system, wherein the mathematical transformation comprises,

$$X' = X * SF;$$

$$Y' = Y * SF;$$

wherein X, Y is the location of a point prior to transformation, X', Y' is the location of the point after transformation, and SF is the scale factor.

27. The mobile device of claim 26, wherein the mathematical transformation is applied to points in a first coordinate system comprising a virtual coordinate system associated with a virtual display area onto which page layout information is mapped to a second coordinate system comprising a device coordinate system corresponding to a pixel resolution of the display of the mobile device, wherein points are mapped from the first coordinate system to the second coordinate system using the mathematical transformation.

28. The mobile device of claim 1, wherein execution of the instructions performs further operations comprising maintaining at least one instance of the page layout information in a manner that is independent of the zoom levels used to view the web page on the display.

29. The mobile device of claim 1, wherein the HTML-based Web content includes cascading style sheet content defining layout and presentation attributes for the Web page.

26

30. A mobile phone, comprising:

a processor,

wireless communications means operatively coupled to the processor, to facilitate communication with a mobile service provider network via which Web content may be accessed;

a touch-sensitive display;

a memory, operatively coupled to the processor; and

storage means, operatively coupled to the processor, in which a plurality of instructions are stored that when executed by the processor enable the mobile phone to perform operations including,

rendering a browser interface via which a user is enabled to request to access to a Web page having an original format comprising HTML-based content defining an original page layout, functionality, and design of content on the Web page;

retrieving HTML-based content associated with the Web page;

translating at least a portion of the HTML-based content from its original format to produce translated content including scalable vector-based content that supports a scalable resolution-independent representation of the HTML-based content that preserves an original page layout, functionality and design of the at least a portion of the HTML-based content when scaled and rendered; and

employing the scalable vector-based content to render a view of at least a portion of the Web page on the display using a first scale factor,

wherein preservation of the functionality defined by the HTML-based content includes preservation of hyperlink functionality.

31. The mobile phone of claim 30, wherein execution of the instructions performs further operations comprising enabling the user to zoom in on a user-selectable portion of a display of the Web page in response to a user interface input made via the touch-sensitive display.

32. The mobile phone of claim 31, wherein the display of the Web page is re-rendered in real-time to effect zooming operations.

33. The mobile phone of claim 30, wherein the Web content includes at least one hyperlink, and wherein execution of the instructions performs further operations comprising:

enabling the user to select the hyperlink via the touch-sensitive display; and, in response thereto,

retrieving and translating the Web content associated with the hyperlink to produce additional scalable vector-based content; and

employing the additional scalable vector-based content to render the Web content associated with the hyperlink on the touch-sensitive display.

34. The mobile phone of claim 30, wherein execution of the instructions performs further operations comprising:

parsing and processing markup language code associated with the Web page to determine the original page layout of display content within the Web page, wherein the original page layout defines a layout location for a plurality of objects, including text objects, graphic layout objects, and/or graphic image objects included in the Web page;

defining a primary datum corresponding to the original page layout; and,

for each object,

defining an object datum corresponding to the layout location for the object;

27

generating a vector from the primary datum to the object datum for the object; and
creating a reference that links the object to its corresponding vector.

35. The mobile phone of claim 30, wherein execution of the instructions performs further operations comprising enabling the Web content to be displayed at different resolutions by scaling the scalable vector-based content to re-render the display in response to associated user inputs.

36. The mobile phone of claim 30, wherein execution of the instructions performs further operations comprising returning the display of the Web content to a previous view in response to a corresponding user input made via the touch-sensitive display.

37. The mobile phone of claim 30, wherein execution of the instructions performs further operations comprising enabling a user to pan a display of the Web content in response to a corresponding user input made via the touch-sensitive display.

38. The mobile phone of claim 37, wherein execution of the instructions performs further operations comprising enabling the display of the Web content to be panned in real-time.

39. The mobile phone of claim 30, wherein the page layout of the Web page is defined to have an original aspect ratio, and wherein the scalable vector-based content is scaled when rendered so as to produce a display having a different aspect ratio.

40. The mobile phone of claim 30, wherein execution of the instructions performs further operations comprising enabling a user to view a column of the Web content at a higher resolution than a current resolution by tapping on the column via the touch-sensitive display, wherein in response thereto, the display is re-rendered such that content corresponding to the selected column is displayed to fit across the touch-sensitive display.

41. The mobile phone of claim 30, wherein the Web content includes at least one image, and wherein execution of the instructions performs further operations comprising enabling a user to view an image at a higher resolution than a current resolution by tapping on the image via the touch-sensitive display, wherein in response thereto, the display is re-rendered such that the image is displayed to fit across at least one of a width and height of a display area of the touch-sensitive display.

42. The mobile phone of claim 30, wherein execution of the instructions performs further operations comprising enabling a user to view a paragraph of the Web content at a higher resolution than a current resolution by tapping on the paragraph via the touch-sensitive display, wherein in response thereto, the display is re-rendered such that content corresponding to the selected paragraph is displayed across at least one of a width and height of a display area of the touch-sensitive display.

43. The mobile phone of claim 30, wherein execution of the instructions performs further operations comprising:
generating a display list associated with the scalable vector-based content; and
employing the display list to re-render the display at different scale factors to enable rapid zooming of the Web page.

44. The mobile phone of claim 30, wherein execution of the instructions performs further operations comprising:

parsing and processing markup language code corresponding to the received Web content to determine page layout information corresponding to a page layout of the content on the Web page;

28

logically grouping selected content into objects;
defining a primary datum corresponding to the page layout;
and,

for each object,
defining an object datum corresponding to a layout location datum for the object's associated display content;
generating a vector from the primary datum to the object datum for the object; and
creating a reference that links the object to its corresponding vector.

45. The mobile phone of claim 44, wherein execution of the instructions performs further operations comprising:
mapping the object vectors to a virtual display area in memory.

46. The mobile phone of claim 45, wherein execution of the instructions performs further operations comprising:

determining a first scale factor and offset in response to one or more corresponding user inputs defining a user-selectable zoom level and pan corresponding to a rendered view of the Web content desired by a user;

determining a virtual display limit bounding box for the virtual display area associated with the first scale factor and offset;

identifying objects having at least a portion of their content falling within the virtual display limit bounding box; and,

for each of such objects,
retrieving content associated with that object; and
applying an appropriate scale factor and offset to the content to render the view of the Web content.

47. The mobile phone of claim 30, wherein execution of the instructions performs further operations comprising:

parsing markup language code corresponding to the received Web content to logically group selected content into objects;

generating page layout information including a bounding box for each object, the bounding box defining width and height dimensions for the object; and

storing information that links each object with its corresponding page layout information;

wherein the page layout information further includes information from which a page layout location of each of the bounding boxes can be determined.

48. The mobile phone of claim 30, wherein the scalable vector-based content includes scalable text content, and wherein execution of the instructions performs further operations comprising scaling a scalable font to render the scalable text content.

49. The mobile phone of claim 30, wherein the original format of the Web page defines a width for the Web page, and wherein execution of the instructions performs further operations comprising:

determining an applicable scale factor to fit the width of the Web page across a display area of the touch-sensitive display; and

employing the scale factor that is determined as the first scale factor.

50. The mobile phone of claim 30, wherein at least a portion of the instructions comprise Java-based instructions configured to be executed on a Java virtual machine.

51. The mobile device of claim 30, wherein the original format of the Web page comprises HTML-based Web content and the vector-based scalable content comprises scalable vector-based page layout information, and wherein execution of the instructions performs further operations comprising:

processing the HTML-based Web content with a rendering engine to generate page layout information corresponding to the original page layout as interpreted by the rendering engine;

employing the page layout information to generate the scalable vector-based page layout information.

52. A mobile device, comprising:

a processor,

wireless communications means, to facilitate wireless communication with a network via which Web content may be accessed;

a touch-sensitive display;

flash memory, operatively coupled to the processor, in which a plurality of instructions are stored that when executed by the processor enable the mobile device to perform operations including,

rendering a browser interface via which a user is enabled to request access to a Web page comprising HTML-based Web content defining an original page layout, functionality, and design of content on the Web page; retrieving and processing the HTML-based Web content to produce scalable content; and

employing the scalable content and/or data derived therefrom to,

render a view of the Web page on the touch-sensitive display; and

re-render the Web page in response to associated user inputs to enable the user to iteratively zoom in and out views of the Web page while preserving an original page layout, functionality, and design defined by the HTML-based Web content as interpreted by a rendering engine,

wherein preservation of the functionality defined by the HTML-based Web content includes preservation of hyperlink functionality.

53. The mobile device of claim **52**, wherein the device comprises a mobile phone.

54. The mobile device of claim **52**, wherein the device comprises one of a Personal Digital Assistant (PDA) or pocket PC.

55. The mobile device of claim **52**, wherein execution of the instructions performs further operations comprising enabling the user to zoom in on a user-selectable portion of a display of the Web page in response to a user interface input made via the touch-sensitive display.

56. The mobile device of claim **55**, wherein the user interface input enables the user to define a window of a current view of the Web page on which to zoom in on.

57. The mobile device of claim **52**, wherein the display of the Web page is re-rendered in real-time to effect zooming operations.

58. The mobile device of claim **52**, wherein the Web page includes at least one hyperlink, and wherein execution of the instructions performs further operations comprising:

enabling the user to select the hyperlink via the touch-sensitive display; and, in response thereto,

retrieving and processing HTML-based Web content associated with the hyperlink to produce additional scalable content; and

employing the additional scalable content and/or data derived therefrom to render the Web content associated with the hyperlink on the touch-sensitive display.

59. The mobile device of claim **52**, wherein at least a portion of the scalable content comprises scalable vector-based content.

60. The mobile device of claim **52**, wherein execution of the instructions performs further operations comprising

returning the display of the Web page to a previous view in response to a corresponding user input made via the touch-sensitive display.

61. The mobile device of claim **52**, wherein execution of the instructions performs further operations comprising enabling a user to pan a display of the Web content in response to a corresponding user input made via the touch-sensitive display.

62. The mobile device of claim **61**, wherein execution of the instructions performs further operations comprising enabling the display of the Web content to be panned in real-time.

63. The mobile device of claim **52**, wherein the page layout of the Web page is defined to have an original aspect ratio, and wherein the scalable content and/or data derived therefrom is scaled to render a display having a different aspect ratio.

64. The mobile device of claim **52**, wherein execution of the instructions performs further operations comprising enabling a user to view a column of the Web content at a higher resolution than a current resolution by tapping on the column via the touch-sensitive display, wherein in response thereto, the display is re-rendered such that content corresponding to the selected column is displayed across the touch-sensitive display.

65. The mobile device of claim **52**, wherein the Web content includes at least one image, and wherein execution of the instructions performs further operations comprising enabling a user to view an image at a higher resolution than a current resolution by tapping on the image via the touch-sensitive display, wherein in response thereto, the display is re-rendered such that the image is displayed to fit a width of a display area of the touch-sensitive display.

66. The mobile device of claim **52**, wherein execution of the instructions performs further operations comprising enabling a user to view a paragraph of the Web content at a higher resolution than a current resolution by tapping on the paragraph via the touch-sensitive display, wherein in response thereto, the display is re-rendered such that content corresponding to the selected paragraph is displayed across at least one of a width and height of a display area of the touch-sensitive display.

67. The mobile device of claim **66**, wherein the content of the paragraph is reformatted to fit characteristics of the display area when the display is re-rendered.

68. The mobile device of claim **52**, wherein the Web page includes text, layout attributes, and images, and wherein execution of the instructions performs further operations comprising:

receiving content corresponding to the text and layout attributes via a first connection; and

receiving content corresponding to at least one image via a second connection.

69. The mobile device of claim **68**, further comprising dynamic memory having at least a portion employed for rendering purposes, wherein execution of the instructions performs further operations comprising:

mapping the object vectors and associated bounding boxes to a virtual display area in the dynamic memory.

70. The mobile device of claim **69**, wherein execution of the instructions performs further operations comprising:

determining a first scale factor and offset in response to one or more corresponding user inputs defining a user-selectable zoom level and pan corresponding to a rendered display of the Web page desired by a user;

determining a virtual display limit bounding box for the virtual display associated with the first scale factor and offset;

31

identifying object bounding boxes having at least a portion falling within the virtual display limit bounding box; and,

for each of such object bounding boxes,

retrieving content associated with that object bounding box; and

applying an appropriate scale factor to the content to render the display.

71. The mobile device of claim 52, further comprising dynamic memory having at least a portion employed for rendering purposes, wherein execution of the instructions performs further operations comprising:

building a display list via use of the scalable content and rendering the display list on a virtual display area in the dynamic memory; and

scaling the display list to re-render the display of the Web page.

72. The mobile device of claim 52, wherein execution of the instructions performs further operations comprising:

parsing HTML-based code corresponding to the received Web content to identify content on the Web page;

logically grouping selected content into objects;

defining a primary datum corresponding to the original page layout; and,

for each object,

defining an object datum corresponding to a layout location datum for the object's associated display content;

generating a vector from the primary datum to the object datum for the object; and

creating a reference that links the object to its corresponding vector.

73. The mobile device of claim 72, wherein execution of the instructions performs further operations comprising:

generating a bounding box for each object, the bounding box representing a portion of a rendered display page occupied by the object's associated group of content.

74. The mobile device of claim 52, wherein the scalable content includes scalable text content, and wherein execution of the instructions performs further operations comprising scaling a scalable font to render the scalable text content.

75. The mobile device of claim 52, wherein the original format of the Web page defines a width for the Web page, and wherein execution of the instructions performs further operations comprising:

determining an applicable scale factor to fit the width of the Web page across a display area of the touch-sensitive display; and

employing the scale factor to render the display area.

76. The mobile device of claim 52, wherein at least a portion of the instructions comprise Java-based instructions configured to be executed on a Java virtual machine.

77. The mobile device of claim 52, wherein a portion of the HTML-based Web content comprises XML-based content.

78. The mobile device of claim 52, wherein a portion of the HTML-based Web content comprises cascading style sheet data.

79. A mobile device, comprising:

processing means;

wireless communications means, to facilitate wireless communication with a network via which Web content may be accessed;

touch-sensitive display means, to facilitate user input and display rendered content;

programmed circuit means; and

storage means, in which a plurality of instructions are stored,

32

wherein, upon execution of the instructions by at least one of the processing means and programmed circuit means, the mobile device is enabled to perform operations, including,

rendering a browser interface via which a user is enabled to request to access to a Web page comprising HTML-based Web content defining an original page layout, functionality, and design of content on the Web page; retrieving and processing the HTML-based Web content to produce scalable content; and

employing the scalable content and/or data derived therefrom to,

render a view of the Web page on the touch-sensitive display; and

re-render the Web page in response to associated user inputs made via the touch-sensitive display means to enable the user to iteratively zoom in and out views of the Web page while preserving an original page layout, functionality, and design defined by the HTML-based Web content as interpreted by a rendering engine,

wherein preservation of the functionality defined by the HTML-based Web content includes preservation of hyperlink functionality.

80. The mobile device of claim 79, wherein the processing means includes a general-purpose processor.

81. The mobile device of claim 79, wherein at least a portion of the programmed circuit means is embodied as a special-purpose processor.

82. The mobile device of claim 79, wherein execution of the instructions performs further operations comprising enabling the user to zoom in on a user-selectable portion of a display of the Web page in response to a user interface input made via the touch-sensitive display.

83. The mobile device of claim 82, wherein the user interface input enables the user to define a window of a current view of the Web page on which to zoom in on.

84. The mobile device of claim 79, wherein the display of the Web page is re-rendered in real-time to effect zooming operations.

85. The mobile device of claim 79, wherein execution of the instructions performs further operations comprising enabling a user to pan a display of the Web content in response to a corresponding user input made via the touch-sensitive display.

86. The mobile device of claim 85, wherein execution of the instructions performs further operations comprising enabling the display of the Web content to be panned in real-time.

87. The mobile device of claim 79, wherein the Web content includes at least one image, and wherein execution of the instructions performs further operations comprising enabling a user to view an image at a higher resolution than a current resolution by tapping on the image via the touch-sensitive display, wherein in response thereto, the display is re-rendered such that the image is displayed to fit across a width of a display area of the touch-sensitive display.

88. The mobile device of claim 79, further comprising dynamic memory having at least a portion employed for rendering purposes, wherein execution of the instructions performs further operations comprising:

building a display list of scalable content via use of the scalable content and rendering the display list on a virtual display area in the dynamic memory; and scaling the scalable content in the display list to re-render the display of the Web page.

* * * * *

Exhibit 3A

0

TITLE OF INVENTION

5 Method and system for resolution independent vector display of HTML and XML content.

10

FIELD OF INVENTION

15

The present invention relates to conversion of Internet and World Wide Web content to scaleable vector representation and more particularly to systems and methods for retrieval and conversion of HTML, XML, and other Internet content to vector representations of that content.

20

BACKGROUND OF THE INVENTION

25

Much of the Internet content has been designed for display on desktop computers with a single target resolution. Even though HTML has the ability to adapt to changes in screen resolution major Internet content providers have chosen to display their Web pages using fixed resolution structures such as tables. This gives them the ability to control the look and feel of their Web sites. This fixed resolution approach has evolved to the point that the fixed resolution layout of Web pages has become the most common method to brand or uniquely identify Web sites.

30

35

While this fixed resolution approach is good for marketing and product differentiation it does present a daunting technical problem for technologies like cell phones and hand held computers wanting to display Internet Web page content. The problem is how to display Internet content designed for desktop computers on small screen, low resolution, or different aspect ratio devices. The present invention, method and system for resolution independent vector display of HTML and XML content, addresses and solves this problem.

40

45

50

Embodied in this invention is the solution to displaying Internet content on cell phones, hand held computers, wireless Internet devices, displays with very high resolution, and very large displays. Cell phone users can see their favorite Internet Web sites in the same graphic layout they are use to from their desktop computers. Hand held computers, with different display aspect ratios, display Web pages scaled to fit their displays. Next generation high-resolution displays will also have problems with Internet content. The high resolution will cause Web pages to display very small. Vector based content will scale up display content to readable useful resolutions. Very large displays using vector scaling can display building size images, advertising, educational, and any Internet or Intranet content.

0

BRIEF SUMMARY OF INVENTION

5 This invention is a method and system for creating resolution
independent vector display of Internet content such as HTML and
XML based documents. Application of vector technology allows
scaling of Web content to fit any resolution or screen size. The
content can be scaled larger and smaller (zoomed) for better
10 viewing.

To accomplish this, HTML and XML data is rendered to a vector
format. The client requests Internet content from a proxy. The
proxy retrieves the Internet content and converts it to a vector
15 format. The vector-formatted content is then sent back to the
client for display.

BRIEF DESCRIPTION OF THE DRAWINGS

20

The process and advantages of the invention may be better
understood by referring to the accompanying drawings in which like
reference numbers represent like parts, in which:

25

FIG. 1 is a schematic representation of the process whereby a
client system requests Internet content and receives a vector
representation of that content.

30

FIG. 2 illustrates one embodiment of the invention where various
computer devices acquire vector representations of Internet
content through a proxy server.

