

UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE PATENT TRIAL AND APPEAL BOARD

GOOGLE LLC,
Petitioner

v.

SOFTVIEW LLC,
Patent Owner

Patent No. 7,844,889

**DECLARATION OF DR. BENJAMIN GOLDBERG IN SUPPORT OF
PETITION FOR *INTER PARTES* REVIEW
OF U.S. PATENT NO. 7,844,889**

TABLE OF CONTENTS

I.	INTRODUCTION	6
II.	BACKGROUND AND QUALIFICATIONS	6
III.	MATERIALS REVIEWED	8
IV.	PERSON OF ORDINARY SKILL IN THE ART AND THE TIME OF THE ALLEGED INVENTION	11
V.	OVERVIEW OF THE '889 PATENT	12
VI.	UNPATENTABLE CLAIMS IN RELATED PATENTS	15
VII.	OVERVIEW OF THE PRIOR ART	15
	A. Zaurus (EX1004)	15
	B. Pad (EX1005 and EX1007)	16
	C. SVG (EX1008)	17
VIII.	CLAIM CONSTRUCTION	17
IX.	DETAILED EXPLANATION OF GROUNDS	21
	A. Ground 1: Zaurus, Pad, and SVG Disclose and/or Suggest Claims 1-115	21
	1. Motivation to Combine	21
	2. Reasonable Expectation of Success	23
	3. Claims 1, 31, 42, 77, 104, and 108	25
	4. Claims 2, 43, and 78	45
	5. Claims 3 and 71	47
	6. Claims 4, 45, and 79	49
	7. Claims 5, 46, and 80	53
	8. Claims 6, 47, and 81	58
	9. Claims 7, 48, and 82	60
	10. Claim 8	63
	11. Claim 9	67
	12. Claim 10	71
	13. Claims 11, 49, and 83	74
	14. Claims 12, 50, and 84	77

Declaration of Dr. Benjamin Goldberg
U.S. Patent No. 7,844,889

15.	Claims 13, 51, 86, and 114	80
16.	Claims 14, 52, 87, 114, and 115	81
17.	Claims 15, 53, and 88	83
18.	Claims 16, 54, and 89	85
19.	Claims 17, 55, and 90	86
20.	Claims 18, 56, and 91	87
21.	Claims 19, 57, and 92	88
22.	Claims 20, 58, and 93	90
23.	Claims 21, 59, and 94	91
24.	Claims 22, 60, and 95	92
25.	Claims 23, 61, and 96	94
26.	Claims 24, 62, and 97	98
27.	Claims 25, 63, and 98	99
28.	Claim 26, 64, and 99	102
29.	Claim 27, 65, and 100	105
30.	Claims 28, 66, and 101	107
31.	Claims 29, 67, and 102	109
32.	Claims 30 and 68.....	111
33.	Claims 32, 70, and 106	111
34.	Claim 33: “The method of claim 1, further comprising maintaining at least one instance of the page layout information in a manner that is independent of zoom levels used to view the web page.”	112
35.	Claim 34: “The method of claim 1, wherein the scalable page layout information comprises scalable vector-based page layout information.”	113
36.	Claim 35.....	114
37.	Claim 36.....	119
38.	Claim 37: “The method of claim 1, wherein the markup language-based Web content includes cascading style	

	sheet content defining layout and presentation attributes for the Web page.”	121
39.	Claims 38 and 73.....	123
40.	Claims 39, and 109.....	124
41.	Claims 40, 75, and 107	127
42.	Claims 41 and 76.....	129
43.	Claims 44 and 74.....	130
44.	Claims 69 and 105.....	131
45.	Claim 72: “The apparatus of claim 42, wherein the storage means comprises a flash memory.”	132
46.	Claim 85: “The machine-readable medium of claim 77, wherein execution of the instructions performs further operations comprising scaling the scalable page layout information and/or data derived therefrom to regenerate views of the Web content in response to associated user inputs to enable the user to iteratively zoom in and out a display of the Web page.”	132
47.	Claim 103: “The machine-readable medium of claim 77, wherein at least a portion of the instructions comprise Java-based instructions configured to be executed on a Java virtual machine.”	133
48.	Claim 110	134
49.	Claim 111	136
50.	Claim 112: “The method of claim 111, further comprising: employing a scalable font to scale text content for a text object to fit within a scaled bounding box.”	137
51.	Claim 113	138
X.	CONCLUSION.....	140
	Appendix A - Comparison to Unpatentable Claims	141
	Appendix B - Unpatentable ’353 Patent Claims.....	266
	Appendix C - Unpatentable ’926 Patent Claims in IPR2013-00004.....	274
	Appendix D - Unpatentable ’926 Patent Claims in 90,009,995	279