35

40

45

50

DETAILED DESCRIPTION OF THE INVENTION

0
5 One embodiment of the process involved in the invention is illustrated in FIG. 1. The client 100 is a system such as a computer or phone. The client makes a request 101 for Internet content from a proxy. This content may be an HTML (HyperText Markup Language) or XML (eXtensible Markup Language) document.
10 The proxy may be a server system across a network connection or a software agent present on the client system. The proxy retrieves the Internet content 102, which may reside anywhere on a network or the proxy itself. The proxy converts the Internet content into
15 scaleable vector information 103. The proxy then sends this vector information back to the client, where it can be displayed 104. The client can then zoom and pan on the vector representation to more clearly see an overview or details in the document.

0
5
10
15
20
25
30
35
40
45
50
55
60
65
70
75
80
85
90
95
100
105
110
115
120
125
130
135
140
145
150
155
160
165
170
175
180
185
190
195
200
205
210
215
220
225
230
235
240
245
250
255
260
265
270
275
280
285
290
295
300
305
310
315
320
325
330
335
340
345
350
355
360
365
370
375
380
385
390
395
400
405
410
415
420
425
430
435
440
445
450
455
460
465
470
475
480
485
490
495
500
505
510
515
520
525
530
535
540
545
550
555
560
565
570
575
580
585
590
595
600
605
610
615
620
625
630
635
640
645
650
655
660
665
670
675
680
685
690
695
700
705
710
715
720
725
730
735
740
745
750
755
760
765
770
775
780
785
790
795
800
805
810
815
820
825
830
835
840
845
850
855
860
865
870
875
880
885
890
895
900
905
910
915
920
925
930
935
940
945
950
955
960
965
970
975
980
985
990
995

FIG. 2 is an illustration of the preferred embodiment of the invention. A user of a computer device, such as a personal computer 120, hand held computer 121 or phone 122 requests to view Internet content. As stated above, this content may be in the form of an HTML or XML document. The request is sent across a network, for instance the Internet, using a protocol such as HTTP (HyperText Transfer Protocol) 130, 131 or WAP (Wireless Access Protocol) 132. The proxy server 125 receives the request and fetches the actual content 135, 136, 137 from the network. The proxy then converts the content into scaleable vector information that is sent back to the client. At this point the user can take advantage of the benefits of vector information to zoom and pan in order to better see the information. The user may select a hyperlink from the content, which starts the process over again.

35 The client may be a personal computer. Many Web pages are designed for a smaller resolution (for example 640x480 pixels) than typical personal computers display (for example 1024x768 pixels). Sometimes selected text on a Web page is much smaller than other text. Allowing users to magnify the entire page allows them to overcome these design flaws. Alternately, the client may be a small device, such as a hand held computer or a cell phone, which has a smaller display area than common Web pages are designed for. A scaleable vector representation would enable these users to access the information on the billions of existing Web pages in a simple and reasonable way.

50 The client software may be a plug-in to a World Wide Web browser such as Netscape Navigator or Microsoft Internet Explorer. The browser may be configured to handle the connection to the proxy server. Alternately, the client software may be a Java applet running in a browser. Or the client software may be a stand-alone program that talks to the proxy server or proxy software directly. The client software may bypass the proxy when requesting information that won't be translated to vectors, such as bitmaps.

0

As mentioned above, the proxy may be a proxy server as understood in the Internet domain. A proxy may also be software that resides on the client system or on the same server as the Internet content.

5

10

Internet content is typically represented as HTML. It may consist of bitmaps, animations, audio files or other information referenced in the parent document. Alternately, Internet content may be stored in XML and transformed using something such as XSLT (eXtensible Style Language Transformation) to a rendering format such HTML, XHTML or another XML language, including an XML-based vector language. Internet content may be stored in many other formats, including, but not limited to, GIF, JPG, PDF, SVF, or MP3.

15

20

The vector information created by the proxy is understood to be a reasonable representation of the Internet content. It may contain text and layout information. It may contain bitmaps or references to bitmaps. Hyperlinks that point to other Internet content may be present in the vector information. The user may select a hyperlink in order to view other Internet content. Lines and other graphic information may also be present in order to represent the content reasonably.

25

30

35

40

45

50

55

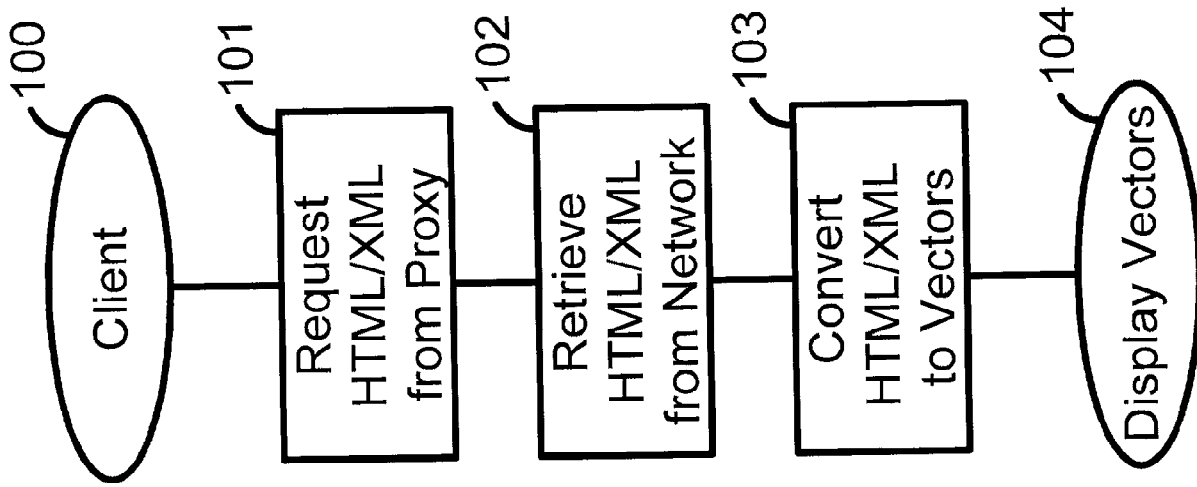


Fig. 1

Exhibit 3B

0

TITLE OF INVENTION

5

Method and system for selection, retrieval, and conversion of computer content to vector format for resolution independent display.

FIELD OF INVENTION

10

The present invention relates to selection and conversion of computer content to vector representation for resolution independent display and more particularly to systems and methods for selection, retrieval, and conversion of CAD drawings, graphic files, spreadsheets, text documents, and other computer content to vector representations for resolution independent zoom and pan enabled display.

15

COPYRIGHT NOTICE

20

A portion of the disclosure of this patent document contains material which is subject to copyright protection. The copyright owner has no objection to the facsimile reproduction by anyone of the patent document or patent disclosure as it appears in the Patent and Trademark Office patent file or records, but otherwise reserves all copyright rights whatsoever.

25

BACKGROUND OF THE INVENTION

30

Most computer content is displayed as fixed resolution text and bitmaps. Some programs like CAD systems and bitmap graphic editors provide the ability to zoom displayed data on desktop computers. Plug ins for Internet browsers like Adobe's Acrobat and SoftSource's SVF viewer allow zoom enabled display of PDF and SVF files. Plug ins have been successful in expanding the use of zoom enabled documents on the Internet.

35

The fixed resolution approach is good solution for displaying most computer content. It does present a daunting technical problem for technologies like cell phones, wireless hand held computers, very large, and high resolution display systems. Resolution independent vector display of computer content solves these problems.

40

Embodied in this invention is the solution to displaying data in a vector format that allows the user to zoom on the content. Computer content on cell phones, hand held computers, wireless Internet devices, displays with very high resolution, and very large displays. Cell phone users can view documents they are use to from their desktop computers. Hand held computers, with different display aspect ratios, display computer content scaled to fit their display. Next generation high-resolution displays

50

0 will also have problems with computer content. The high resolution
will cause computer fixed resolution content like bitmaps to
display very small. Vector based content will scale up display
content to readable useful resolutions. Very large displays using
5 vector scaling can display building size images, advertising,
educational, and any content.

10 BRIEF SUMMARY OF INVENTION

This invention is a method and system for creating resolution
independent Internet vector display of computer content such as
CAD drawings, graphic files, spreadsheets, and other computer
content. Application of vector technology allows scaling of
15 computer content to fit any resolution or screen size. The content
can be zoomed (scaled larger and smaller) for better viewing.

To accomplish this, computer content is rendered to a vector
format. The client requests computer content from a universal
vector translator. The universal vector translator retrieves the
computer content and converts it to a vector format. The vector-
20 formatted content is then sent back to the client for display.

25 BRIEF DESCRIPTION OF THE DRAWINGS

The process and advantages of the invention may be better
understood by referring to the accompanying drawings in which like
reference numbers represent like parts, in which:

30 FIG. 1 is a schematic representation of the process whereby a
client system requests computer content and receives and displays
a vector representation of that content.

35 FIG. 2 illustrates one embodiment of the invention where various
computer devices request and acquire computer content, convert
that content to vector format, and display vector representations
of computer content on client devices.

40 FIG. 3 illustrates the client to server and server to client
information flow. The client requests computer content and
transfers the computer content in vector format for display.

45 FIG. 4 illustrates the server to computer source, in this case the
Internet, and the computer content is its original format
transferred to the server. The computer content is then converted
to vector format the sent on to the requesting client for display.
50

0

DETAILED DESCRIPTION OF THE INVENTION

5

10

15

One embodiment of the process involved in the invention is illustrated in FIG. 1. The client device 100 is a system such as a computer or phone. The client makes a request 101 for computer content from a server. This content may be a CAD drawing, graphic file, spreadsheet, text based document, or other computer content. The computer content vector converter can work on a server system across a network connection or as a software agent present on the client system. The server retrieves the Internet content 102, which may reside anywhere on a network, the Internet or on the server itself. The server based universal vector translator converts the Internet content into scaleable vector information 103. The server then sends this vector information back to the client 104. The vector content is then displayed on the client device 105. The client can then zoom and pan on the vector representation to more clearly view details in the document.

20

25

30

35

40

FIG. 2 is an illustration of the preferred embodiment of the invention. A user of a computer device, such as a personal computer 120, wireless hand held computer 121 or cell phone 122 requests to view computer content. As stated above, the computer content may be in the form of a CAD drawing, graphic file, spreadsheet, text based document, or other computer content. The request is sent across a network, for instance the Internet, using a protocol such as HTTP (HyperText Transfer Protocol) 130, 131 or WAP (Wireless Access Protocol) 132. The server 150 receives the request and fetches the actual content 140 Internet ULR (Uniform Resource Locator), 141 FTP (File Transfer Protocol) using ULI (Uniform Resource Identifier), 142 from the network, local machine, or other source. The server based universal vector translator then converts the content into scaleable vector information that is sent back to the client 160, 161, 162. The vector representation of the computer content is displayed on the client device. At this point the user can take advantage of the benefits of vector information to zoom and pan in order to better view the information. The user may select a hyperlink from the content, which starts the process over again.

45

FIG. 3 is an illustration of the preferred embodiment of the client side of the invention. A user of a computer device, such as a personal computer 120, requests to view computer content. The request is sent across a network, for instance the Internet, using a protocol such as HTTP (HyperText Transfer Protocol) 130. The converted the content is sent back to the client 160 for display.

50

FIG. 4 is an illustration of the preferred embodiment of the server side of the invention. The request is sent across a network, for instance the Internet, using a protocol such as HTTP (HyperText Transfer Protocol) 170. The computer content is then transferred to the server in its original format 171. The computer content is converted into vector format. The converted computer content is then transferred to the requesting client for display.

0

The client may be a personal computer or other high resolution display device. Much of the computer content is designed for lower resolution displays (for example 640x480 pixels). Modern computer systems, projector, and wall size displays can have resolutions measured in the millions of pixels. Allowing users to magnify the computer content allows them to overcome the original contents design limitations. Alternately, the client may be a small device, such as a hand held computer or a cell phone, which have a smaller display area. A scaleable vector representation enables access any computer content on any resolution display device.

5

10

As mentioned above, the server based universal vector translator may be a server as understood in the network and Internet domain. The server based universal vector translator may also be software that resides on the client system or on the same server as the computer content.

15

20

The vector information created by the server based universal vector translator is understood to be a reasonable representation of the computer content. It may contain vector and text and layout information. It may contain bitmaps or references to bitmaps. Hyperlinks that point to other computer content may be present in the vector representation. The user may select a hyperlink in order to view other content. Lines and other graphic information may also be present in order to represent the content reasonably.