Declaration of Dr. Benjamin Goldberg

U.S. Patent No. 7,844,889

Appendix E - Unpatentable '926 Patent Claims in 95,000,635.....292

I. INTRODUCTION

1. I have been retained as an independent expert consultant in this proceeding before the United States Patent and Trademark Office (“PTO”) regarding U.S. Patent No. 7,844,889 (“the ’889 patent”) (EX1001).¹ I have been asked to consider whether prior art references disclose or suggest the features recited in claims 1-115 (“the challenged claims”) of the ’889 patent. My opinions are set forth below.

II. BACKGROUND AND QUALIFICATIONS

2. I am a tenured Associate Professor in the Department of Computer Science of the Courant Institute of Mathematical Sciences, New York University (“NYU”), in New York, New York. I have held this position since September 1994. From 1987 to 1994, I was an Assistant Professor in the Department of Computer Science at NYU. From September 2014 through August 2022, I was the Director of Graduate Studies for the MS programs in the Department of Computer Science, having previously served in that role from September 2009 through August 2012. I served as the Director of Undergraduate Studies for the Department of Computer Science from September 1995 through August 1998 and from September 2003

¹ In this Declaration, I refer to exhibits that I understand are to be attached to the petition for *Inter Partes* Review of the ’889 patent.

through August 2006. In addition, I have held a one-year visiting professorship at the Institut National de Recherche en Informatique et en Automatique (INRIA), a national laboratory in France, and was twice appointed to a month-long position as an invited professor at the Ecole Normale Supérieure, a university in Paris.

3. I received my Doctoral degree in Computer Science from Yale University, New Haven, Connecticut in 1988, having previously received Master of Science and Master of Philosophy degrees in Computer Science from Yale in 1984. My undergraduate degree from Williams College in 1982 was a Bachelor of Arts degree with highest honors in Mathematical Sciences.

4. I have taught courses and lectured at the undergraduate and graduate level in programming languages, hardware design, operating systems, user interfaces, web services, computer networks, and other areas related to the technology at issue in this matter.

5. Additional information concerning the computer science courses that I have taught, my professional publications and presentations in the field of computer science, and cases in which I have testified as an expert at trial or deposition in the past five years are set forth in my current Curriculum Vitae, a copy of which is attached as EX1006.

6. I am being compensated at a rate of \$525 per hour for my work in this case. I will be compensated at the same rate for any deposition I give in this matter. My compensation is in no way conditioned on the outcome of this matter.

III. MATERIALS REVIEWED

7. The opinions contained in this Declaration are based on the documents I reviewed, my professional judgment, as well as my education, experience, and knowledge regarding systems and processes in the field of graphics interface technology.