25

30

35

40

45

50

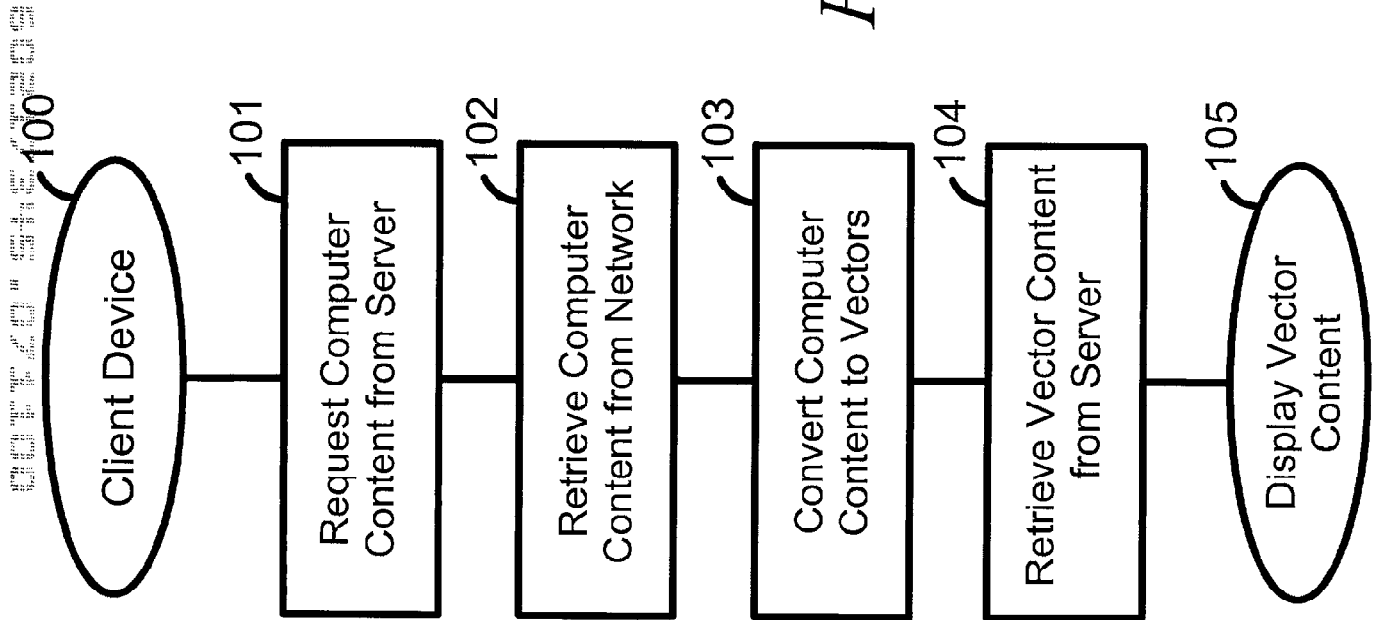


Fig. 1

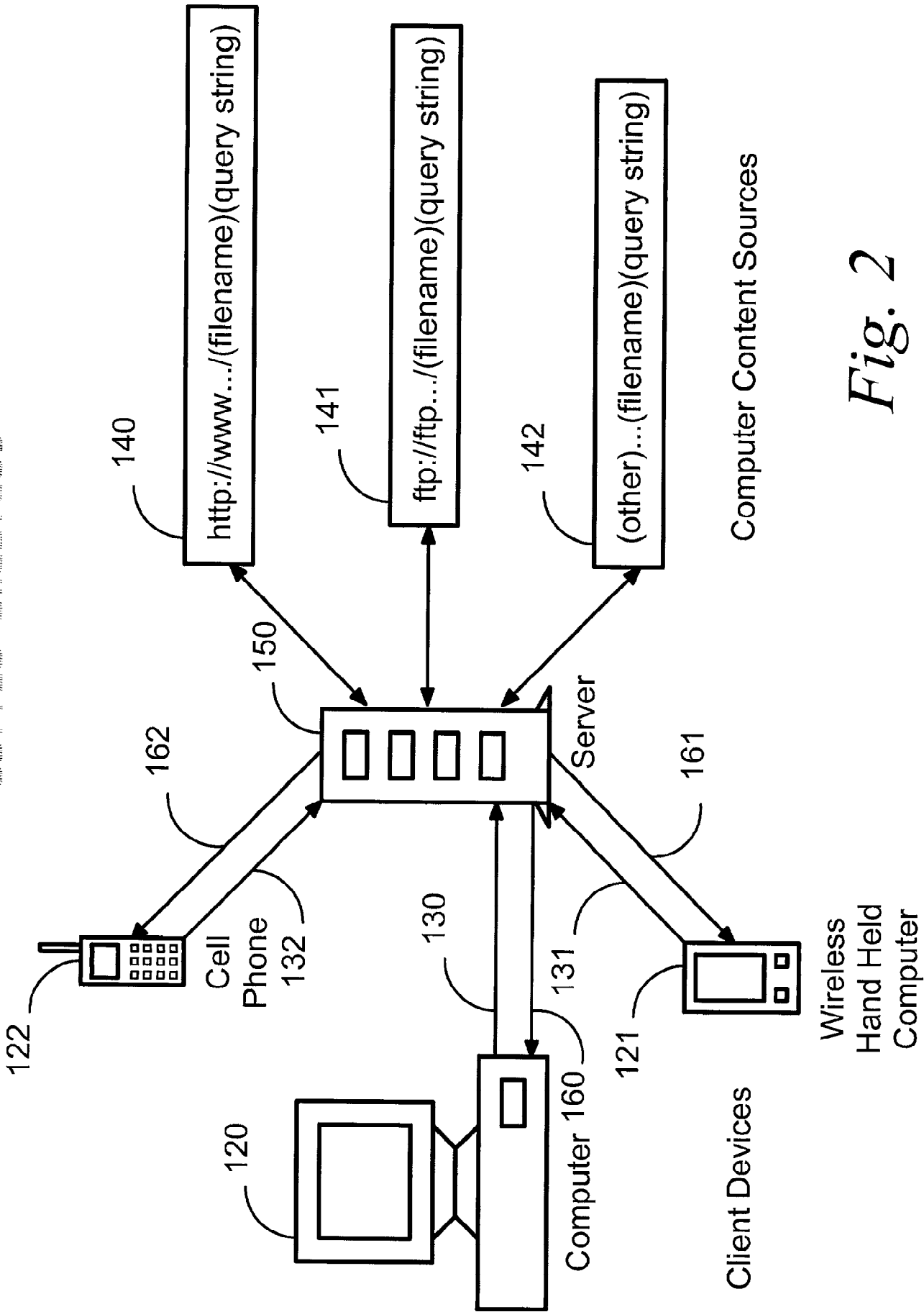


Fig. 2

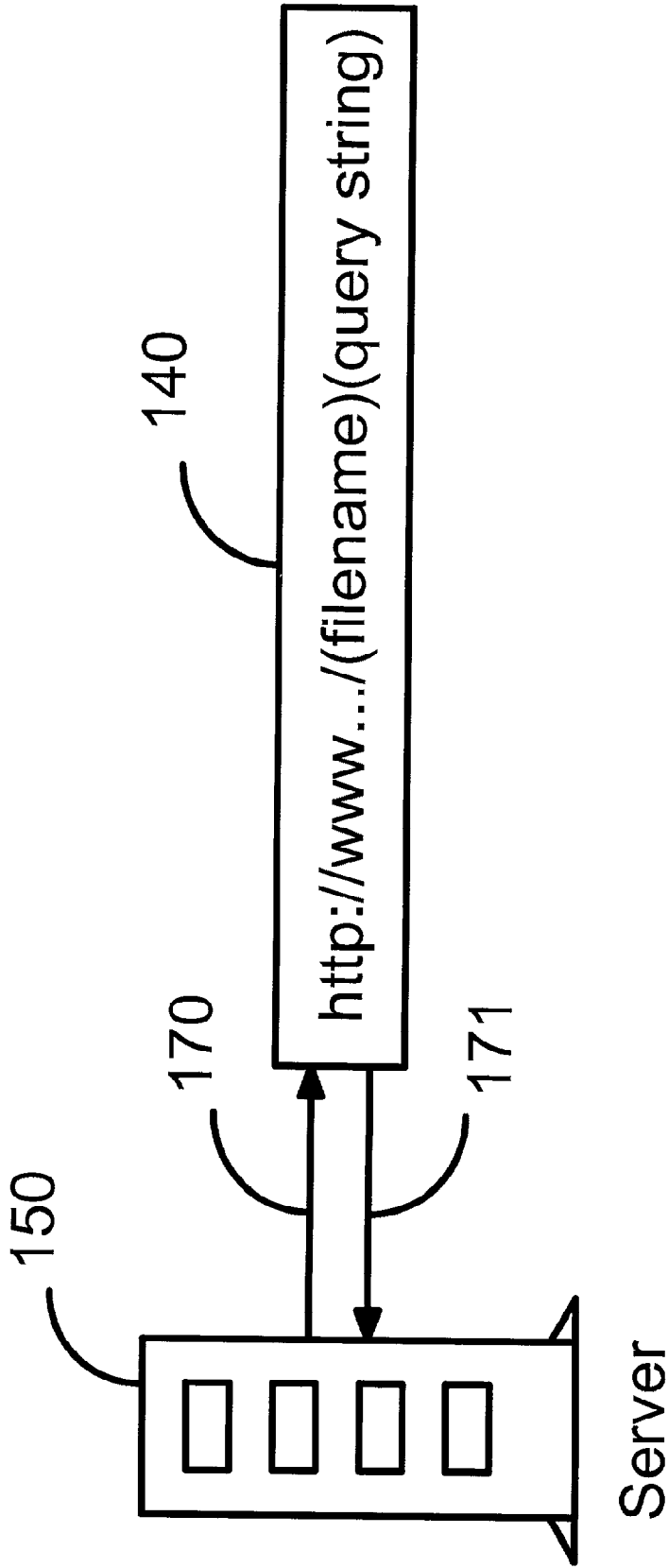


Fig. 4

Exhibit 3C

ABSTRACT OF THE DISCLOSURE

Apparatus and methods are provided for creating resolution independent vector display of Internet content to allow it to be scaled (zoomed) larger and smaller for better viewing or to fit any resolution or screen size. According to one embodiment, novel server processing of Web content is provided that converts Web content requested by a client to a scaleable vector format, such as SVF. The vector format enables the client to substantially retain an original page layout within a set of layouts originally intended to be associated with the requested Web content by including page layout information. Finally, the vector-formatted Web content is provided to the client. A client-side vector viewer receives requested Web content and displays a vector representation of the requested Web content that substantially retains page layout and/or graphics associated with the requested Web content.

0982851-040701

RESOLUTION INDEPENDENT VECTOR DISPLAY OF INTERNET CONTENT

[0001] This application claims the benefit of U.S. Provisional Application No. 60/211,019, filed June 12, 2000, which is incorporated herein by reference.

COPYRIGHT NOTICE

[0002] Contained herein is material that is subject to copyright protection. The copyright owner has no objection to the facsimile reproduction of the patent disclosure by any person as it appears in the Patent and Trademark Office patent files or records, but otherwise reserves all rights to the copyright whatsoever.

BACKGROUND OF THE INVENTION

Field of the Invention

[0003] The invention relates generally to conversion of Internet and World Wide Web content to scaleable vector representation. More particularly, the invention relates to apparatus and methods for zoom enabling the display of content in an Internet information browser by retrieving and converting HyperText Markup Language (HTML), eXtensible Markup Language (XML), and other Internet content to vector representations of that content.

Description of the Related Art

[0004] Text only Internet information browsers began as a project at the CERN, European Organization for Nuclear Research, facility in Geneva Switzerland. From its inception the intent was to provide a mesh or web of access to data with a common user

Docket No. 005207.P001
Express Mail No. EL845313650US 1

0992511 010704
102040 1158260

BRIEF SUMMARY OF THE INVENTION

[0007] A method and apparatus for supporting resolution independent vector display of Internet content is disclosed. According to one embodiment, novel client processing of Web content is provided. The client receives requested Web content and displays a vector representation of the requested Web content that substantially retains page layout and/or graphics associated with the requested Web content.

[0008] According to another embodiment, novel server processing of Web content is provided. First, the server receives a request for Web content from a client. The requested Web content is then converted to a vector format to produce vector-formatted Web content corresponding to the requested Web content. The vector format enables the client to substantially retain an original page layout within a set of layouts originally intended to be associated with the requested Web content by including page layout information. Finally, the vector-formatted Web content is provided to the client.

[0009] Other features of the present invention will be apparent from the accompanying drawings and from the detailed description which follows.

000001 040701

BRIEF DESCRIPTION OF THE DRAWINGS

[0010] The appended claims set forth the features of the invention with particularity. The invention, together with its advantages, may be best understood from the following detailed description taken in conjunction with the accompanying drawings of which:

[0011] **Figure 1** is a flow diagram illustrating a process whereby a client system requests Internet content and receives a vector representation of that content according to one embodiment of the present invention.

[0012] **Figure 2** illustrates various computer devices acquiring vector representations of Internet content through a proxy server according to one embodiment of the present invention.

[0013] **Figure 3** is a flow diagram illustrating a process whereby layout information is defined and used in combination with a scale factor and offset to zoom enable Internet content according to one embodiment of the present invention.

[0014] **Figure 4** conceptually illustrates the use of layout information in combination with a scale factor and offset to zoom enable Internet content according to one embodiment of the present invention.

[0015] **Figure 5** is an example of a computer system upon which one embodiment of the present invention may be implemented.

[0016] **Figure 6** conceptually illustrates a first architectural model in which content is converted to vector format as a service provided by a web site according to one embodiment of the present invention.

[0017] **Figure 7** conceptually illustrates a second architectural model in which content is converted to vector format as a service provided by a wireless service provider according to one embodiment of the present invention.

Docket No. 005207.P001
Express Mail No. EL845313650US

[0018] **Figure 8** conceptually illustrates a third architectural model in which content is converted to vector format as a service provided by an Internet Service Provider (ISP) according to one embodiment of the present invention.

000001 040704

Docket No. 005207.P001
Express Mail No. EL845313650US 5

005207.P001

application, the map tiles surrounding the viewed map could be downloaded and stored while the user was working with the initial tile, enabling an experience remarkably free from the current frustrations of waiting for a new map to be transferred for even the smallest change in magnification or coverage. If the user knows ahead of time what city they will visit on a business trip, maps and additional travel information in great detail could also be pre-loaded using a high bandwidth connection at home or in the office before heading out to shop or conduct business in the city. Additionally, SVF is a more efficient way to store web content. Resulting content files are reduced in size by anywhere from 20 to 80 percent over their source. SVF is also very compressible. With target file size reduction in the range of 90%, SVF files can take up as little as 1/10th the space of the web files in current use. This means that pre-converted content can be moved up to 10 times the rate of current web pages, and as much as 10 times as many pages, maps, stock charts, etc. can be stored for instant retrieval on the hand held platform as can be handled with current web technology.

[0028] When used on content created natively in SVF, additional capability can be extended to the client-side viewer. Because SVF is, at its heart, a database, information in database form can be included, linked to or manipulated through the viewer environment. That means, for example, that data streams for stocks can be transferred to the client and then displayed and manipulated real time on the end device. Imagine checking a quote on a security and instantaneously being able to look at the performance of the stock over any period from minutes to ten years in duration, graph it, and compare the graphs with indexes or stocks within a pre-identified group. Because the data can move as a native database, and not as a set of static graphical images, setting up this impressively powerful world on a device as small as a wireless PDA is easy.

Docket No. 005207.P001
Express Mail No. EI.845313650US 9

005207.P001

started immediately. In addition to streaming, the server-side content converter may also layer the content by type. This means that text can be put in one layer, links in another, GIF images in another, Javascript in another and so on. Layers can be turned on or off depending upon client capabilities, making files for less capable clients, or for users interested in a reduced functionality, higher transfer performance mode to be handled automatically.

[0033] All operational modes may be controlled through an administrative interface or accessible through a straightforward API.

[0034] The system works with existing firewalls and within standard security protocols. In more secure modes, the server-side content converter and the client-side viewer may operate using Public/Private key authentication and encryption.

[0035] **Figure 1** is a flow diagram illustrating a process whereby a client system requests Internet content and receives a vector representation of that content according to one embodiment of the present invention. In this example, the client 100 is a system such as a computer or phone. The client makes a request at processing block 101 for Internet content from a proxy. This content may be an HTML or XML document. The proxy may be a server system across a network connection or a software agent present on the client system. At processing block 102, the proxy retrieves the Internet content, which may reside anywhere on a network or the proxy itself. At processing block 103, the proxy converts the Internet content into scaleable vector information, such as SVF. Then, at processing block 104, the proxy sends this vector information back to the client, where it can be displayed. The client can then zoom and pan on the vector representation to more clearly see an overview or details in the document.

Docket No. 005207.P001
Express Mail No. EL845313650US

11

CLAIMS

What is claimed is:

005207.P001

- 1 1. A method of presenting Web content on a client, the method comprising
2 displaying a vector representation of requested Web content that substantially
3 retains page layout and/or graphics associated with the requested Web content.
- 1 2. The method of claim 1, wherein prior to said displaying the method further
2 comprising converting the requested Web content to the vector representation, the
3 vector representation including information regarding an intended original page
4 layout for the Web content.
- 1 3. The method of claim 2, wherein the converting is performed prior to receipt of the
2 Web content by the client.
- 1 4. The method of claim 2, wherein the converting is performed local to the client.
- 1 5. The method of claim 1, wherein the client comprises a wireless Internet device, a
2 cellular phone, a handheld computer, a computer with a high resolution display, a
3 computer with a very large display, an electronic billboard, or a device having a
4 display with an unconventional aspect ratio.
- 1 6. A method comprising:
2 receiving a request for Web content from a client; and
3 converting the requested Web content to a vector format and producing vector-
4 formatted Web content corresponding to the requested Web content, the
5 vector format enabling the client to substantially retain an original page

Docket No. 005207.P001

Express Mail No. EL845313650US

19

0928511-040701

3 responsive to the client request, the server converting the requested Web content
4 to a vector format and producing vector-formatted Web content
5 corresponding to the requested Web content, the vector format enabling
6 the client to substantially retain an original page layout within a set of
7 layouts originally intended to be associated with the requested Web
8 content by including page layout information;
9 the server transferring the vector-formatted Web content to the client;
10 the client receiving the vector-formatted Web content from the server; and
11 the client displaying a vector representation of the requested Web content on the
12 client while substantially retaining the original page layout associated with
13 the requested Web content by rendering the vector-formatted Web content
14 in accordance with the page layout information.

1 13. The method of claim 12, further comprising manipulating the vector
2 representation of the requested Web content and redisplaying the resulting vector
3 representation without the need for further requests to the server.

1 14. The method of claim 13, wherein the manipulating the vector representation
2 comprises panning on the vector representation.

1 15. The method of claim 13, wherein the manipulating the vector representation
2 comprises zooming on the vector representation.

1 16. A system for transferring and displaying Web content comprising:
2 a server to deliver Web content in response to content requests and to convert the
3 Web content to a vector format that retains an original page layout within a

005207.P001

4 set of layouts originally intended to be associated with the Web content by
5 including page layout information; and
6 a client communicatively coupled to the server to receive requested Web content
7 and to display a vector representation of the requested Web content on the
8 client while substantially retaining the original page layout associated with
9 the requested Web content by rendering the vector-formatted Web content
10 in accordance with the page layout information.

- 1 17. The system of claim 16, wherein the server is associated with a web site and
2 conversion of Web content to the vector format is a service provided by the web
3 site.
- 1 18. The system of claim 16, wherein the server is associated with a wireless service
2 provider and conversion of Web content to the vector format is a service provided
3 by the wireless service provider.
- 1 19. The system of claim 16, wherein the server is associated with an Internet Service
2 Provider (ISP) and conversion of Web content to the vector format is a service
3 provided by the ISP.
- 1 20. The system of claim 16, wherein the client comprises a wireless Internet device.
- 1 21. The system of claim 16, wherein the client comprises a cellular phone.
- 1 22. The system of claim 16, wherein the client comprises a handheld computer.
- 1 23. The system of claim 16, wherein the client comprises a computer with a high
2 resolution display.

005207.P001
EL845313650US

1 24. The system of claim 16, wherein the client comprises a computer with a very large
2 display.

1 25. The system of claim 16, wherein the client comprises an electronic billboard.

1 26. The system of claim 16, wherein the client comprises a device having a display
2 with an unconventional aspect ratio.

1 27. A system for transferring and displaying Web content comprising:
2 a content delivery means for delivering Web content in response to content
3 requests and for converting the Web content to a vector format that retains
4 an original page layout within a set of layouts originally intended to be
5 associated with the Web content by including page layout information; and
6 a content display means, communicatively coupled to the content delivery means,
7 for receiving requested Web content and for displaying a vector
8 representation of the requested Web content on the client while
9 substantially retaining the original page layout associated with the
10 requested Web content by rendering the vector-formatted Web content in
11 accordance with the page layout information.

1 28. A machine-readable medium having stored thereon data representing sequences of
2 instructions, the sequences of instructions which, when executed by a processor,
3 cause the processor to:
4 receive a request for Web content from a client; and
5 produce vector-formatted Web content corresponding to the requested Web
6 content by converting the requested Web content to a vector format, the

Docket No. 005207.P001
Express Mail No. EL845313650US 23

09823311 040704

7 vector format enabling the client to substantially retain an original page
8 layout within a set of layouts originally intended to be associated with the
9 requested Web content by including page layout information; and
10 provide the vector-formatted Web content to the client.

1 29. A machine-readable medium having stored thereon data representing sequences of
2 instructions, the sequences of instructions which, when executed by a processor,
3 of a client cause the processor to:
4 receive, at the client, vector-formatted Web content representing requested Web
5 content converted to a vector format; and
6 display a vector representation of the requested Web content on the client by
7 rendering the vector-formatted Web content.

INTERNATIONAL PRELIMINARY EXAMINATION REPORT

International application No.

PCT/US01/40920

V. Reasoned statement under Article 35(2) with regard to novelty, inventive step or industrial applicability; citations and explanations supporting such statement

1. statement

Novelty (N)	Claims <u>1-38</u>	YES
	Claims <u>NONE</u>	NO
Inventive Step (IS)	Claims <u>1-38</u>	YES
	Claims <u>NONE</u>	NO
Industrial Applicability (IA)	Claims <u>1-38</u>	YES
	Claims <u>NONE</u>	NO

2. citations and explanations (Rule 70.7)

Claims 1-38 meet the criteria set out in PCT Article 35(2)-(4), because the prior art does not teach or fairly suggest a method comprising step of *translating Web content from its original format into a scalable vector representation of the Web content, wherein the scalable vector representation of the Web content provides a scalable resolution-independent display of the content that substantially retains the original page layout and attributes of the content defined by its original format when rendered.*

NEW CITATIONS

US 6,057,854 A (DAVIS, JR. et al.) 02 MAY 2000, see col. 2, lines 5-35.
 US 6,011,905 A (HUTTENLOCHER et al.) 04 JANUARY 2000, see abstract and fig. 10.

878097

828511

Exhibit 3D

BRIEF SUMMARY OF THE INVENTION

[0007] A method and apparatus for supporting resolution independent vector display of Internet content is disclosed. According to one embodiment, novel client processing of Web content is provided. The client receives requested Web content and displays a vector representation of the requested Web content that substantially retains page layout and/or graphics associated with the requested Web content.

[0008] According to another embodiment, novel server processing of Web content is provided. First, the server receives a request for Web content from a client. The requested Web content is then translated into a scalable vector format to produce vector-formatted Web content corresponding to the requested Web content. The vector format enables the client to substantially retain an original page layout within a set of layouts originally intended to be associated with the requested Web content by including page layout information in a vector database. Finally, the vector-formatted Web content is provided to the client.

[0009] Other features of the present invention will be apparent from the accompanying drawings and from the detailed description that follows.

CLAIMS

What is claimed is:

- 1 1. A method comprising:
2 retrieving Web content having an original format defining an original page layout
3 and attributes of the Web content from an Internet site in response to a request of the Web
4 content from a client; and
5 translating the Web content from its original format into a scalable vector
6 representation of the Web content, wherein the scalable vector representation of the Web
7 content provides a scalable resolution-independent display of the content that
8 substantially retains the original page layout and attributes of the content defined by its
9 original format when rendered.
- 1 2. The method of claim 1, wherein the Web content comprises a Web page that is
2 stored on the Internet site in one or more markup language documents that include
3 markup language code defining the original page layout and attributes of the Web page.
- 1 3. The method of claim 1, wherein the Web content is translated from its original
2 format into the scalable vector representation on the client.
- 1 4. The method of claim 1, wherein the content is translated from its original format
2 into the scalable vector representation on a computing device that is remote from the
3 client, further comprising sending the scalable vector representation of the Web content to
4 the client.