8. In forming my opinions expressed in this Declaration, I reviewed the following materials:

Declaration of Dr. Benjamin Goldberg
U.S. Patent No. 7,844,889

EX1001	U.S. Patent No. 7,844,889 to Rohrabauh et al. (“the ’889 patent”)
EX1003	Prosecution History of U.S. Patent App. No. 11/735,482
EX1004	Power Zaurus Specifications: User Manual - Mobile Business Tool - Model MI-106 / MI-106M / MI-110M (Nov. 1997) (“Zaurus”)
EX1005	Pad++ Reference Manual Version 0.2.7 (Oct. 2, 1996) (“Pad_1”)
EX1007	Pad++ Tour (Apr. 1997) (“Pad_2”)
EX1008	Ferraiolo, Jon, <i>Scalable Vector Graphics Requirements: W3C Working Draft</i> (Oct. 29, 1998) (“SVG”)
EX1011	U.S. Patent No. 7,831,926 to Rohrabauh et al. (“the ’926 patent”)
EX1015	U.S. Patent No. 7,461,353 to Rohrabauh et al. (“the ’353 patent”)
EX1035	World Wide Web Consortium (W3C) Working Draft, <i>Level 1 Document Object Model Specification</i> , W3C (July 20, 1998)
EX1036	Lisa Rein, <i>Structuring Graphical Data</i> , XML.com (June 22, 1998)
EX1037	Definition of Cascading Style Sheets (CSS) from W3C Proposed Recommendation, <i>Cascading Style Sheets, Level 2</i> , W3C (Mar. 24, 1998)
EX1040	W3C, <i>11 Tables</i> , W3C (Wayback Machine Capture on Aug. 16, 2000)
EX1041	W3C, <i>16 Frames</i> , W3C (Wayback Machine Capture on Aug. 16, 2000)
EX1042	W3C, <i>Paragraphs</i> , W3C (Wayback Machine Capture on June 6, 1997)
EX1043	W3C, <i>HTML 3.2 Reference Specification publication history</i> , W3C (Jan. 14, 1997)
EX1046	W3C Recommendation, <i>Extensible Markup Language (XML) 1.0</i> , W3C (dated Feb. 10, 1998; Wayback Machine Capture dated Feb. 13, 1998)

EX1050	<i>Marc Andreessen</i> , BRITANNICA, https://www.britannica.com/biography/Marc-Andreessen (last visited Jan. 27, 2026)
EX1051	<i>History of the Mozilla Project</i> , MOZILLA, https://www.mozilla.org/en-US/about/history/ (last visited Jan. 27, 2026)
EX1052	<i>OpenGL Programming Guide</i> , Second Edition, Addison-Wesley, 1997 (available at https://laplace.physics.ubc.ca/People/ajpenner/GPGPU/redbook.pdf)
EX1053	Adobe, <i>PostScript Language Reference</i> , 3d ed. (Feb. 1999) (available at https://www.adobe.com/jp/print/postscript/pdfs/PLRM.pdf)
EX1054	OpenGL 1.1 Specification (Version 1.1 - 4 March 1997) (available at https://registry.khronos.org/OpenGL/specs/gl/glspec11.pdf)
EX1055	Sun Microsystems, Inc., <i>PersonalJava Technology 2</i> , (1999) (available at https://www.oracle.com/technetwork/java/pjava-ds-150025.pdf)
EX1056	A. Guttman & Michael Stonebraker, <i>R-Trees: A Dynamic Index Structure for Spatial Searching</i> (EECS Dep't, Univ. of Cal., Berkeley, Tech. Rep. No. UCB/ERL M83/64, 1983) (available at https://www2.eecs.berkeley.edu/Pubs/TechRpts/1983/ERL-m-83-64.pdf)

9. I also considered any other documents and materials I refer to in this Declaration.

10. My opinions contained in this declaration are based on the documents I reviewed and my knowledge and professional judgment. My opinions have also been guided by my appreciation of how a person of ordinary skill in the art would

have understood the state of the art, the prior art, and the claims and the specification of the '889 patent at the time of the alleged invention, which I discuss below.

11. I have been asked to consider that the time of the alleged invention of the '889 patent was around 2000 (including and up to June 12, 2000), which corresponds to the filing date of the earliest application associated with the '889 patent. (EX1001, Cover.)

12. Based on my experience and expertise, it is my opinion that certain references disclose and/or suggest all the features recited in challenged claims of the '889 patent, as I discuss in detail below.

IV. PERSON OF ORDINARY SKILL IN THE ART AND THE TIME OF THE ALLEGED INVENTION

13. I am familiar with the level of ordinary skill in the art regarding the '889 patent as of around June 12, 2000. Considering the '889 patent, the technology, the educational level and experience of workers in the field relating to the patent, and problems and solutions in that field, and drawing on my own experience, I believe a person of ordinary skill in the art at the time of the alleged invention of the '889 patent would have had an undergraduate degree in electrical engineering, computer engineering, computer science or a related field along with two years of work experience in the field of web development, graphics rendering, and/or mobile applications. More education can substitute for practical experience and vice versa.

14. My opinions in this Declaration regarding the '889 patent and the prior art (including the state of the art) are from the perspective of one of ordinary skill in the art as I defined above, during the relevant timeframe (e.g., the time of the alleged invention), which I discussed above as being around June 12, 2000.

V. OVERVIEW OF THE '889 PATENT

15. The '889 patent is directed to improving the readability of web pages on small screen devices by converting HTML and Cascading Style Sheets (CSS) into a resolution-independent, scalable representation that can be zoomed and panned while preserving the page's original layout and functionality. (EX1001, Abs., 1:53-2:49.) The '889 patent relies on conventional browsers such as Mozilla for pre-rendering such as parsing, grouping objects, making bounding boxes, and defining page layout. (EX1001, FIG. 5 (150-154), 17:21-31; *see also id.* at 15:38-17:20.)

16. The '889 patent utilizes the SVF (Simple Vector Format) considered by the W3C (World Wide Web Consortium) to translate content into a scalable vector representation. (EX1001, 4:26-47, 5:52-6:15, 6:34-49, 8:39-62, 12:6-12.) Relative to the prior art, the '889 patent asserts to disclose unique steps for using layout data generated during pre-rendering to generate a scalable vector representation of the original page content. (EX1001, 17:32-35.) These steps include defining a primary page datum and computing vectors from that datum to each object's datum, storing

references that link object content and attributes to those vectors and boxes. (EX1001, 17:32-19:3, FIG. 4C-4D, 5, 6.) While rendering, the device builds a display list, applies user-selectable scale and offset, computes a display-limit bounding box, fetches and scales the content that intersects that region (e.g., fonts, images, borders), and renders the content in a way that enables fast zooming and panning. (EX1001, 19:4-20:34, FIG. 4E-4G, 5, 6.)

17. The '889 patent acknowledges that building the display list of vectors and rapid zooming of vector-based objects was well known in the art. (EX1001, 19:4-21.) The '889 patent asserts to enable “context zooms” (e.g., tap to zoom a column, image, or paragraph) without reducing quality, and the use of display lists to improve rendering speed. (EX1001, Abs., 20:35-21:2.) The '889 patent illustrates these interface behaviors such as tap-to-zoom for columns, images, and paragraphs on a mobile device. (EX1001, FIGs. 7-9.)

18. The independent claim of the '889 patent is directed to mobile handheld device browsing by (i) retrieving Web content, (ii) processing the Web page content to generate page layout information, (iii) employing the page layout information to generate scalable page layout information, and (iv) employing the scalable page layout information to generate views of the Web page to enable browsing while preserving the original layout, functionality, and design of the Web page. (EX1001, 22:6-22:36.)

19. SoftView filed the patent application with 118 claims. (EX1003, 41-64.) The one Office action allowed some claims and rejected other claims under (i) indefiniteness, and (ii) patent eligibility. (EX1003, 202-209.) In response, SoftView (i) made significant amendments, (ii) canceled some claims, (iii) presented some new claims, and (iv) and presented arguments. (EX1003, 310-388.) During prosecution, SoftView explained that “‘in real time’ as used herein means the operation is performed in a few seconds or less...while the terminology ‘in real-time’ may encompass ‘a fraction of a second,’ the scope of real-time shall not be limited to a fraction of a second.” (EX1003, 344-346.) SoftView also acknowledged that “one of ordinary skill in the browser art would not expect Web pages rendered using an implementation in accordance with the teachings disclosed in the present application to render pages as exact scaled replicas of the same page rendered by a conventional desktop browser” and that “exact (i.e., perfect magnification) ... results are not required by the scope of the [claims].” (EX1003, 359.) SoftView also explained that term “approximately proportionally” covers any approach where increasing the zoom results in using more pixels to render the text. (EX1003, 383.) SoftView subsequently authorized the Examiner to enter an Examiner’s amendment, after which the claims were allowed. (EX1003, 409-441.) SoftView submitted comments disagreeing with the Examiner’s characterization of the claims in the Notice of Allowance. (EX1003, 489-496.) SoftView explained that “if the recited

claim language includes the term ‘scalable’ or ‘scaled’ without including the term ‘vector-based,’ then the scope of the claim does not require the corresponding element to be vector-based.” (EX1003, 493.) On February 1, 2011, the USPTO entered a certificate of correction for typos in certain issued claims. (EX1001, 42.)