1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
101
102
103
104
105
106
107
108
109
110
111
112
113
114
115
116
117
118
119
120
121
122
123
124
125
126
127
128
129
130
131
132
133
134
135
136
137
138
139
140
141
142
143
144
145
146
147
148
149
150
151
152
153
154
155
156
157
158
159
160
161
162
163
164
165
166
167
168
169
170
171
172
173
174
175
176
177
178
179
180
181
182
183
184
185
186
187
188
189
190
191
192
193
194
195
196
197
198
199
200
201
202
203
204
205
206
207
208
209
210
211
212
213
214
215
216
217
218
219
220
221
222
223
224
225
226
227
228
229
230
231
232
233
234
235
236
237
238
239
240
241
242
243
244
245
246
247
248
249
250
251
252
253
254
255
256
257
258
259
260
261
262
263
264
265
266
267
268
269
270
271
272
273
274
275
276
277
278
279
280
281
282
283
284
285
286
287
288
289
290
291
292
293
294
295
296
297
298
299
300
301
302
303
304
305
306
307
308
309
310
311
312
313
314
315
316
317
318
319
320
321
322
323
324
325
326
327
328
329
330
331
332
333
334
335
336
337
338
339
340
341
342
343
344
345
346
347
348
349
350
351
352
353
354
355
356
357
358
359
360
361
362
363
364
365
366
367
368
369
370
371
372
373
374
375
376
377
378
379
380
381
382
383
384
385
386
387
388
389
390
391
392
393
394
395
396
397
398
399
400
401
402
403
404
405
406
407
408
409
410
411
412
413
414
415
416
417
418
419
420
421
422
423
424
425
426
427
428
429
430
431
432
433
434
435
436
437
438
439
440
441
442
443
444
445
446
447
448
449
450
451
452
453
454
455
456
457
458
459
460
461
462
463
464
465
466
467
468
469
470
471
472
473
474
475
476
477
478
479
480
481
482
483
484
485
486
487
488
489
490
491
492
493
494
495
496
497
498
499
500
501
502
503
504
505
506
507
508
509
510
511
512
513
514
515
516
517
518
519
520
521
522
523
524
525
526
527
528
529
530
531
532
533
534
535
536
537
538
539
540
541
542
543
544
545
546
547
548
549
550
551
552
553
554
555
556
557
558
559
560
561
562
563
564
565
566
567
568
569
570
571
572
573
574
575
576
577
578
579
580
581
582
583
584
585
586
587
588
589
590
591
592
593
594
595
596
597
598
599
600
601
602
603
604
605
606
607
608
609
610
611
612
613
614
615
616
617
618
619
620
621
622
623
624
625
626
627
628
629
630
631
632
633
634
635
636
637
638
639
640
641
642
643
644
645
646
647
648
649
650
651
652
653
654
655
656
657
658
659
660
661
662
663
664
665
666
667
668
669
670
671
672
673
674
675
676
677
678
679
680
681
682
683
684
685
686
687
688
689
690
691
692
693
694
695
696
697
698
699
700
701
702
703
704
705
706
707
708
709
710
711
712
713
714
715
716
717
718
719
720
721
722
723
724
725
726
727
728
729
730
731
732
733
734
735
736
737
738
739
740
741
742
743
744
745
746
747
748
749
750
751
752
753
754
755
756
757
758
759
760
761
762
763
764
765
766
767
768
769
770
771
772
773
774
775
776
777
778
779
780
781
782
783
784
785
786
787
788
789
790
791
792
793
794
795
796
797
798
799
800
801
802
803
804
805
806
807
808
809
810
811
812
813
814
815
816
817
818
819
820
821
822
823
824
825
826
827
828
829
830
831
832
833
834
835
836
837
838
839
840
841
842
843
844
845
846
847
848
849
850
851
852
853
854
855
856
857
858
859
860
861
862
863
864
865
866
867
868
869
870
871
872
873
874
875
876
877
878
879
880
881
882
883
884
885
886
887
888
889
890
891
892
893
894
895
896
897
898
899
900
901
902
903
904
905
906
907
908
909
910
911
912
913
914
915
916
917
918
919
920
921
922
923
924
925
926
927
928
929
930
931
932
933
934
935
936
937
938
939
940
941
942
943
944
945
946
947
948
949
950
951
952
953
954
955
956
957
958
959
960
961
962
963
964
965
966
967
968
969
970
971
972
973
974
975
976
977
978
979
980
981
982
983
984
985
986
987
988
989
990
991
992
993
994
995
996
997
998
999
1000

1 5. The method of claim 4, wherein the computing device that is remote from the
2 client is operated by the Internet site that receives a request for the Web content from the
3 client via the Internet, retrieves the Web content in its original format and translates it
4 into the scalable vector representation of the Web content, and returns the scalable vector
5 representation of the Web content via the Internet to the client.

1 6. The method of claim 4, wherein the computing device that is remote from the
2 client comprises a proxy server that receives the request of the Web content from the
3 client via the Internet, retrieves the Web content in its original format from the Internet
4 site, translates the content from its original format into the scalable vector representation
5 of the Web content, and sends the scalable vector representation of the Web content back
6 to the client via the Internet.

1 7. The method of claim 6, wherein the proxy server appears transparent to the client.

1 8. The method of claim 1, wherein the client comprises one of a wireless Internet
2 device, a cellular phone, a handheld computer, a desktop computer or workstation, a
3 laptop computer, an electronic billboard, or a device having a display with an
4 unconventional aspect ratio.

1 9. A method comprising:
2 enabling a client to request Web content comprising a Web page that is stored on
3 an Internet site in an original format that include markup language code defining an
4 original page layout and attributes of objects included in the Web page,

5 routing the request through a proxy server that retrieves the Web content in its
6 original format from the Internet site and translates it into a scalable vector representation
7 of the Web content, wherein the scalable vector representation of the Web content
8 provides a scalable resolution-independent display of the content that substantially retains
9 the original page layout and attributes of the Web page's objects defined by the original
10 format of the Web page when rendered; and

11 sending the scalable vector representation of the Web content from the proxy
12 server back to the client.

1 10. The method of claim 9, wherein the client enables a user to request the Web page
2 in a conventional manner by entering a URL (uniform resource locator) for the Web page
3 via a browser hosted by the client, and a software component running on the client
4 intercepts the request for the Web page and routes the request to the proxy server rather
5 than an Internet site corresponding to the URL such that the proxy server appears
6 transparent to the client.

1 11. The method of claim 9, wherein translating the Web content from its original
2 format into the scalable vector representation of the Web content comprises:

3 parsing the markup language code to determine the original page layout of the
4 Web page, wherein the original page layout defines a layout location for a plurality of text
5 objects and/or graphic image objects included in the Web page;

6 defining a primary datum corresponding to the original page layout;

7 defining an object datum corresponding to the layout location for each of said
8 plurality of text objects, graphic layout objects, and/or graphic image objects;

THE COURT REPORTER'S OFFICE

9 generating a vector from the primary datum to the object datum for each of said
10 plurality of text objects, graphic layout objects, and/or graphic image objects; and

11 creating a reference that links each of said plurality of text objects, graphic layout
12 objects, and/or graphic image objects to its corresponding vector.

1 12. The method of claim 11, further comprising translating graphic image objects
2 from an original format into a scalable bitmap format.

1 13. The method of claim 11, further comprising storing attribute information
2 pertaining to each text object, said attribute information including a color and typefont for
3 each text object.

1 14. The method of claim 11, further comprising generating a bounding box for each
2 object that substantially circumscribes a page layout area occupied by that object in
3 correspondence with the original page layout.

1 15. A method of displaying Web content on a client, comprising:
2 responsive to a request for Web content by the client, said Web content
3 comprising a Web page that is defined to be rendered at a predetermined resolution,
4 receiving, at the client, vector-formatted Web content comprising a scalable
5 vector representation of the Web content that provides a scalable resolution-independent
6 display of the Web content that substantially retains an original page layout and attributes
7 of the Web content corresponding to an appearance of the Web page when it is rendered
8 at its predetermined resolution; and

9 rendering the vector-formatted Web content on the client such that it is displayed
10 to have a different resolution than the predetermined resolution.

1 16. The method of claim 15, wherein the Web content is displaced at a different
2 resolution by scaling the vector-formatted Web content.

1 17. The method of claim 16, further comprising enabling a user of the client to select
2 a zoom factor by which the vector-formatted Web content is scaled.

1 18. The method of claim 16, further comprising enabling a user to cause a display of
2 the Web content to be both scaled and offset.

1 19. The method of claim 15, wherein the page layout of the Web page is defined to
2 have an original aspect ratio, and wherein the vector-formatted Web content is scaled so
3 as to produce a display having a different aspect ratio.

1 20. The method of claim 15, wherein the scalable vector representation of the Web
2 content comprises Simple Vector Format (SVF).

1 21. A computer system comprising:
2 a processor,
3 a communications device coupled to the processor, to enable the computer system
4 to be linked via the Internet to a client and an Internet site; and
5 a memory, coupled to the processor, in which a plurality of machine-executable
6 instructions are stored that when executed by the processor enable the computer system to
7 perform the operations of:

8 retrieving Web content having an original format defining an original page
9 layout and attributes of the Web content from the Internet site in response to a
10 request of the Web content from the client; and

11 translating the Web content from its original format into a scalable vector
12 representation of the Web content, wherein the scalable vector representation of
13 the Web content provides a scalable resolution-independent display of the content
14 that substantially retains the original page layout and attributes of the content
15 defined by its original format when rendered.

1 22. The computer system of claim 21, wherein execution of the plurality of the
2 machine instructions translates the Web content from its original format into the scalable
3 vector representation of the Web content by performing the operation of:

4 parsing the markup language code to determine the original page layout of the
5 Web page, wherein the original page layout defines a layout location for a plurality of text
6 objects and/or graphic image objects included in the Web page;

7 defining a primary datum corresponding to the original page layout;

8 defining an object datum corresponding to the layout location for each of said
9 plurality of text objects, graphic layout objects, and/or graphic image objects;

10 generating a vector from the primary datum to the object datum for each of said
11 plurality of text objects, graphic layout objects, and/or graphic image objects; and

12 creating a reference that links each of said plurality of text objects, graphic layout
13 objects, and/or graphic image objects to its corresponding vector.

1 23. The computer system of claim 22, wherein execution of the plurality of the
2 machine instructions further performs the operation of translating graphic image objects
3 from an original format into a scalable bitmap format.

1 24. The computer system of claim 22, wherein execution of the plurality of the
2 machine instructions further performs the operation storing attribute information
3 pertaining to each text object, said attribute information including a color and typefont for
4 each text object.

1 25. A computer system comprising:
2 a processor,
3 a communications device coupled to the processor, to enable the computer system
4 to be linked via the Internet an Internet site; and

5 a memory, coupled to the processor, in which a plurality of machine-executable
6 instructions are stored that when executed by the processor enable the computer system to
7 perform the operations of:

8 receiving vector-formatted Web content comprising a scalable vector
9 representation of a Web page that is originally defined to be rendered at a
10 predetermined resolution, said vector-formatted Web content providing a scalable
11 resolution-independent display of the Web page that substantially retains an
12 original page layout and attributes of the Web page corresponding to an
13 appearance of the Web page when it is rendered at its predetermined resolution;

14 and

15 rendering the vector-formatted Web content on the client such that it is
16 displayed to have a different resolution than the predetermined resolution.

1 26. The computer system of claim 25, wherein the Web content is displayed at a
2 different resolution by scaling the vector-formatted Web content.

1 27. The computer system of claim 25, further comprising enabling a user of the client
2 to select a zoom factor by which the vector-formatted Web content is scaled.

1 28. The computer system of claim 25, wherein execution of the plurality of machine
2 instructions by the processor further performs the operation of enabling a user to cause a
3 display of the Web content to be both scaled and offset.

1 29. The computer system of claim 25, wherein the page layout of the Web page is
2 defined to have an original aspect ratio, and wherein the vector-formatted Web content is
3 scaled so as to produce a display having a different aspect ratio.

1 30. A machine-readable medium having stored thereon a plurality of machine-
2 executable instructions that when executed by a machine performs the operations of:

3 retrieving Web content having an original format defining an original page layout
4 and attributes of the Web content from an Internet site in response to a request of the Web
5 content from the client;

6 translating the Web content from its original format into a scalable vector
7 representation of the Web content, wherein the scalable vector representation of the Web
8 content provides a scalable resolution-independent display of the content that
9 substantially retains the original page layout and attributes of the content defined by its
10 original format when rendered.

1 31. The machine-readable medium of claim 30, wherein execution of the plurality of
2 the machine instructions translates the Web content from its original format into the
3 scalable vector representation of the Web content by performing the operation of:

4
5
6
7
8
9
10
11
12
13
1
2
3
1
2
3
4
1
2
3
4

parsing the markup language code to determine the original page layout of the Web page, wherein the original page layout defines a layout location for a plurality of text objects and/or graphic image objects included in the Web page;

defining a primary datum corresponding to the original page layout;

defining an object datum corresponding to the layout location for each of said plurality of text objects, graphic layout objects, and/or graphic image objects;

generating a vector from the primary datum to the object datum for each of said plurality of text objects, graphic layout objects, and/or graphic image objects; and

creating a reference that links each of said plurality of text objects, graphic layout objects, and/or graphic image objects to its corresponding vector.

32. The machine-readable medium of claim 31, wherein execution of the plurality of the machine instructions further performs the operation of translating graphic image objects from an original format into a scalable bitmap format.

33. The machine-readable medium of claim 31, wherein execution of the plurality of the machine instructions further performs the operation storing attribute information pertaining to each text object, said attribute information including a color and typefont for each text object.

34. A machine-readable medium having stored thereon a plurality of machine-executable instructions that when executed by a machine performs the operations of:

receiving vector-formatted Web content comprising a scalable vector representation of a Web page that is originally defined to be rendered at a predetermined

5 resolution, said vector-formatted Web content providing a scalable resolution-
6 independent display of the Web page that substantially retains an original page layout and
7 attributes of the Web page corresponding to an appearance of the Web page when it is
8 rendered at its predetermined resolution; and
9 rendering the vector-formatted Web content on the client such that it is displayed
10 to have a different resolution than the predetermined resolution.

1 35. The machine-readable medium of claim 34, wherein the Web content is displayed
2 at a different resolution by scaling the vector-formatted Web content.

1 36. The machine-readable medium of claim 34, further comprising enabling a user of
2 the client to select a zoom factor by which the vector-formatted Web content is scaled.

1 37. The machine-readable medium of claim 34, wherein execution of the plurality of
2 machine instructions further performs the operation of enabling a user to cause a display
3 of the Web content to be both scaled and offset.

1 38. The machine-readable medium 34, wherein the original page layout of the Web
2 page is defined to have an original aspect ratio, and wherein the vector-formatted Web
3 content is scaled so as to produce a display having a different aspect ratio.

ABSTRACT OF THE DISCLOSURE

Apparatus and methods are provided for creating resolution-independent vector display of Internet content to allow it to be scaled (zoomed) larger and smaller for better viewing or to fit any resolution or screen size. According to one embodiment, novel server processing of Web content is provided that converts Web content requested by a client to a scalable vector format, such as Simple Vector Format. The vector format enables the Web content to be rendered by the client such that the rendered display substantially retains an original page layout defined in mark-up language document(s) in which the Web content is stored. In one embodiment, a proxy server receives Web content requests from a client and translates the content from an original format into a scalable vector representation. The scalable vector representation is then sent to client, where it is scaled and/or offset, enabling users to zoom and/or pan the Web content.

AMENDMENT

In the Specification

Please replace paragraph [0007] with the following amended paragraph:

[0007] In accordance with aspects of the invention, A method and apparatus methods, apparatus, and software for supporting resolution independent vector display of Internet (i.e., Web) content [[is]] are disclosed. According to one embodiment aspect, novel client processing of vector-formatted Web content is provided to enable Web content to be scaled and panned on a client while supporting traditional Web browser functionality. The client receives requested Web content and displays a vector representation of the requested Web content that substantially retains page layout and/or graphics associated with the requested Web content. The client processing enables the vector-formatted web content to be rendered at various scales and offsets, enabling users to iteratively zoom and pan display of web content. Moreover, support for traditional Web page functionality, such as hyperlinks, is provided, enabling Web content to be accessed in an enhanced manner on devices having various screen resolutions, including mobile devices such as cellular phones, PDAs, and handheld devices, as well as desktop computers and laptops.

Please remove paragraph [0008]

Please replace paragraph [0053] with the following amended paragraph:

[0053] In addition to HTML and other markup and scripting language content, it is very common for web pages to include graphical content. In general, graphical content is usually stored in an image file or files that are external from the parent HTML document for the web page. For example, the parent HTML document may contain one or more embedded image tags that reference the location where those images are stored. As before, the graphic images may be stored locally, or may be stored on

Exhibit 3E

**RESOLUTION INDEPENDENT VECTOR DISPLAY OF INTERNET CONTENT
RELATED APPLICATIONS**

[0001] The present application is a Divisional Application of U.S. Application No. 09/878,097, filed June 8, 2001, which is a Continuation-in-Part of U.S. Non-Provisional Application No. 09/828,511, filed April 7, 2001, entitled "RESOLUTION INDEPENDENT VECTOR DISPLAY OF INTERNET CONTENT," the benefit of the filing date of which is claimed under 35 U.S.C. § 120. This application further claims the benefit of the filing dates of U.S. Provisional Application No. 60/211,019, filed June 12, 2000, entitled "METHOD AND SYSTEM FOR RESOLUTION INDEPENDENT DISPLAY OF HTML AND XML CONTENT" and U.S. Provisional Application No. 60/217,345, filed July 11, 2000, entitled "METHOD AND SYSTEM FOR SELECTION, RETRIEVAL, AND CONVERSION OF COMPUTER CONTENT TO VECTOR FORMAT FOR RESOLUTION INDEPENDENT DISPLAY," under 35 U.S.C. § 119(e).

COPYRIGHT NOTICE

[0002] Contained herein is material that is subject to copyright protection. The copyright owner has no objection to the facsimile reproduction of the patent disclosure by any person as it appears in the Patent and Trademark Office patent files or records, but otherwise reserves all rights to the copyright whatsoever.

BACKGROUND OF THE INVENTION

Field of the Invention

[0003] The invention relates generally to translation of Internet and World Wide Web content to scalable vector representation. More particularly, the invention relates to apparatus and methods for zoom enabling the display of content in an Internet information browser by

retrieving and translating HyperText Markup Language (HTML), eXtensible Markup Language (XML), and other Internet content to vector representations of that content.

Description of the Related Art

[0004] Text only Internet information browsers began as a project at the CERN, European Organization for Nuclear Research, facility in Geneva Switzerland. From its inception the intent was to provide a mesh or web of access to data with a common user interface. Browsers moved from the academic environment when NCSA, the National Center for Supercomputing Applications at the University of Illinois in Urbana-Champaign developed Mosaic, an Internet information browser and World Wide Web client.

[0005] Internet content is stored in multiple file formats. These formats include HTML (Hyper Text Markup Language) and XML (eXtended Markup Language) as well as graphic file format GIF (Graphics Interchange Format) and JPEG (Joint Photographic Experts Group). These four file formats constitute the majority of Internet content. Font size and resizing display area for content can alter the size of the display of Internet content in existing browsers. The majority of Internet content displays as a flat single resolution with no browser support for zoom.

[0006] Much of the Internet content has been designed for display on desktop computers with a single target resolution. Even though HTML has the ability to adapt to changes in screen resolution, major Internet content providers have chosen to create their Web pages using fixed resolution structures, such as tables. This gives them the ability to control the look and feel of their Web sites. This fixed resolution approach has evolved to the point that the fixed resolution layout of Web pages has become the most common method to brand or uniquely identify Web sites. While this fixed resolution approach is good for site branding and product differentiation it does present a daunting technical problem for display of Internet content (designed for desktop computers) on small screen, low resolution, or different aspect ratio devices, such as cell phones and hand held computers.

BRIEF SUMMARY OF THE INVENTION

[0007] A method and apparatus for supporting resolution independent vector display of Internet content is disclosed. According to one embodiment, novel client processing of Web content is provided. The client receives requested Web content and displays a vector representation of the requested Web content that substantially retains page layout and/or graphics associated with the requested Web content.

[0008] According to another embodiment, novel server processing of Web content is provided. First, the server receives a request for Web content from a client. The requested Web content is then translated into a scalable vector format to produce vector-formatted Web content corresponding to the requested Web content. The vector format enables the client to substantially retain an original page layout within a set of layouts originally intended to be associated with the requested Web content by including page layout information in a vector database. Finally, the vector-formatted Web content is provided to the client.