VI. UNPATENTABLE CLAIMS IN RELATED PATENTS

20. I understand that claims of the related ’926 and ’353 patents have previously been found unpatentable and canceled. (EX1011, 42-48; EX1015, 52.) I have been asked to consider whether the challenged claims would be disclosed and/or suggested by the previously canceled claims. As such, I have provided my analysis in Appendices A-E.

VII. OVERVIEW OF THE PRIOR ART

A. Zaurus (EX1004)

21. Zaurus discloses a handheld device with an HTML browser that supports touch inputs and switching between reduced and magnified displays, among other features. (EX1004, 10-16, 20-21, 40, 46-47.) Zaurus is designed to offer the user a portable and compact browsing experience. (EX1004, 10.) The Internet browser includes HTML 3.2 with client-side clickable maps and offline saving of pages. (EX1004, 40.) Zaurus describes user selectable zooming while browsing. (EX1004, 14, 15 (“Touch the screen with a pen. Let's try various operations by touching the screen with a pen.”), 16, 46 (“Switch between reduced

display and magnified display”), 47 (“To switch the screen from a reduced view to a magnified view, touch Reduce ▼ followed by Magnify. To switch back from a magnified view to a reduced view, touch Magnify ▼ followed by Reduce.”).) Zaurus lists a bundled touch pen and PC linking software (“Zaurus Power Connection”), further supporting exchanging data with a desktop environment. (EX1004, 20.)

B. Pad (EX1005 and EX1007)

22. Pad++ (“Pad”) discloses an HTML browser with zooming capabilities that functions as “a zooming version of Mosaic and Netscape.” (EX1007, 86; *see also id.*, 82 (“only loads the contents of the files that you zoom in on”), 86-96, 113 (clicking a link “performs an animated pan/zoom to the location that the author specified”), 116 (“When you pan and zoom the Pad++ surface, sticky object remain stationary”), 141 (“Pad++ uses an R-Tree internally to quickly determine which objects are visible in a given view”).) Pad discloses mechanisms to animate and control view changes for small screens and personal digital assistants (PDAs). (EX1007, 139-141.) Pad provides a zoomable surface and rendering primitives for line, polygon, text, and image drawing, with an efficient, byte-compiled scripting interface that can “render directly on the Pad++ surface.” (EX1005, 79; EX1007, 73.) Pad further discloses file I/O, read, write, and rendering commands for changing views, drawing lines, polygons, and text. (EX1005, 5, 82.)

C. SVG (EX1008)

23. SVG discloses standards-based vector graphics that can be implemented in browsers designed for “small mobile devices.” (EX1008, 1 (“graphics in Web documents will be smaller, faster, more interactive, and be displayable on a wider range of device resolutions from small mobile devices through office computer monitors to high resolution printers”).) The SVG standard “should support zoom and pan on graphics, with true scalability.” (EX1008, 14.) SVG was designed to be widely implemented “on a wider range of device resolutions from small mobile devices through office computer monitors to high resolution printers.” (EX1008, 1.) SVG also supports transformations such as nested two by three matrices for scaling, translation, and rotation to provide uniform scalability. (EX1008, 10-12.)

VIII. CLAIM CONSTRUCTION

24. I understand that when considering the meaning of patent claims, one must consider the language of the claims, the specification, and the prosecution history of the patent. I also understand that claim terms are typically given their ordinary and customary meanings as they would have been understood by a person of ordinary skill in the art at the time of the invention for the patent. For the '889 patent, as I explained above in Section IV, the relevant time that I have been asked to consider is around June 12, 2000. For purposes of my opinions in this declaration,

I have been asked to consider the challenged claims as understood by a person of ordinary skill in the art at the time of the alleged invention, and I have applied such understandings in my analysis of the '889 patent and the prior art.

25. With respect to the term “storage means,” I have been asked to assume that this term refers to the function of “storing a plurality of instructions,” and that the corresponding structure includes a floppy disk, optical disk, CD-ROM, magneto-optical disk, ROM, RAM, EPROM, EEPROM, magnetic or optical card, or flash memory. This understanding is consistent with the specification. (*See* EX1001, 4:5-24.)

26. I understand that the following terms appearing in the '889 patent were previously construed by a district court and/or submitted as proposed constructions in district court.