[0009] Other features of the present invention will be apparent from the accompanying drawings and from the detailed description that follows.

translated into a scalable vector representation (i.e., SVF, also referred to herein as “vectorized content”) through use of a proxy server 32 and sent to the requesting client. Upon being received by the client, the vectorized content is processed and rendered using a thin client to enable a user to view the content on the client device.

[0049] With reference to the flowchart of FIGURE 2A, the foregoing process is initiated by a client in a block 100, wherein the client submits a request to proxy server 32 to retrieve and convert selected content. As depicted by a transfer path 34, this comprises sending data 36, which includes content network location indicia from which the content can be retrieved and proxy server network location information by which the content request may be delivered to over Internet 24 to proxy server 32. Typically, it will be desired to retrieve a particular web page. Accordingly, the content network location indicia will comprise a URL (uniform resource locator) for the web page. Similarly, the proxy server network location information may also comprise a URL corresponding to a network access point for the proxy server. Optionally, the location information may comprise a network IP address for one or both of the content location and the proxy server location. If the content is to be retrieved from an Internet resource, the request will typically be sent using the HyperText Transfer Protocol (HTTP) over the TCP/IP transport.

[0050] Next, in a block 102, the request is received by the proxy server and the proxy server checks its cache to see if it already has the request content in its cache. If it does, it sends this cached content back to the client. If it does not have the requested content cached, the proxy server sends out a request to retrieve the content from the network resource. For illustrative purposes, it will be assumed for the present example that the desired content comprises a web page that is stored on web server 26. Typically, when the requested content comprises a web page, the content may be retrieved using conventional web content retrieval techniques, such as that employed by various modern browser clients, including Netscape Navigator and Internet Explorer. This generally comprises providing routing information, such as the URL for the web page (URL 38) to routing services provided by Internet 24,

corresponding to the embedded references will usually require additional transfer time. Furthermore, graphic content oftentimes comprises significantly larger file sizes than HTML content, leading to significant transfer times in some instances. For simplicity, the transfer of the various HTML documents and graphic files for the content request are depicted by HTML documents 52 and graphic documents 54, which are transferred over a transfer path 56.

[0056] When the HTML documents and graphic content are received by proxy server 32, a scalable vector representation of the web page is generated in a block 114 by an HTML translator 58. In brief, HTML translator 58 translates HTML, XML, and cascaded style sheet (CSS) layout content into a scalable vector representation, such as SVF. Details of the HTML translation process are contained below. In addition, the graphic images are converted into a compressed bitmap format in a block 116 by a graphics translator 60. The vectorized content 62 and compressed bitmaps 64 are then streamed back to the client (i.e., computer 18) in a block 118, as depicted by a transfer path 66. In one embodiment, the content portions are sent in separate streams using multiple connections. In another embodiment, the content portions are sent via a multiplexed stream using a single connection. As the vectorized content and compressed bitmap data are received by the client device, they are processed by a thin client 68 running on the client device, whereby a representation of the original web page content may be rendered on the client device's display screen at various user-selectable scaled resolutions and pan offsets in a block 120, thereby enabling a user to more clearly see an overview or details in the web page. Further details of the client side processing are provided below.

[0057] As discussed above, wireless clients may also access the vectorized network (e.g., web site) content provided via proxy server 24. The majority of this process is identical to that described above for land-line clients (e.g., computers 18, 20, and 22), except for provisions required for sending data to and receiving data from wireless devices. In general, most wireless devices will access the Internet via a wireless service provider (i.e., a wireless

At present, the software for the Mozilla rendering engine may be accessed via the Internet at www.mozilla.org. Accordingly, in one embodiment, the present invention uses core functionality provided by the Mozilla rendering engine source code to perform the functions of block 150, 152, and 154.

[0083] At this point, the present invention deviates substantially from the prior art by using the various object layout data generated during the pre-rendering process to generate a scalable vector representation of the original page content. First, in a block 156, a datum point is defined for the page and the bounding box for each object. For example, as shown in FIGURE 4C, a rendered page datum 262 is defined to be coincident with the upper left hand corner of the display frame of the rendered page for the web page. Generally, any point on the page may be used as the page datum – the only requirement is that the page datum that is selected is used consistently throughout the process. The use of the upper left hand corner of the display frame is advantageous since the location of the first object encountered in the HTML code for a page is located relative to this corner.

[0084] In general, the datum points for each object may also be located any place on the object, as long as the object datum points are used in a predictable manner. For example, as depicted in FIGURE 4C, various datum points for corresponding objects are defined to be coincident with the upper left hand corner of the bounding box for that object, wherein the object's datum point shares the root reference number of the object with an appended "C."

[0085] Once the page's datum point and an object's datum point are known, a vector between these points is generated for each object in a block 158. With reference to FIGURE 4D, in one embodiment, wherein the page datum point corresponds to the upper left and corner of the display frame and is assigned an XY value 266 of 0,0, the vector for a given object may be stored as the XY value of the datum point of that object relative to 0,0, such as a value of 150, 225 (ref. num. 268) for a vector 250D pointing to an object datum 250C, and a value of 150, 425 (ref. num. 270) for a vector 252D pointing to an object datum 252C. In another embodiment, each vector may be stored as XY data relative to a 0,0 datum point

corresponding to the upper left hand corner of the frame the object belongs to. For example, a vector 250D' from a frame datum 214D to object datum 250C is stored as 20, 200 (ref. num. 268'), while a vector 252D from frame datum 214D to object datum 252C is stored as 20, 425. In this embodiment, offset information for each frame relative to a known datum will also be stored, as depicted by a vector 214D.

[0086] The scalable vector representation is completed in a block 160, wherein a reference is created for each object that includes or links an object's content and attributes, such as object type (e.g., text, image), object typeface, and boundary box parameters, to the object's vector. For example, object 250B is a graphic image having a vector 250D and a bounding box that is 180 pixels high and 350 pixels wide, while object 252B is a graphic image having a vector 252D and a bounding box that includes a height of 200 pixels and a width of 350 pixels. . This enables client-side operations to be performed that only initially consider the vectors, wherein if it is determined that a vector's endpoint (and/or the bounding box corresponding to the object the vector points to) would appear off of a display, there is no need to retrieve the content and attribute data linked to the vector. This concept is explained in further detail in the following section.

[0087] It is noted that a portion of the display content produced on a client device will never contain any rendered content, as this portion is reserved for the browser's user interface. In WINDOWS™ environments, this portion will include the browser's window frame, as well as the pulldown and icon menus provided in the browser's user interface, which are depicted by a box 264 in the Figures herein.

[0088] Client-Side Software and Processing

[0089] As discussed above, the present invention supports a wide variety of clients, including land-based clients and wireless clients. Each client requires some client-side software that enables the scalable vector content data provided to it to be rendered at a user-

CLAIMS

What is claimed is:

- 1 1. A method comprising:
 - 2 retrieving Web content having an original format defining an original page layout and
 - 3 attributes of the Web content from an Internet site in response to a request of the Web content
 - 4 from a client; and
 - 5 translating the Web content from its original format into a scalable vector
 - 6 representation of the Web content, wherein the scalable vector representation of the Web
 - 7 content provides a scalable resolution-independent display of the content that substantially
 - 8 retains the original page layout and attributes of the content defined by its original format
 - 9 when rendered.
- 1 2. The method of claim 1, wherein the Web content comprises a Web page that is stored
- 2 on the Internet site in one or more markup language documents that include markup language
- 3 code defining the original page layout and attributes of the Web page.
- 1 3. The method of claim 1, wherein the Web content is translated from its original format
- 2 into the scalable vector representation on the client.
- 1 4. The method of claim 1, wherein the content is translated from its original format into
- 2 the scalable vector representation on a computing device that is remote from the client,
- 3 further comprising sending the scalable vector representation of the Web content to the client.
- 1 5. The method of claim 4, wherein the computing device that is remote from the client is
- 2 operated by the Internet site that receives a request for the Web content from the client via the
- 3 Internet, retrieves the Web content in its original format and translates it into the scalable

4 vector representation of the Web content, and returns the scalable vector representation of the
5 Web content via the Internet to the client.

1 6. The method of claim 4, wherein the computing device that is remote from the client
2 comprises a proxy server that receives the request of the Web content from the client via the
3 Internet, retrieves the Web content in its original format from the Internet site, translates the
4 content from its original format into the scalable vector representation of the Web content,
5 and sends the scalable vector representation of the Web content back to the client via the
6 Internet.

1 7. The method of claim 6, wherein the proxy server appears transparent to the client.

1 8. The method of claim 1, wherein the client comprises one of a wireless Internet device,
2 a cellular phone, a handheld computer, a desktop computer or workstation, a laptop
3 computer, an electronic billboard, or a device having a display with an unconventional aspect
4 ratio.

1 9. A computer system comprising:

2 a processor,

3 a communications device coupled to the processor, to enable the computer system to
4 be linked via the Internet to a client and an Internet site; and

5 a memory, coupled to the processor, in which a plurality of machine-executable
6 instructions are stored that when executed by the processor enable the computer system to
7 perform the operations of:

8 retrieving Web content having an original format defining an original page

9 layout and attributes of the Web content from the Internet site in response to a request

10 of the Web content from the client; and

11 translating the Web content from its original format into a scalable vector
12 representation of the Web content, wherein the scalable vector representation of the
13 Web content provides a scalable resolution-independent display of the content that
14 substantially retains the original page layout and attributes of the content defined by
15 its original format when rendered.

1 10. The computer system of claim 9, wherein execution of the plurality of the machine
2 instructions translates the Web content from its original format into the scalable vector
3 representation of the Web content by performing the operation of:

4 parsing the markup language code to determine the original page layout of the Web
5 page, wherein the original page layout defines a layout location for a plurality of text objects
6 and/or graphic image objects included in the Web page;

7 defining a primary datum corresponding to the original page layout;

8 defining an object datum corresponding to the layout location for each of said
9 plurality of text objects, graphic layout objects, and/or graphic image objects;

10 generating a vector from the primary datum to the object datum for each of said
11 plurality of text objects, graphic layout objects, and/or graphic image objects; and

12 creating a reference that links each of said plurality of text objects, graphic layout
13 objects, and/or graphic image objects to its corresponding vector.

1 11. The computer system of claim 9, wherein execution of the plurality of the machine
2 instructions further performs the operation of translating graphic image objects from an
3 original format into a scalable bitmap format.

1 12. The computer system of claim 9, wherein execution of the plurality of the machine
2 instructions further performs the operation storing attribute information pertaining to each
3 text object, said attribute information including a color and typefont for each text object.

1 13. A machine-readable medium having stored thereon a plurality of machine-executable
2 instructions that when executed by a machine performs the operations of:

3 retrieving Web content having an original format defining an original page layout and
4 attributes of the Web content from an Internet site in response to a request of the Web content
5 from the client;

6 translating the Web content from its original format into a scalable vector
7 representation of the Web content, wherein the scalable vector representation of the Web
8 content provides a scalable resolution-independent display of the content that substantially
9 retains the original page layout and attributes of the content defined by its original format
10 when rendered.

1 14. The machine-readable medium of claim 13, wherein execution of the plurality of the
2 machine instructions translates the Web content from its original format into the scalable
3 vector representation of the Web content by performing the operation of:

4 parsing the markup language code to determine the original page layout of the Web
5 page, wherein the original page layout defines a layout location for a plurality of text objects
6 and/or graphic image objects included in the Web page;

7 defining a primary datum corresponding to the original page layout;

8 defining an object datum corresponding to the layout location for each of said
9 plurality of text objects, graphic layout objects, and/or graphic image objects;

10 generating a vector from the primary datum to the object datum for each of said
11 plurality of text objects, graphic layout objects, and/or graphic image objects; and
12 creating a reference that links each of said plurality of text objects, graphic layout
13 objects, and/or graphic image objects to its corresponding vector.

1 15. The machine-readable medium of claim 13, wherein execution of the plurality of the
2 machine instructions further performs the operation of translating graphic image objects from
3 an original format into a scalable bitmap format.

1 16. The machine-readable medium of claim 13, wherein execution of the plurality of the
2 machine instructions further performs the operation storing attribute information pertaining to
3 each text object, said attribute information including a color and typefont for each text object.

1

ABSTRACT OF THE DISCLOSURE

Apparatus and methods are provided for creating resolution-independent vector display of Internet content to allow it to be scaled (zoomed) larger and smaller for better viewing or to fit any resolution or screen size. According to one embodiment, novel server processing of Web content is provided that converts Web content requested by a client to a
5 scalable vector format, such as Simple Vector Format. The vector format enables the Web content to be rendered by the client such that the rendered display substantially retains an original page layout defined in mark-up language document(s) in which the Web content is stored. In one embodiment, a proxy server receives Web content requests from a client and
10 translates the content from an original format into a scalable vector representation. The scalable vector representation is then sent to client, where it is scaled and/or offset, enabling users to zoom and/or pan the Web content.

Please replace paragraph [0065] with the following amended paragraph:

[0049] The proxy server responds to client content requests by delivering content in one of the requested formats, by retrieving the content in an appropriate format from its cache, or from an upstream content source (again using standard HTTP content negotiation features), or by translating upstream content from a supported original format to SVF or the client bitmap format.

Please replace paragraph [0068] with the following amended paragraph:

[0068] The various content translators used by the proxy server accept (via HTTP PUT) or request (driven by HTTP proxy GET/POST) content in supported, but client-unsupported, formats; and return (via HTTP PUT or GET/POST response) one or more representations of that content in a client-supported format. In the embodiments illustrated in FIGURE 1A-C, two translators are used: HTML translator 58 and image translator 60. Future content types may be accommodated by new translators, by extending existing translators to cover the new content types, or by extending the clients client's capabilities. Standard HTTP content negotiation mechanisms are used to inform the proxy server of the clients client's capabilities and expectations on each request.

Please replace paragraph [0076] with the following amended paragraph:

[0076] With reference to the flowchart of FIGURE 5, the process for translating the HTML content into a scalable vector representation proceeds as follows. The process is initiated when the proxy server receives the HTML corresponding to the parent document (and frame documents, if appropriate), whereupon a pre-rendering parsing of the HTML is performed to determine where to place the various objects on the display page in a block 150. For example, elements such as tables, column

definitions, graphic images, paragraphs and line breaks are identified. If frames are included, each frame is examined in the sequential order it appears in the HTML document, or the order in which the HTML documents corresponding to the frames in a frameset are downloaded to the browser. During further processing, the actual objects are rendered in their respective positions. Some of these objects are rendered almost immediately, such as plain text, while other objects, such as graphic objects, must first be retrieved prior to being fully-rendered. With respect to tables, there are some instances in which [[the]] all of the objects corresponding to the cells in the table must be retrieved prior to rendering any of the table, while a well-designed table can be rendered incrementally. For example, by using Column grouping, the format of the corresponding table can be quickly determined by the browser. In some instances, one or more bitmaps may actually need to be fetched before the page layout can be determined.

Please replace paragraph [0086] with the following amended paragraph:

[0086] The scalable vector representation is completed in a block 160, wherein a reference is created for each object that includes or links an object's content and attributes, such as object type (e.g., text, image), object typeface, and boundary box parameters, to the object's vector. For example, object 250B is a graphic image having a vector 250D and a bounding box that is 180 pixels high and 350 pixels wide, while object 252B is a graphic image having a vector 252D and a bounding box that includes a height of 200 pixels and a width of 350 pixels. [.] This enables client-side operations to be performed that only initially consider the vectors, wherein if it is determined that a vector's endpoint (and/or the bounding box corresponding to the object the vector points to) would appear off of a display, there is no need to retrieve the content and attribute data linked to the vector. This concept is explained in further detail in the following section.

17. (New) The method of claim 1, wherein the browser is configured to run on at least one of a desktop computer, workstation, and laptop computer.

18. (New) The method of claim 1, wherein the browser is configured to run on a device having a display with an unconventional aspect ratio.

19. (New) The method of claim 1, further comprising:

parsing markup language code to determine the original page layout of the Web page, wherein the original page layout defines a layout location for a plurality of objects, including text objects, graphic layout objects, and/or graphic image objects included in the Web page;

defining a primary datum corresponding to the original page layout; and,
for each object,

defining an object datum corresponding to the layout location for the object;

generating a vector from the primary datum to the object datum for the object; and

creating a reference that links the object to its corresponding vector.

20. (New) The method of claim 19, further comprising translating graphic image objects from an original format into a scalable format.

21. (New) The method of claim 19, further comprising storing attribute information pertaining to each text object, said attribute information including a color and typefont for each text object.

22. (New) The method of claim 6, wherein the requester is an Internet site.

Please replace the BRIEF SUMMARY OF THE INVENTION section with the following paragraphs:

[0007] In accordance with aspects of the invention, mobile devices enabled to support resolution-independent scalable display of Internet (Web) content to allow Web pages to be scaled (zoomed) and panned for better viewing on smaller screen sizes are disclosed. The mobile devices employ novel processing of original Web content, including HTML-based content, XML, cascade style sheets, etc. to generate scalable content. The scalable content and/or data derived therefrom are then employed to enable the Web content to be rapidly rendered, zoomed, and panned. Moreover, the rendered displays provide substantially the same or identical layout as the original Web page, enabling users to easily navigate to selected content and features on familiar Web pages. Display lists may also be employed to provide further enhancements in rendering speed. Additionally, hardware-based programmed logic may also be employed to facilitate various operations.

[0008] According to further aspects, some mobile devices may employ touch-sensitive display screens that enable users to provide various inputs to control display of content within Web pages. Exemplary user inputs include tap-based inputs to selectively zoom in on columns, images, and paragraphs. Users can also define a window to zoom in on via the touch-sensitive display.

[0009] Other features of the present invention will be apparent from the accompanying drawings and from the detailed description that follows.

Abstract

Please replace the Abstract with the following:

Mobile devices enabled to support resolution-independent scalable display of Internet (Web) content to allow Web pages to be scaled (zoomed) and panned for better viewing on smaller screen sizes. The mobile devices employ software-based processing of original Web content, including HTML-based content, XML, cascade style sheets, etc. to generate scalable content. The scalable content and/or data derived therefrom are then employed to enable the Web content to be rapidly rendered, zoomed, and panned. Moreover, the rendered displays provide substantially the same or identical layout as the original Web page, enabling users to easily navigate to selected content and features on familiar Web pages. Display lists may also be employed to provide further enhancements in rendering speed. Additionally, hardware-based programmed logic may be employed to facilitate various operations.

In the Claims

Please cancel claims 1-39 without prejudice.

Please add new claims 40 – 141 as follows.

40. (New) A mobile device, comprising:

a processor,

a wireless communications device operatively coupled to the processor, to facilitate communication with a network via which Web content may be accessed;

a display;

a memory, operatively coupled to the processor; and

storage means, operatively coupled to the processor, in which a plurality of instructions are stored that when executed by the processor enable the mobile device to perform operations including,

enabling a user to request access to a Web page having an original format comprising HTML-based Web content defining an original page layout of content on the Web page;

retrieving at least a portion of the HTML-based Web content associated with the Web page;

translating the at least a portion of the HTML-based Web content to produce scalable page layout information; and

employing the scalable page layout information and/or data derived therefrom to,

render at least a portion of the Web page on the display using a first scale factor; and

re-render the Web page in response to associated user inputs to enable a user to zoom in and out a display of the Web page.