Term	Construction
“A machine-readable medium”	“A medium for storing 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.”
“format”	“a particular way that information is encoded”
“original” / “original page layout, functionality, and design of content on the Web	The term “original” means “as designed for a desktop computer.”

page” / “original format” / “original page layout of content on the Web page” / “original page layout”	
“translating”	“converting”
“scalable page layout information”	“page layout information capable of being rendered at multiple zoom levels”
“scalable” / “scaling” / “scaled” / “scale” ²	“capable of being rendered at multiple zoom levels” / “rendering at multiple zoom levels” / “rendered at multiple zoom levels” / “render at multiple zoom levels”
“preserving” / “preservation” / “preserving the original page layout, functionality, and design of the content on the Web page defined by the [markup language-based Web content / HTML-based Web content as interpreted by the rendering engine]”	The term “preserve” and its variants have their plain and ordinary meaning.
“enabling a user to zoom on [a column / a paragraph / an image] [of the Web page / of the Web content]”	“enabling a user to use scalable content to zoom on [a column / a paragraph / an image] [of the Web page / of the Web content]”

² I understand the term “scale” is only intended to encompass the term when used as a verb.

“processing ... the markup language-based Web content ... to generate page layout information” / “processing the HTML-based Web content ... to generate page layout information”	“converting Web content from [HTML / markup language] to [scalable content / page layout information / a scalable vector representation of the Web page]” / “converting at least a portion of the Web content from HTML to scalable content” / “converting Web content associated with the hyperlink from [HTML / HTML and CSS] to an additional scalable vector representation”
“tapping”	Plain and ordinary meaning
“fit across”	Plain and ordinary meaning
“object(s)”	“a grouping of webpage content, such as the paragraph of text or a graphic image”
“object datum”	“reference point for an object used in a predictable manner”
“vector”	“for each object, a directional data structure that stores X & Y values from a point for the page or frame to the object”
“scalable vector-based page layout information”	“page layout information that (1) is capable of being rendered at multiple zoom levels and (2) includes one or more vectors”
“bounding box”	“an outlined shape within which an object’s content (<i>e.g.</i> , text or graphic image) will appear”

27. I understand that the Board construed a claim term during the IPRs of the '353 and '926 patents, as shown below.

Term	Construction
preserve[s] an original page layout, functionality and design	maintains the features of the web page's capabilities and appearances in a manner consistent with the translated portion of HTML code defining those capabilities and appearances

28. Where applicable, I have also applied those constructions in my analysis.

IX. DETAILED EXPLANATION OF GROUNDS

29. In my opinion, the prior art I discuss below discloses and/or suggests the limitations of claims 1-115.

A. Ground 1: Zaurus, Pad, and SVG Disclose and/or Suggest Claims 1-115

1. Motivation to Combine

30. A person of ordinary skill in the art would have been motivated to combine the references to improve small-screen navigation using known techniques. Zaurus's HTML support, Pad's HTML support, and SVG were basic software designs directed at rendering Web pages that a person of ordinary skill in the art would have been motivated to combine. (*See e.g.*, EX1008; EX1035; EX1037; EX1043; EX1046.)

31. A person of ordinary skill in the art would have been motivated to incorporate the browsing techniques taught by Pad into the Zaurus browser to improve the browsing experience on the Zaurus device. The browser on the Zaurus device already supports viewing content in either a “magnified view” or a “reduced view.” (EX1004, 46-47.) A person of ordinary skill in the art would have been motivated to incorporate Pad’s zooming techniques into the Zaurus browser so that the Zaurus browser could display content at more than two views, which “seems especially attractive for systems which have small screens, such as handheld computers (i.e. PDA’s).” (EX1007, 141.) Pad’s FAQ identifies motivation to apply Pad to small-screen systems such as handheld computers and explains performance measures (level of detail, adaptive render scheduling) for fast zoom animations. (EX1007, 136-142.)

32. Pad discloses that its “Pad++ web browser” functions as “a zooming version of Mosaic and Netscape.” (EX1007, 86.) A person of ordinary skill in the art would have understood that Mosaic and Netscape were the dominant desktop web browsers of that era. (EX1050, 1-2 (noting Netscape “became the most popular browser...taking over 75 percent of the market share by mid-1996” and was co-founded by Marc Andreessen, who “played a key role in creating the Web browser

Mosaic”).)³ By describing the mobile-compatible Pad browser as a “version of” these desktop browsers (EX1007, 86), a person of ordinary skill in the art would have recognized Pad as a predictable solution for replicating desktop browsing on a handheld device.