41. (New) The mobile device of claim 40, wherein the device comprises a mobile phone.

42. (New) The mobile device of claim 40, wherein the device comprises one of a Personal Digital Assistant (PDA), handheld computer, notebook computer, or laptop computer.

43. (New) The mobile device of claim 40, wherein execution of the instructions performs further operations comprising enabling the user to zoom in on a user-selectable portion of a display of the Web page in response to a corresponding user interface input.

44. (New) The mobile device of claim 43, wherein the display of the Web page is re-rendered substantially in real-time to effect zooming operations.

45. (New) The mobile device of claim 40, wherein the Web content includes at least one hyperlink, and wherein execution of the instructions performs further operations comprising:

enabling the user to select the hyperlink; and, in response thereto,

retrieving and translating HTML-based Web content associated with the hyperlink to produce additional scalable page layout information; and

employing the additional scalable page layout information and/or data derived therefrom to render the Web content associated with the hyperlink on the display.

46. (New) The mobile device of claim 40, wherein execution of the instructions performs further operations comprising:

parsing HTML-based code to determine the original page layout of display content within the Web page, wherein the original page layout defines a layout location

for a plurality of objects, including text objects, graphic layout objects, and/or graphic image objects included in the Web page;

defining a primary datum corresponding to the original page layout; and,

for each object,

defining an object datum corresponding to the layout location for the object;

generating a vector from the primary datum to the object datum for the object; and

creating a reference that links the object to its corresponding vector.

47. (New) The mobile device of claim 40, wherein execution of the instructions performs further operations comprising enabling the Web content to be displayed at different resolutions by,

generating scalable content via use of the scalable page layout information; and

scaling the scalable content to re-render the display in response to associated user inputs.

48. (New) The mobile device of claim 40, wherein execution of the instructions performs further operations comprising returning the display of the Web content to a previous view in response to a corresponding user input made.

49. (New) The mobile device of claim 40, wherein execution of the instructions performs further operations comprising enabling a user to pan a display of the Web content in response to a corresponding user input.

50. (New) The mobile device of claim 49, wherein execution of the instructions performs further operations comprising enabling the display of the Web content to be panned substantially in real-time.

51. (New) The mobile device of claim 40, wherein the page layout of the Web page is defined to have an original aspect ratio, and wherein the scalable page layout information and/or data derived therefrom is scaled to render a display having a different aspect ratio.

52. (New) The mobile device of claim 40, wherein execution of the instructions performs further operations comprising enabling a user to view a column of the Web content at a higher resolution than a current resolution via a corresponding user interface input, wherein in response thereto, the display is re-rendered such that content corresponding to the selected column is displayed substantially across the display.

53. (New) The mobile device of claim 52, wherein the content of the column is reformatted to fit characteristics of the display when the display is re-rendered.

54. (New) The mobile device of claim 52, wherein the display comprises a touch-sensitive display, and wherein the corresponding user interface input comprises tapping on the column.

55. (New) The mobile device of claim 40, wherein the Web content includes at least one image, and wherein execution of the instructions performs further operations comprising enabling a user to view an image at a higher resolution than a current resolution via a corresponding user interface input, wherein in response thereto, the display is re-rendered such that the image is displayed substantially across at least one of a width and height of a display area of the display.

56. (New) The mobile device of claim 52, wherein the display comprises a touch-sensitive display, and wherein the corresponding user interface input comprises tapping on the image.

57. (New) The mobile device of claim 40, wherein execution of the instructions performs further operations comprising enabling a user to view a paragraph of the Web content at a higher resolution than a current resolution via a corresponding user interface input, wherein in response thereto, the display is re-rendered such that content corresponding to the selected paragraph is displayed substantially across at least one of a width and height of a display area of the display.

58. (New) The mobile device of claim 57, wherein the content of the paragraph is reformatted to fit characteristics of the display area when the display is re-rendered.

59. (New) The mobile device of claim 40, wherein the Web page includes text, layout attributes, and images, and wherein execution of the instructions performs further operations comprising:

- receiving content corresponding to the text and layout attributes via a first connection; and

- receiving content corresponding to at least one image via a second connection.

60. (New) The mobile device of claim 40, wherein execution of the instructions performs further operations comprising:

- generating a display list of vectors derived, at least in part, via use of the scalable page layout information; and

- employing the display list to re-render the display of the Web page.

61. (New) The mobile device of claim 40, wherein execution of the instructions performs further operations comprising:

- parsing HTML-based code corresponding to the received Web content to determine the original page layout of the content on the Web page;

- logically grouping selected content into objects;

- defining a primary datum corresponding to the original page layout; and,

retrieving content associated with that object bounding box;
clipping portions of the content that are outside the virtual display bounding box; and
applying an appropriate scale factor to the content within the virtual display bounding box to render the content on the display.

65. (New) The mobile device of claim 40, wherein the Web content includes text content, and wherein execution of the instructions performs further operations comprising scaling a scalable font to render the text content at different scale factors.

66. (New) The mobile device of claim 40, wherein the original format of the Web page defines a height and width for the Web page, and wherein execution of the instructions performs further operations comprising:

determining an applicable scale factor to display at least one of the width and height of the Web page substantially across a display area of the display; and
employing the scale factor that is determined as the first scale factor.

67. (New) The mobile device of claim 40, wherein at least a portion of the instructions comprise Java-based instructions configured to be executed on a Java virtual machine.

68. (New) The mobile device of claim 40, wherein the Web content includes cascaded style sheets.

69. (New) The mobile device of claim 40, wherein the scalable page layout information is vector-based.

70. (New) The method of claim 40, wherein the network comprises a mobile service provider network.

71. (New) A wireless device, comprising:

- processing means,
- wireless communications means, to facilitate wireless communication with a network via which Web content may be accessed;
- a display;
- memory; and
- storage means, in which a plurality of instructions are stored that when executed by the processing means enable the wireless device to perform operations including,
 - rendering a browser interface via which a user is enabled to request access to a Web page, the Web page including associated Web content having an original format defining an original page layout and attributes of content on the Web page;
 - retrieving and translating at least a portion of the Web content from its original format into scalable content that supports a scalable resolution-independent display of the content that substantially retains the original page layout and attributes of the content defined by its original format when rendered;
 - and
 - employing the scalable content to render at least a portion of the Web page on the display using a first scale factor.

72. (New) The wireless device of claim 71, wherein execution of the instructions performs further operations comprising enabling the user to zoom in on a user-selectable portion of a display of the Web page in response to a corresponding user interface input.

73. (New) The wireless device of claim 72, wherein the display of the Web page is re-rendered substantially in real-time to effect zooming operations.

77. (New) The wireless device of claim 71, wherein execution of the instructions performs further operations comprising returning the display of the Web content to a previous view in response to a corresponding user input.

78. (New) The wireless device of claim 71, wherein execution of the instructions performs further operations comprising enabling a user to pan a display of the Web content in response to a corresponding user input.

79. (New) The wireless device of claim 78, wherein execution of the instructions performs further operations comprising enabling the display of the Web content to be panned substantially in real-time.

80. (New) The wireless device of claim 71, wherein the page layout of the Web page is defined to have an original aspect ratio, and wherein the scalable content is scaled when rendered so as to produce a display having a different aspect ratio.

81. (New) The wireless device of claim 71, wherein the display comprises a touch-sensitive display, and wherein execution of the instructions performs further operations comprising enabling a user to view a column of the Web content at a higher resolution than a current resolution by tapping on the column via the touch-sensitive display, wherein in response thereto, the display is re-rendered such that content corresponding to the selected column is displayed substantially across the touch-sensitive display.

82. (New) The wireless device of claim 81, wherein the content of the column is reformatted to fit characteristics of the touch-sensitive display when the display is re-rendered.

83. (New) The wireless device of claim 71, wherein the display comprises a touch-sensitive display, and wherein the Web content includes at least one image, and wherein execution of the instructions performs further operations comprising enabling a

user to view an image at a higher resolution than a current resolution by tapping on the image via the touch-sensitive display, wherein in response thereto, the display is re-rendered such that the image is displayed substantially across at least one of a width and height of a display area of the touch-sensitive display.

84. (New) The wireless device of claim 71, wherein the display comprises a touch-sensitive display, and wherein execution of the instructions performs further operations comprising enabling a user to view a paragraph of the Web content at a higher resolution than a current resolution by tapping on the paragraph via the touch-sensitive display, wherein in response thereto, the display is re-rendered such that content corresponding to the selected paragraph is displayed substantially across at least one of a width and height of a display area of the touch-sensitive display.

85. (New) The wireless device of claim 84, wherein the content of the paragraph is reformatted to fit characteristics of the display area when the display is re-rendered.

86. (New) The wireless device of claim 71, wherein the Web page includes text, layout attributes, and images, and wherein execution of the instructions performs further operations comprising:

receiving content corresponding to the text and layout attributes via a first connection; and

receiving content corresponding to at least one image via a second connection.

87. (New) The wireless device of claim 71, wherein execution of the instructions performs further operations comprising:

generating a vector-based display list associated with the scalable content; and

employing the display list to re-render the display at different scale factors to enable rapid zooming of the Web page.

to a rendered display of the Web content desired by a user;

determining a virtual display bounding box for the virtual display associated with the first scale factor and offset;

identifying object bounding boxes having at least a portion falling within the virtual display bounding box; and,

for each of such object bounding boxes,

retrieving content associated with that object bounding box;

clipping portions of the content that are outside the virtual display bounding box; and

applying an appropriate scale factor to the content within the virtual display bounding box to render the content on the display.

92. (New) The wireless device of claim 71, wherein the scalable content includes scalable text content, and wherein execution of the instructions performs further operations comprising scaling a scalable font to render the scalable text content.

93. (New) The wireless device of claim 71, wherein the original format of the Web page defines a height and width for the Web page, and wherein execution of the instructions performs further operations comprising:

determining an applicable scale factor to display at least one of the width and height of the Web page substantially across a display area of the display; and

employing the scale factor that is determined as the first scale factor.

94. (New) The wireless device of claim 71, wherein at least a portion of the instructions comprise Java-based instructions configured to be executed on a Java virtual machine.

95. (New) The wireless device of claim 71, wherein the device comprises a mobile phone.

phone.

101. (New) The mobile device of claim 99, wherein the device comprises one of a Personal Digital Assistant (PDA), handheld computer, notebook computer, or laptop computer.

102. (New) The mobile device of claim 99, wherein execution of the instructions performs further operations comprising enabling the user to zoom in on a user-selectable portion of a display of the Web page in response to a user interface input made via the touch-sensitive display.

103. (New) The mobile device of claim 102, wherein the user interface input enables the user to define a window of a current view of the Web page on which to zoom in on.

104. (New) The mobile device of claim 99, wherein the display of the Web page is re-rendered substantially in real-time to effect zooming operations.

105. (New) The mobile device of claim 99, wherein the Web page includes at least one hyperlink, and wherein execution of the instructions performs further operations comprising:

enabling the user to select the hyperlink via the touch-sensitive display; and, in response thereto,

retrieving and processing HTML-based Web content associated with the hyperlink to produce additional scalable content; and

employing the additional scalable content and/or data derived therefrom to render the Web content associated with the hyperlink on the touch-sensitive display.

106. (New) The mobile device of claim 99, wherein at least a portion of the scalable content comprises scalable vector-based content.

107. (New) The mobile device of claim 99, wherein execution of the instructions performs further operations comprising returning the display of the Web page to a previous view in response to a corresponding user input made via the touch-sensitive display.

108. (New) The mobile device of claim 99, wherein execution of the instructions performs further operations comprising enabling a user to pan a display of the Web content in response to a corresponding user input made via the touch-sensitive display.

109. (New) The mobile device of claim 108, wherein execution of the instructions performs further operations comprising enabling the display of the Web content to be panned substantially in real-time.

110. (New) The mobile device of claim 99, wherein the page layout of the Web page is defined to have an original aspect ratio, and wherein the scalable content and/or data derived therefrom is scaled to render a display having a different aspect ratio.

111. (New) The mobile device of claim 99, wherein execution of the instructions performs further operations comprising enabling a user to view a column of the Web content at a higher resolution than a current resolution by tapping on the column via the touch-sensitive display, wherein in response thereto, the display is re-rendered such that content corresponding to the selected column is displayed substantially across the touch-sensitive display.

112. (New) The mobile device of claim 111, wherein the content of the column is reformatted to fit characteristics of the touch-sensitive display when the display is re-rendered.

113. (New) The mobile device of claim 99, wherein the Web content includes at least one image, and wherein execution of the instructions performs further operations

comprising enabling a user to view an image at a higher resolution than a current resolution by tapping on the image via the touch-sensitive display, wherein in response thereto, the display is re-rendered such that the image is displayed substantially across at least one of a width and height of a display area of the touch-sensitive display.

114. (New) The mobile device of claim 99, wherein execution of the instructions performs further operations comprising enabling a user to view a paragraph of the Web content at a higher resolution than a current resolution by tapping on the paragraph via the touch-sensitive display, wherein in response thereto, the display is re-rendered such that content corresponding to the selected paragraph is displayed substantially across at least one of a width and height of a display area of the touch-sensitive display.

115. (New) The mobile device of claim 114, wherein the content of the paragraph is reformatted to fit characteristics of the display area when the display is re-rendered.

116. (New) The mobile device of claim 99, wherein the Web page includes text, layout attributes, and images, and wherein execution of the instructions performs further operations comprising:

receiving content corresponding to the text and layout attributes via a first connection; and

receiving content corresponding to at least one image via a second connection.

117. (New) The mobile device of claim 99, further comprising dynamic memory having at least a portion employed for rendering purposes, wherein execution of the instructions performs further operations comprising:

building a display list via use of the scalable content and rendering display list content on a virtual display in the dynamic memory; and

scaling the display list content to re-render the display of the Web page.

determining a first scale factor and offset in response to one or more corresponding user inputs defining a user-selectable zoom level and pan corresponding to a rendered display of the Web page desired by a user;

determining a virtual display bounding box for the virtual display associated with the first scale factor and offset;

identifying object bounding boxes having at least a portion falling within the virtual display bounding box; and,

for each of such object bounding boxes,

retrieving content associated with that object bounding box;

clipping portions of the content that are outside the virtual display bounding box; and

applying an appropriate scale factor to the content within the virtual display bounding box to render the content on the display.

122. (New) The mobile device of claim 99, wherein the scalable content includes scalable text content, and wherein execution of the instructions performs further operations comprising scaling a scalable font to render the scalable text content.

123. (New) The mobile device of claim 99, wherein the original format of the Web page defines a height and width for the Web page, and wherein execution of the instructions performs further operations comprising:

determining an applicable scale factor to display at least one of the width and height of the Web page substantially across a display area of the touch-sensitive display; and

employing the scale factor to render the display area.

124. (New) The mobile device of claim 99, wherein at least a portion of the instructions comprise Java-based instructions configured to be executed on a Java

made via the touch-sensitive display means to enable the user to zoom in and out a display of the Web page.

129. (New) The mobile device of claim 128, wherein the processing means includes a general-purpose processor.

130. (New) The mobile device of claim 128, wherein at least a portion of the programmed circuit means is embodied as a special-purpose processor.

131. (New) The mobile device of claim 128, wherein execution of the instructions performs further operations comprising enabling the user to zoom in on a user-selectable portion of a display of the Web page in response to a user interface input made via the touch-sensitive display.

132. (New) The mobile device of claim 131, wherein the user interface input enables the user to define a window of a current view of the Web page on which to zoom in on.

133. (New) The mobile device of claim 128, wherein the display of the Web page is re-rendered substantially in real-time to effect zooming operations.

134. (New) The mobile device of claim 128, wherein execution of the instructions performs further operations comprising enabling a user to pan a display of the Web content in response to a corresponding user input made via the touch-sensitive display.

135. (New) The mobile device of claim 134, wherein execution of the instructions performs further operations comprising enabling the display of the Web content to be panned substantially in real-time.

136. (New) The mobile device of claim 128, wherein the Web content includes at least one image, and wherein execution of the instructions performs further operations comprising enabling a user to view an image at a higher resolution than a current

resolution by tapping on the image via the touch-sensitive display, wherein in response thereto, the display is re-rendered such that the image is displayed substantially across at least one of a width and height of a display area of the touch-sensitive display.

137. (New) The mobile device of claim 128, further comprising dynamic memory having at least a portion employed for rendering purposes, wherein execution of the instructions performs further operations comprising:

building a display list via use of the scalable content and rendering display list objects on a virtual display in the dynamic memory; and

scaling display list objects to re-render the display of the Web page.

138. (New) The mobile device of claim 128, wherein the network comprises a mobile service provider network.

139. (New) The wireless device of claim 128, wherein the device comprises a mobile phone.

140. (New) The wireless device of claim 128, wherein the device comprises one of a Personal Digital Assistant (PDA), handheld computer, or handheld device.

141. (New) The wireless device of claim 128, wherein a portion of the scalable content comprises vector-based content.

Art Unit: 2176

	<p>instructions are stored that when executed by the processor enable the device to perform operations including, (see claim 25), and The device of claim 25, wherein the device comprises a wireless hand-held device (see claim 43)</p>
<p>f) scaling the scalable content to render the Web page on the display such that the original width of the Web page is rendered to fit substantially across the display.</p>	<p>scalable vector representation of at least a portion of the Web page that supports a scalable resolution-independent display of the Web content that substantially retains the original page layout and attributes of the Web page when it is rendered on the client device (see claim 15), and rendering the vector-formatted Web content on the client device using a first scaling factor to generate a first display of the Web content on the client device (see claim 15).</p>

In addition, it is obvious that the claimed limitation cites above, (b), (d), and (f) the original width and height of the Web page is rendered to fit substantially across the display is equivalent to and equated to receiving, at the client device, vector-formatted Web content comprising a scalable vector representation of at least a portion of the Web page that supports a scalable resolution-independent display of the Web content that substantially retains *the original page layout and attributes of the Web page* when it is rendered on the client device (see claim 15), and rendering the vector-formatted Web content on the client device using a first scaling factor to generate a first display of the Web content on the client device (see claim 15), to produce a method of Resolution independent vector displaying on Internet content of wireless hand-help devices of copending Application No. 11/045,757.

Art Unit: 2176

Allowable Subject Matter

Claims 88-91, 118-121, 125-126, 158-159, objected to as being dependent upon a rejected base claims, but would be allowable if rewritten in independent form including all of the limitations of the base claim and any intervening claims and rewritten to overcome 35 USC 101, and 35 USC 112, and Terminal Disclaimer.

Claims Rejection – 35 U.S.C. 103

The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

(a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negated by the manner in which the invention was made.

Claims 71-87, 92, 95-117, 122-123, 127-157, and 160-179 are rejected under 35 U.S.C. 103(a) as being unpatentable over Chithambaram et al. US006674445B1 -Provisional No.60/159,069 filed 10/12/1999 [hereinafter “Chithambaram”], in view of Roy et al. US006642925B2 -Continuation of No.08/757,706 filed 10/30/1996 [hereinafter “Roy”].

Independent claim 71, Chithambaram teaches:

Art Unit: 2176

A wireless device, a display, a memory, and storage means,

(See Chithambaram Column 3, Lines 65-67, discloses a personal digital assistance (PDA).)

Comprising: wireless communications means, to facilitate wireless communication with a network via which Web content may be accessed; a display; memory; and storage means, in which a plurality of instructions are stored that when executed by the processing means enable the wireless device to perform operations including, rendering a browser interface via which a user is enabled to request access to a Web page,

(See Chithambaram Fig. 1 Column 5, Lines 25-30, discloses a hardware and software environment for the architecture uses a network/Internet 118 to connect technicians utilizing clients such as a thin client 102 (e.g. a PDA, WINCE, or PALM device) or a thick client 104 (e.g., a computer system running a browser) to server computers 106.)

retrieving, via the wireless communication means, and translating at least a portion of the HTML-based Web content from its original format into scalable content that supports a scalable resolution-independent display of the content Web page that substantially retains the original page layout and attributes of the content defined by its original format when rendered; and, scaling the scalable

content to render the Web page on the display such that the original width of the Web page is rendered to fit substantially across the display.

(See Chithambaram Fig. 1 Column 5, Lines 25-30, discloses a hardware and software environment for the architecture uses a network/Internet 118 to connect technicians utilizing clients such as a thin client 102 (e.g. a PDA, WINCE, or PALM device) or a thick client 104 (e.g., a computer system running a browser) to server computers 106.

Also see Chithambaram Column 4 Line 60 → Column 5, Line 10, teaching the indexing raster (i.e. Raster maps provide multiple zoom levels with each zoom level, by scaling existing raster tiles) and vector based for interacting and highlighting with user objects.

Also see Chithambaram Column 6 Lines 55-65, teaching the SVG (Scalable Vector Graphics) allows vector graphic shapes (e.g., paths consisting of straight lines and curves), images, and text. Graphical objects can be grouped, styled, transformed, and composite into previously rendered objects.

Also, see Chithambaram Column 8 Lines 30-65, teaching the offset for location of the object is obtained and the offset is encoded using bounding box to zoom in and filtering out the unwanted object for display on the PDA.)

In addition, Chithambaram does not expressly teach, but Roy teaches:

Art Unit: 2176

the Web page including associated HTML-based Web content having an original format defining an original width and height of the Web page and an original page layout and attributes of content on the Web page;

(See Roy Column 10, Lines 1-15, discloses an HTML document using specify the width and height of the map in pixels with the WIDTH=NNN and HEIGHT=NNN parameters.

For example: `<EMBED SRC="http://www.mapguide.com/map pictures/usa.mwf" WIDTH=300HEIGHT=200>`. This entry displays a map of the US in the current document. The map picture is 300x200 pixels in size.)

It would have been obvious to a person of ordinary skill in the art at the time the invention was made to have modified the teaching Chithambaram, to include the Web page including associated HTML-based Web content having an original format defining an original width and height of the Web page and an original page layout and attributes of content on the Web page as taught by Roy to produce a predictable result, as evidence, using Roy's specify the width and height of the map in pixels with the WIDTH=NNN and HEIGHT=NNN parameters with Chithambaram's SVG (Scalable Vector Graphics) and the offset for location of the object is obtained and the offset is encoded using bounding box of to zoom in and filtering out the unwanted object for display on the PDA (see Chithambaram Column 6 Lines 55-65, and also see Chithambaram Column 8 Lines 30-65).

Independent claim 99,

Art Unit: 2176

the rejection of claim 71 is fully incorporated, which cites above, and is similarly rejected under the same rationale. In addition, Chithambaram teaches:

**employing the scalable content and/or data derived therefrom to,
render the Web page on the touch-sensitive display;**

(See Chithambaram Column 6 Lines 55-65, teaching the SVG (Scalable Vector Graphics) allows vector graphic shapes (e.g., paths consisting of straight lines and curves), images, and text. Graphical objects can be grouped, styled, transformed, and composite into previously rendered objects.

Also, see Chithambaram Column 8 Lines 30-65, teaching the offset for location of the object is obtained and the offset is encoded using bounding box to zoom in and filtering out the unwanted object for display on the PDA (i.e. It is noted PDA is inherently includes the tough sensitive display as claimed)

**and re-render the Web page in response to associated user inputs to
enable the user to zoom in and out a display of the Web page;**

(See Chithambaram Column 8 Lines 30-65, teaching the offset for location of the object is obtained and the offset is encoded using bounding box to zoom in and filtering out the unwanted object for display on the PDA.)

Independent claim 128,

the rejection of claims 71 and 99 are fully incorporated, which cite above, and are similarly rejected under the same rationale.

Independent claim 143,

Art Unit: 2176

the rejection of claims 71 and 99 are fully incorporated, which cite above, and are similarly rejected under the same rationale.

Independent claim 174,

the rejection of claims 71 and 99 are fully incorporated, which cite above, and are similarly rejected under the same rationale. In addition, Chithambaram teaches:

translating at least a portion of the HTML-based Web content from its original format into scalable content that supports a scalable resolution-independent display of the Web page that substantially retains the original page layout and attributes of the content defined by its original format when rendered;

(See Chithambaram Fig. 1 Column 5, Lines 25-30, discloses a hardware and software environment for the architecture uses a network/Internet 118 to connect technicians utilizing clients such as a thin client 102 (e.g. a PDA, WINCE, or PALM device) or a thick client 104 (e.g., a computer system running a browser) to server computers 106.

Also see Chithambaram Column 4 Line 60 → Column 5, Line 10, teaching the indexing raster (i.e. Raster maps provide multiple zoom levels with each zoom level, by scaling existing raster tiles) and vector based for interacting and highlighting with user objects.

Also see Chithambaram Column 6 Lines 55-65, teaching the SVG (Scalable Vector Graphics) allows vector graphic shapes (e.g., paths consisting of straight lines

Art Unit: 2176

and curves), images, and text. Graphical objects can be grouped, styled, transformed, and composite into previously rendered objects.

Also, see Chithambaram Column 8 Lines 30-65, teaching the offset for location of the object is obtained and the offset is encoded using bounding box to zoom in and filtering out the unwanted object for display on the PDA.)

and employing the scalable content to render the Web page on the display using a first scale factor; and enabling the Web page to be displayed at a different resolution by scaling the scalable content using a second scale factor to re-render the display;

(See Chithambaram Column 6 Lines 55-65, teaching the SVG (Scalable Vector Graphics) allows vector graphic shapes (e.g., paths consisting of straight lines and curves), images, and text. Graphical objects can be grouped, styled, transformed, and composite into previously rendered objects.

Also, see Chithambaram Column 8 Lines 30-65, teaching the offset for location of the object is obtained and the offset is encoded using bounding box to zoom in and filtering out the unwanted object for display on the PDA.)

Art Unit: 2176

Claim 72, Chithambaram teaches:

enabling the user to zoom in on a user-selectable portion of a display of the Web page in response to a corresponding user interface input.

(See Chithambaram Column 7 Lines 4-6 and Column 8 Lines 30-65, teaching using bounding box to zoom in and filtering out the unwanted object for display on the PDA.)

Claim 73, Chithambaram teaches:

wherein the display of the Web page is re-rendered to effect zooming operations.

(See Chithambaram Column 7 Lines 4-6 and Column 8 Lines 30-65, teaching using bounding box to zoom in and filtering out the unwanted object for display on the PDA.)

In addition, Chithambaram does not expressly teach, but Roy teaches:

substantially in real-time,

(See Roy Column 3, Lines 40-45, discloses real-time access to dynamic map pictures and associated map data through a Web browser interface suitable for a wide range of users.

See also Roy at Column 2, Lines 20-31, teaching the vector-based data includes Zoom in feature.

It would have been obvious to a person of ordinary skill in the art at the time the invention was made to have modified the teaching Chithambaram, to include the Web page is re-rendered to effect zooming operations in real time as taught by Roy to produce a predictable result, as evidence, using Roy's re-rendered to effect zooming

Art Unit: 2176

operations in real time and the offset for location of the object is obtained and the offset is encoded using bounding box of to zoom in and filtering out the unwanted object for display on the PDA (see Chithambaram Column 6 Lines 55-65, and also see Chithambaram Column 8 Lines 30-65).

Claim 74, Chithambaram teaches:

wherein the Web content page includes at least one hyperlink,

(See Chithambaram Column 10 Lines 20-25, teaching attributes stored in records 532-540 may include a URL, an id, and other relevant information.)

and wherein execution of the instructions performs further operations comprising: enabling the user to select the hyperlink; and, in response thereto, retrieving and translating Web content associated with the hyperlink.

(See Chithambaram Column 10 Lines 20-25, teaching attributes stored in records 532-540 may include a URL, an id, and other relevant information.)

to produce additional scalable content; and employing the additional scalable content to render the Web content associated with the hyperlink on the display.

(See Chithambaram Fig. 1 Column 5, Lines 25-30, discloses a hardware and software environment for the architecture uses a network/Internet 118 to connect technicians

Art Unit: 2176

utilizing clients such as a thin client 102 (e.g. a PDA, WINCE, or PALM device) or a thick client 104 (e.g., a computer system running a browser) to server computers 106.)

Claim 75, Chithambaram teaches:

parsing markup language code to determine the original page layout of display content within the Web page, wherein the original page layout defines a layout location for a plurality of objects, including text objects, graphic layout objects, and/or graphic image objects included in the Web page; defining a primary datum corresponding to the original page layout; and, for each object, defining an object datum corresponding to the layout location for the object; generating a vector from the primary datum to the object datum for the object; and creating a reference that links the object to its corresponding vector

(See Chithambaram Fig. 1 Column 5, Lines 25-30, discloses a hardware and software environment for the architecture uses a network/Internet 118 to connect technicians utilizing clients such as a thin client 102 (e.g. a PDA, WINCE, or PALM device) or a thick client 104 (e.g., a computer system running a browser) to server computers 106.

Also see Chithambaram Column 4 Line 60 → Column 5, Line 10, teaching the indexing raster (i.e. Raster maps provide multiple zoom levels with each zoom level, by scaling existing raster tiles) and vector based for interacting and highlighting with user objects.

Art Unit: 2176

Also see Chithambaram Column 6 Lines 55-65, teaching the SVG (Scalable Vector Graphics) allows vector graphic shapes (e.g., paths consisting of straight lines and curves), images, and text. Graphical objects can be grouped, styled, transformed, and composite into previously rendered objects.

Also, see Chithambaram Column 8 Lines 30-65, teaching the offset for location of the object is obtained and the offset is encoded using bounding box to zoom in and filtering out the unwanted object for display on the PDA.)

Claim 76, Chithambaram teaches:

enabling the Web page to be displayed at different resolutions by scaling the scalable content to re-render the display in response to associated user inputs.

(See Chithambaram Column 6 Lines 55-65, teaching the SVG (Scalable Vector Graphics) allows vector graphic shapes (e.g., paths consisting of straight lines and curves), images, and text. Graphical objects can be grouped, styled, transformed, and composite into previously rendered objects.

Also, see Chithambaram Column 8 Lines 30-65, teaching the offset for location of the object is obtained and the offset is encoded using bounding box to zoom in and filtering out the unwanted object for display on the PDA.)

Art Unit: 2176

scaling existing raster tiles) and vector based for interacting and highlighting with user objects.

Also, see Chithambaram Column 8 Lines 30-65, teaching the offset for location of the object is obtained and the offset is encoded using bounding box to zoom in and filtering out the unwanted object for display on the PDA. Using the broadest reasonable interpretation, Examiner equates the claimed **view a column of the Web at a higher resolution than a current resolution via a corresponding user input** as equivalent to multiple zoom levels with each zoom level, by scaling existing raster tiles) and vector based for interacting and highlighting with user objects as taught by Chithambaram.


Claim 82, Chithambaram teaches:

wherein the content of the column is reformatted to fit characteristics of the display when the display is re-rendered.

(See Chithambaram Column 4 Line 60 → Column 5, Line 10, teaching the indexing raster (i.e. Raster maps provide multiple zoom levels with each zoom level, by scaling existing raster tiles) and vector based for interacting and highlighting with user objects.

Also see Chithambaram Column 6 Lines 55-65, teaching the SVG (Scalable Vector Graphics) allows vector graphic shapes (e.g., paths consisting of straight lines and curves), images, and text. Graphical objects can be grouped, styled, transformed, and composite into previously rendered objects.

Continuation of Substance of Interview including description of the general nature of what was agreed to if an agreement was reached, or any other comments: Applicant requested interview to discuss the claimed invention, directing said limitation toward the invention viewpoint, such as translating at least a portion of the HTML-based Web content from its original format into scalable content that supports a scalable resolution-independent display of the Web page that substantially retains the original page layout and attributes of the content defined by its original format when rendered at the mobile device; and not at the server; and also discussed 112 second rejection (i.e. will be consider upon the official submission of the responses to the First Office Action,) 101 rejection (i.e. the Examiners agree to withdraw the 101 rejection, which set forth in the First Office Action,) and Obviousness Double Patent (ODP) (i.e. will be consider upon the official submission of the responses to the First Office Action); and discussed the difference with the prior art. Examiners would like to thank you Mr. Burnett and Mr. Rohrabought for a very informative interviewed, and also explained the Examiners' position and advised Applicant's attorneys that arguments are noticed, and will be consider upon the official submission of the responses to the First Office Action, Therefore further consideration will be considered upon submission of an official amendment.



12/7/07

that includes the filing date for each provisional application.

Examiner Interview

An in-person examiner interview was conducted at the USPTO on December 3, 2007. The attendees included Examiner Quoc A. Tran, SPE Doug Hutton, Inventor Gary Rohrabough, and attorney representative R. Alan Burnett.

Demonstration of Device

During the interview, a demonstration of a device and software based on the underlying teachings of the claimed invention was presented. The demonstration device was a Toshiba Pocket PC running a version of the SoftView™ browser, as discussed in further detail below. Originally, a demonstration of various emulators running on a laptop computer were attempted – however, since there was no network connectivity, the emulators could not retrieve Web pages from the Internet, and thus the demonstration could not be completed. Under the Toshiba Pocket PC demonstration, the demonstration Web pages were preloaded on the device (in preparation for not having network connectivity). Inventor Gary Rohrabough demonstrated the SoftView™ browser's ability to scale and render Web pages to fit the Toshiba's display, selectively zoom on user-defined windows, images, columns, and paragraphs, and generally zoom and pan Web pages and perform browser functions such as activating hyperlinks. Claims corresponding to each of these features are included in the present application.

Discussion of 35 U.S.C. § 112, Second Paragraph and 35 U.S.C. § 101 rejections.

Prior to the meeting and as discussed below, Examiner Tran believed the claims were directed to an apparatus (device) and method of using the apparatus. It was made clear that the claims were directed simply to an apparatus that was configured to perform various operations via execution of corresponding instructions. As discussed below and indicated by SPE Hutton, as well as agreed to during the

interview, each of the pending claims meets the requirement of 35 U.S.C. § 112, Second Paragraph and 35 U.S.C. § 101 with respect to the foregoing issue.

There was also some discussion generally about the use of “substantially” in the claims. Examiner Tran indicated he would consider whether such use was definite in view of the arguments presented in response to the current action (*i.e.*, this response).

Discussion of 35 U.S.C. § 103 Rejections

A discussion of the rejections under 35 U.S.C. §103(a) as being unpatentable over *Chithambaram*, in view of *Roy* was conducted. In connection with the discussion was a video demonstrating how the Autodesk MapGuide technology disclosed in *Chithambaram* and *Roy* works (in addition, see further discussion below). The video shows a desktop browser display of various MapGuide sites, and clearly demonstrates that the MapGuide implementation is as an embedded application (plug-in) that operates separately from the browser. The video shows the tracking of packets (using a packet-sniffer utility) received from the MapGuide host site, and demonstrates that the data delivered to the MapGuide plug-in does not comprise HTML-based content, but rather comprises proprietary MapGuide data and associated data associated with HTTP Requests and Responses.

During the interview, it was acknowledged by Examiner Tran that he did not appreciate that under the pending claims, the claimed operations were being performed by the device itself, and that none of the claim operations were being performed by a proxy server. In the parent application, the device received scalable content that was translated from its original HTML-based form (*i.e.*, from the original HTML-based Web page content). It is now understood that this is not the case. In view of this consideration, Examiner Tran indicated that a new search would be conducted, and the present §103 rejections were not applicable.

Obviousness-type Double Patenting

A discussion of the obviousness-type double patenting rejection was also discussed. Applicants asserted that the present claims are not obvious over the issued claims of the parent 7,210,099 patent claims. Examiner Tran said he would need to reconsider this rejection in view of his new understanding of the claims and arguments presented in response to the current Office Action. These arguments are presented below.

Discussion of adding Method and Beauregard Claims

During discussions concerning interpretation of the pending claims, Examiner Tran identified that the typical application included method and Beauregard claims in addition to apparatus claims concerning an apparatus configured to perform analogous method operations (as claimed in the method claims) via execution of instructions, *e.g.*, as claimed in the Beauregard claims. Examiner Tran also indicated inclusion of such claims would be helpful in interpreting the apparatus claims. Applicants agreed this would be helpful, and offered to include such claims in present response. Accordingly, Method and Beauregard claims that claim elements substantially analogous to those presented in the pending apparatus (device) claims have been included in the present amendment.

Claim Rejections – 35 U.S.C. § 112, Second Paragraph

Claims 71-73, 75, 79, 81-82, 87, 99, 105, 109-114, 118, 123, 125, 158, 133, 135-136, 143-147, 151, 156, 174-175, and 179 were rejected under 35 U.S.C. 112, second paragraph, as being indefinite for failing to particularly point out and distinctly claim the subject matter which applicant regards as the invention. The §112 rejection of individual claims are addressed below. However, a discussion of the appropriate use of some of the claim terms is first presented.

Use of the terminology “Substantially” in the Claims

The following is quoted from the MPEP¹

2173.05(b) Relative Terminology [R-5] - 2100 Patentability

2173.05(b) Relative Terminology [R-5]

The fact that claim language, including terms of degree, may not be precise, does not automatically render the claim indefinite under 35 U.S.C. 112, second paragraph. *Seattle Box Co., v. Industrial Crating & Packing, Inc.*, 731 F.2d 818, 221 USPQ 568 (Fed. Cir. 1984). Acceptability of the claim language depends on whether one of ordinary skill in the art would understand what is claimed, in light of the specification.

WHEN A TERM OF DEGREE IS PRESENT, DETERMINE WHETHER A STANDARD IS DISCLOSED OR WHETHER ONE OF ORDINARY SKILL IN THE ART WOULD BE APPRISED OF THE SCOPE OF THE CLAIM

When a term of degree is presented in a claim, first a determination is to be made as to whether the specification provides some standard for measuring that degree. If it does not, a determination is made as to whether one of ordinary skill in the art, in view of the prior art and the status of the art, would be nevertheless reasonably apprised of the scope of the invention. Even if the specification uses the same term of degree as in the claim, a rejection may be proper if the scope of the term is not understood when read in light of the specification. While, as a general proposition, broadening modifiers are standard tools in claim drafting in order to avoid reliance on the doctrine of equivalents in infringement actions, when the scope of the claim is unclear a rejection under 35 U.S.C. 112, second paragraph, is proper. See *In re Wiggins*, 488 F. 2d 538, 541, 179 USPQ 421, 423 (CCPA 1973).

When relative terms are used in claims wherein the improvement over the prior art rests entirely upon size or weight of an element in a combination of elements, the adequacy of the disclosure of a standard is of greater criticality.

REFERENCE TO AN OBJECT THAT IS VARIABLE MAY RENDER A CLAIM INDEFINITE

A claim may be rendered indefinite by reference to an object that is variable. For example, the Board has held that a limitation in a claim to a bicycle that recited "said front and rear wheels so spaced as to give a wheelbase that is between 58 percent and 75 percent of the height of the rider that the bicycle was designed for" was indefinite because the relationship of parts was not based on any known standard for sizing a bicycle to a rider, but on a rider of unspecified build. *Ex parte Brummer*, 12 USPQ2d 1653 (Bd. Pat. App. & Inter. 1989). On the other hand, a claim limitation specifying that a certain part of a pediatric wheelchair be "so dimensioned as to be insertable through the space between the doorframe of an automobile and one of the seats" was held to

¹ as published at http://www.uspto.gov/web/offices/pac/mpep/documents/2100_2173_05_b.htm.

be definite. *Orthokinetics, Inc. v. Safety Travel Chairs, Inc.*, 806 F.2d 1565, 1 USPQ2d 1081 (Fed. Cir. 1986). The court stated that the phrase "so dimensioned" is as accurate as the subject matter permits, noting that the patent law does not require that all possible lengths corresponding to the spaces in hundreds of different automobiles be listed in the patent, let alone that they be listed in the claims.

...

D. "Substantially"

The term "substantially" is often used in conjunction with another term to describe a particular characteristic of the claimed invention. It is a broad term. *In re Nehrenberg*, 280 F.2d 161, 126 USPQ 383 (CCPA 1960). The court held that the limitation "to substantially increase the efficiency of the compound as a copper extractant" was definite in view of the general guidelines contained in the specification. *In re Mattison*, 509 F.2d 563, 184 USPQ 484 (CCPA 1975). The court held that the limitation "which produces substantially equal E and H plane illumination patterns" was definite because one of ordinary skill in the art would know what was meant by "substantially equal." *Andrew Corp. v. Gabriel Electronics*, 847 F.2d 819, 6 USPQ2d 2010 (Fed. Cir. 1988).

The use of "substantially" in claims is widely used in modern patent practice. Notably, according to USPTO electronic records, there are 884,854 United States Patents issued since 1976 that include at least one claim including the word "substantially".² It is further noted that the number of patents that have issued since 1976 is 2,801,584.³ Thus, approximately 32% of patents issued since 1976 include at least one claim that recites the word "substantially."

With respect to claim 71, the Examiner states,

Claim 71, recites the limitation "substantially" in Pages 2 and 3, renders the claim indefinite. *Sine* [sic] the current Application merely discloses, "Web content and displays a vector representation of the requested Web content that substantially retains page layout and/or graphics associated with the requested Web content." See current discloses Para 9, which is rendering indefinite, since one of ordinary skill in the art would not be reasonably appraised of the scope of the invention. Also claim 71, recites the limitation "may be" in Page 2,

² Search query of "substantially" in the claims (Field 1: Claims) using the USPTO patent search page at <http://patft.uspto.gov/netahtml/PTO/search-bool.html>.

³ Search query of "A" in the claims (Field 1: Claims) using the USPTO patent search page at <http://patft.uspto.gov/netahtml/PTO/search-bool.html>.

renders the claim indefinite, since one of ordinary skill in the art would not be reasonably appraised [sic] of the scope of the invention.

Applicants have amended claim 71 to remove the use of “may be” to further prosecution of the instant claims; however, the Applicants respectfully assert that the prior terminology was definite. The amendment to this subparagraph is to clarify the claim element and was not made in consideration of any cited art; thus no prosecution history under *Festo* should apply.

Applicants respectfully assert the following terminology in claim 71 would be clearly understood by one of ordinary skill in the art to be definite:

“... translating at least a portion of the HTML-based Web content from its original format into scalable content that supports a scalable resolution-independent display of the Web page ***that substantially retains the original page layout and attributes of the content defined by its original format when rendered.***” (Emphasis added)

Applicants note that this or similar terminology was in the original claims and in the original Summary of the Invention section of the present application and its parent (09/878,097 now US 7,210,099). Additionally, each of the issued claims 1, 7, and 12 of US 7,210,099 contains identical language.

First, as explained below in further detail, one of skill in the art would appreciate the fact that the same Web page (in its original HTML-based format) may not be rendered identically by conventional desktop browsers due to various factors. Moreover, one of skill in the art would understand that due to the combination of both incrementally scalable and mathematically scalable content that could be derived by translating the original HTML-based Web page content, the corresponding scaled page may not result in an exact scaling of the Web page content. For example, under embodiments of the present application the overall layout of a web page may be scaled by a non-integer mathematical factor, such as 40%. However, since Web page text definitions and most browser (more accurately

Operating System) text definitions are defined by incremental font sizes (*e.g.*, a font size defined by an integer, such as 6, 8, 9, 10, 11, 12, 14, *etc.*), a mathematical (*i.e.*, vector) scaling of a Web page layout may produce a slightly different scaling of the Web page text content. For example, suppose text with a font size of 14 is to be scaled. 40% of 14 is 5.6, which is not a standard font size supported by a typical browser or OS. The nearest fixed font size is 6; however, to ensure the text fits the scaled down object area, the font size is rounded down to the nearest font size supported by the browser/OS. In order to accommodate for this, portions of the rendered Web page may need to be adjusted (that is the layout and/or size of those portions may need to be modified) to meet the rendered font capabilities of the browser/OS. This does not imply that it is not possible to have truly (*i.e.*, mathematically) scalable fonts, as the use of such fonts are disclosed in the present application. However, use of fixed size fonts are also disclosed, which may result in scaled Web pages having a slightly different layout (yet substantially similar) than a non-scaled Web page (*i.e.*, the layout of a Web page as defined by its original HTML code).

A similar type of situation exists when using today's Web browsers, such as Microsoft Internet Explorer, Apple Safari, Mozilla Firefox or an Opera browser. There are instances under which the same Web page is rendered to have a (non-scaled) page layout that is slightly different depending on which of these browsers is used. Moreover, the underlying operating system (which is used to generate various fonts) may also have an impact. For instance, Apple Safari on an Apple computer running an Apple OS may not render the same Web page exactly the same when using a PC running a Windows OS.⁴ Various examples of differences in rendering the same Web page are presented below.

⁴ The Apple Safari browser has recently been ported to run on Windows.

A general purpose of the novel resolution-independent Web Page scaling, zooming, and panning techniques disclosed in the present application is to provide user's with a similar browsing experience on substantially any device as they experience using a desktop browser. The location of content in the rendered page is substantially the same as the page appears when rendered by a desktop browser; the difference is the rendered page is scaled to accommodate the different display capabilities of various devices, from handheld devices with small screens (and corresponding smaller resolution capabilities), to desktop PC's and laptops, to very large screen devices, such as large digital advertising billboards and stadium scoreboards. Notably, the techniques may also be employed on desktop PC's and laptops to enabling scaling of Web pages to better fit the screen and enhance readability. Moreover, the full Web page scaling capability works to provide the same browsing experience regardless of the resolution of the screen or display. Thus, a full Web browsing experience is supported on a wide range of devices having different display capabilities and screen sizes. This greatly enhances the usability of Web browsers on these devices.

In summary, applicants respectfully assert the use of the terminology, "... translating at least a portion of the HTML-based Web content from its original format into scalable content that supports a scalable resolution-independent display of the Web page that substantially retains the original page layout and attributes of the content defined by its original format when rendered" is definite and meets the requirements of 35 U.S.C. §112.

The Examiner further discusses the use of "substantially" in claim 71 on page 3 of the previous (August 31, 2007) response. The corresponding claim subparagraph recites,

scaling the scalable content to render the Web page on the display such that the original width of the Web page is rendered *to fit* **substantially across**

the display.

Applicants respectfully assert that the use of substantially is definite in this context, since one of ordinary skill in the art would be reasonably apprised of the scope of this claim element.

For example, Figs. 7A, 8A, and 9A show examples of Web pages rendered to fit substantially across the display of the illustrated Palm IIIc touchscreen display. One of skill in the art would recognize that it may be desirable to provide a border of a few pixels or more around the edges of the rendered Web page for readability purposes and/or aesthetics. Additionally, depending on the scrolling scheme employed, a portion of the browser may be used for scroll bars or the like, such as shown in Figs. 7A, 8A, and 9A. Generally, depending on the underlying operating system (and possibly browser features), the width of the scroll bars may vary, no scroll bars may be displayed, or scroll bars may be overlaid over a portion of the browser's page rendering area, enabling the entire width of the display to be used for browser page rendering. Examples of operating systems and/or browser implementations with different scroll bar widths are shown below:



NYT Web page as rendered on a Mozilla Firefox desktop browser running under the Microsoft Windows XP operating system



Image 1A shows a Web page rendered by a Softview™ browser on a Sony Clié running a version of Palm OS, while Image A2 shows a Web page rendered by a Softview™ browser on a Pocket PC running a version of Windows CE OS. (It is noted that the principle developers of the Softview™ browsers were Gary Rohrabough and Scott Sherman, the inventors of the claimed inventions in the present application, and the Softview™ browsers employ the resolution-independent Web page scaling, zooming, and scrolling techniques disclosed in the application.)

As illustrated in Image 1A, and similarly illustrated in Figs. 7A, 8A, and 9A of the present application (the Palm IIIc also ran on a version of Palm OS), the

Softview™ browser implementation running on a Palm OS employs a vertical and horizontal scroll bar with arrows at the ends that are wider than the bars themselves. Also, the completely filled horizontal scroll bars in each of Image 1A, Figs. 7A, 8A, and 9A, indicates that horizontal scrolling is not applicable, as the Web page view has been rendered to fit across the width of the browser display area.⁵ The scroll bars used by Windows CE are somewhat different – they include separate arrow controls that are the same width as the scroll bars. In a manner similar to the Palm OS examples, the Web page in Image 1B is rendered to fit across the width of the browser display area. Of course, for operating systems/browsers that use overlaid scroll bars, the actual browser display area would be slightly larger. Thus, depending on the type of scroll bar implementation, the portion of the display available to render the Web page (*i.e.*, the browser display area) will vary a small amount. As noted above, border areas may also be desired for readability and/or aesthetics. The net result is that under any of the discussed scroll bar schemes and/or border areas schemes, the rendered Web page will substantially fit the width of the display. Accordingly, Applicants respectfully assert the use of the terminology, “...to fit substantially across the display” is definite within the meaning of 35. U.S.C. §112.

With further respect to claim 71, the examiner states,

“Claim 71, recites the limitation “web content” and “web page” in Pages 2-3. There are insufficient antecedent basis for this limitation in the claim.

Applicants respectfully assert claim 71, prior to the present amendment, did not have an antecedent basis problem for these terms. Moreover, it is clear from the following subparagraph in amended claim 71, which contains the first mention of the

⁵ It is noted that one of ordinary skill in the art would recognize the browser display area is the portion of the rendered display reserved for rendering the Web page content. This typically includes the display area that is not occupied by browser menu items, tool bars (as applicable) and scroll bars (as applicable).

terms “Web content” and “Web page,” that each first mention is proper.

rendering a browser interface via which a user is enabled to request access to a Web page hosted by an Internet Web site, the Web page comprising HTML-based **Web content** having an original format defining an original width and height of the Web page and an original page layout and attributes of content on the Web page;

With respect to claim 73, the Examiner asserts the use of “substantially” renders the claim indefinite. Claim 73 recites,

73. The wireless device of claim 72, wherein the display of the Web page is **re-rendered substantially in real-time** to effect zooming operations. (Emphasis added)

Thus, the use context of “substantially” in claim 73 is re-rendering the display of a Web page substantially in real time. One of skill in the art would recognize the meaning of the terminology “real time” varies depending on the particular use context. For example, for an embedded real-time operating system or implementation, real-time might mean a timeframe in the millisecond or even microsecond range. In this context, the time context is machine time and real-time means instantaneous. In another use context, such as replying to e-mail, real-time is significantly longer. For example, many people refer to responding to e-mail in “real time” – this means the people respond to new e-mails as they come in, as compared with waiting until the end of the day or some other time to respond to e-mails in more of a batch manner. In a real time flight tracking context, the data that is provided may actually reflect a tracking position that is several seconds, or even minutes, old.

One of skill in the art would recognize that in a software user-interface context, which is applicable to the present claims, the use of real-time typically means the user is enabled to continue an operation in a non-disrupted manner,

meaning the user doesn't have to wait a period of time of significance for the operation to be performed. In this context, real-time is perceived by the user's sense of time.

As defined by SearchSMB.com Definitions⁶

real time

DEFINITION- Also see real-time clock and real-time operating system.

Real time is a level of computer responsiveness that a user senses as sufficiently immediate or that enables the computer to keep up with some external process (for example, to present visualizations of the weather as it constantly changes). *Real-time* is an adjective pertaining to computers or processes that operate in real time. Real time describes a human rather than a machine sense of time.

In the days when mainframe batch computers were predominant, an expression for a mainframe that interacted immediately with users working from connected terminals was *online in real time*.

The inclusion of "substantially" in the use of a "substantially in real-time" context is to differentiate the claim from meaning it occurs instantaneously, which would be an erroneous interpretation under the proper use context. Rather, the operation is performed in a non-disrupted manner, as experienced by the user. For the purpose of a defined time period, "substantially in real time" as used herein means the operation is performed in a few seconds or less.

With respect to claim 75, the Examiner asserts the use of "and/or" in the claim renders the claim indefinite. The applicable part of Claim 75 (prior to the present amendment) recited,

parsing markup language code to determine the original page layout of display content within the Web page, wherein the original page layout defines a layout location for a plurality of objects, ***including text objects, graphic layout objects, and/or graphic image objects*** included in the Web page; (Emphasis added)

Applicants respectfully assert that the use of "and/or" in the foregoing claim is

⁶ http://searchsmb.techtarget.com/sDefinition/0,,sid44_gci214344,00.html

proper, and one of ordinary skill in the art would be reasonably apprised of the scope of the foregoing claim element. However, to advance prosecution, applicants have amended claim 75 to now recite,

parsing markup language code to determine the original page layout of display content within the Web page, wherein the original page layout defines a layout location for a plurality of objects, including ***at least one of text objects, graphic layout objects, and graphic image objects*** included in the Web page;

It will be understood that this recitation means the plurality of objects may include any of the following singular or combinations:

1. text objects
2. graphic layout objects
3. graphic image objects
4. a combination of text objects and graphic layout objects
5. a combination of text objects and graphic image objects
6. a combination of graphic layout objects and graphic image objects
7. a combination of text objects, graphic layout objects, and graphic image objects.

Applicants respectfully assert that amended claim 75 is definite.

With respect to claim 79, the Examiner asserts the use of “substantially” renders the claim indefinite. Claim 79 recites,

79. The wireless device of claim 78, wherein execution of the instructions performs further operations comprising enabling the display of the Web page ***to be panned substantially in real-time***. (Emphasis added)

Applicants respectfully assert that one of ordinary skill in the art would be reasonably apprised of the scope of this claim element. In particular, the discussion above concerning the use of similar terminology in claim 73 is likewise applicable to

claim 79.

With respect to claim 81, the Examiner asserts the use of “higher” in the claim renders the claim indefinite. Applicants respectfully disagree. However, to advance prosecution, Applicants have amended claim 81 to now recite:

81. (Currently Amended) The wireless device of claim 71, wherein execution of the instructions performs further operations comprising enabling a user to zoom on view a column of the Web page ~~at a higher resolution than a current resolution~~ via a corresponding user input, wherein in response thereto, the display is re-rendered such that content corresponding to the selected column is displayed substantially across the display.

Similar amendments have been made to each of claims 83, 84, 111, 113, 114, 136, 144, 146, and 147. Applicants respectfully assert that each of these amended claims is definite.

With respect to claim 82, the Examiner asserts the use of “substantially” renders the claim indefinite. Applicants believe the Examiner intended to apply this to both claims 81 and 82 (by dependency), as the word “substantially” is present in claim 81 but not 82. The use of “substantially” in claim 81 concerns displaying the column substantially across the display. For the same reasons as discussed above with respect to displaying the web page substantially across the display, Applicants respectfully assert that this element of claims 81 and 82 (by dependency) is definite.

With respect to claim 87, the Examiner asserts the use of “rapid” renders the claim indefinite. Claim 87 has been amended to remove the use of “rapid.” Applicants respectfully assert that amended Claim 87 is now definite.

With respect to independent claim 99, the Examiner asserts the user of “and/or” and “may be” render the claim indefinite. Claim 99 has been amended to remove this terminology. In particular, the applicable portion of the claim for which “and/or” was previously used now recites,

employing at least one of the scalable content ~~and/or~~ and data derived therefrom to,

render the Web page on the display; and

re-render the Web page in response to associated user inputs to enable the user to zoom in and out a display of the Web page.

It will be understood that employing at least one of the scalable content and data derived therefrom means,

1. Employing the scalable content to perform the render and re-render operations.
2. Employing data derived from the scalable content to perform the render and re-render operations.
3. Employing a combination of the scalable content and data derived from the scalable content to perform the render and re-render operations.

With respect to claim 105, the examiner asserts the use of “and/or” renders the claim indefinite. Claim 105 has been amended to remove the “and/or” terminology and uses similar language as discussed above with respect to Claim 99. Applicants respectfully assert that claim 105 is definite.

With respect to claims 109-114, the examiner asserts the use of “and/or”, “higher”, and “substantially” renders the claims indefinite.

Applicants respectfully assert that the use of “enabling the Web page to be panned substantially in real-time” in claim 109 is definite for reasons similar to those argued above with respect to claim 73.

Applicants have amended claim 110 to remove the “and/or” terminology in a manner similar to claim 99 discussed above.

Applicants have amended claims 110-114 (claim 112 indirectly by dependency) to remove the terminology “higher” in a manner similar to that discussed above with respect to claim 81.

With respect to claim 118, the Examiner asserts the use of “its” renders the claims indefinite. Claim 118 has been amended to remove the term “its”; the applicable part of claim 118 now recites,

creating a reference that links the object to its corresponding the vector that is generated.

Applicants respectfully assert claim 118 is now definite.

With respect to claim 123, the Examiner asserts the use of “substantially” renders the claim indefinite. Applicants respectfully assert the use of “to display at least one of the width and height of the Web page substantially across a display area of the display” is definite for reasons similar to those argued above with respect to the use of similar language in claim 71.

With respect to claims 125 and 158, the Examiner asserts the use of “HTML-based content comprises XML-based content” renders the claims indefinite. Applicants respectfully traverse this rejection.

It is well-known to those of ordinary skill in the art that Web pages are defined, at least in part, by corresponding HTML code contained in one or more HTML documents. In general, such HTML code may be stored in one or more files (*i.e.*, static content), or all or a portion of the HTML code may be dynamically generated in response to user access requests (*e.g.*, through scripts and other techniques). While the Web page content includes HTML code, it is not limited to HTML code. Thus, the use of the terminology “HTML-based content” in the present claims. As discussed throughout the specification and shown in the drawings of the present application, the Web page content may include XML code, cascaded style sheet (CSS) data, and scripting language code, such as JavaScript. Those of ordinary skill in the art would recognize that Web page content may also contain other types of content. Moreover, one of ordinary skill in the art would recognize that XML code comprises XML-based content.

Notably, extensive discussion of HTML documents and their use is disclosed in