33. SVG discloses techniques for zooming and panning content with true scalability on a variety of devices including small mobile devices. (EX1008, 1, 14.) A person of ordinary skill in the art would have been motivated to incorporate SVG’s techniques into the Zaurus browser to ensure that the zoomed content is readable and that its design and functionality is preserved. With Pad’s zooming techniques with SVG’s standards-based vector graphics, the Zaurus browser would display a scalable and interactive interface on the touchscreen display of the Zaurus handheld device.

2. Reasonable Expectation of Success

34. A person of ordinary skill in the art would also have had a reasonable expectation of success in combining these known elements. Each reference

³ Notably, the ’889 patent explains that the invention uses “core functionality provided by the Mozilla rendering engine source code.” (EX1001, 17:21-31.) A person of ordinary skill in the art would have known that Netscape is the predecessor to Mozilla. (EX1051, 1 (“The Mozilla project was created in 1998 with the release of the Netscape browser suite source code.”).)

demonstrates that the underlying technology (handheld hardware, touch screen inputs, and HTML rendering) was well known. Zaurus's HTML browser, Pad's HTML browser, and SVG were also based on basic software designs known to a person of ordinary skill in the art to be interoperable with each other. (*See e.g.*, EX1008; EX1035; EX1037; EX1043; EX1046.) A person of ordinary skill in the art would have had no technical obstacles and every reason to expect the combination of interoperable concepts to function as intended.

35. A person of ordinary skill in the art would have a reasonable expectation of success in combining Pad's zoom and layout mechanisms with SVG's vector model and deploying them on the handheld Zaurus device. Zaurus supported a stylus-operated Internet browser with basic zooming ("magnified view" vs. "reduced view"), demonstrating that the hardware could handle variable display states and that incorporating a zoomable UI would be a predictable adaptation to a known device and input method. (EX1004, 15-16, 20-21, 46-47.) Likewise, SVG was designed to be implemented on "small mobile devices," ensuring that the vector graphics format was compatible with Zaurus. (EX1008, 1.)

36. Furthermore, Pad implemented smooth animated zoom of objects via bounding boxes (EX1005, 23, 30), which provided a framework for the zoomable interactions taught by SVG to attain scalable zoom and pan (EX1008, 14-15.) Pad disclosed items and SVG disclosed object access and per-object events;

incorporating SVG's scriptable objects would have been a predictable change aligned with the standard. (EX1005, 22-23; EX1008, 15.) Pad's grids and visibility options correspond to SVG's grouping and layering, which would have allowed a person of ordinary skill in the art to make a routine integration of known mechanisms to organize controls and content at various zoom levels. (EX1005, 54-56, 71; EX1008, 13.) Thus, a person of ordinary skill in the art would have expected the combination to function as intended: a handheld browser (Zaurus) utilizing vector scaling (SVG) to enable continuous zooming (Pad) of Web content.

3. Claims 1, 31, 42, 77, 104, and 108

**i) [1.pre] "A method comprising:"
[108.pre] "A method comprising:"**

37. Zaurus performs handheld web-browsing operations (e.g., connecting to the Internet and following links by touch), and Pad and SVG supply the operations executed by the claimed "method." (EX1004, 10, 40; EX1005, 60.)

ii) [42.pre] "An apparatus comprising a hand-held mobile device, including:"

38. Zaurus is described as a compact, highly portable device for "browsing" the Internet ("a hand-held mobile Internet device"). (EX1004, 10.)

iii) [42.a] "a processor,"

39. Zaurus's specifications disclose a "CPU 32-bit RISC" ("processor"). (EX1004, 18.)

- iv) **[42.b] “a communications device operatively coupled to the processor, to enable the apparatus to be wirelessly linked to a network to access Web content;”**

40. Zaurus teaches Internet data communication using wireless telephony adapters and states that to connect to the Internet (or NIFTY Serve) the user will need “a PHS adapter or a digital cellular phone adapter,” and describing data communication carried out by using a digital cellular phone with the adapter (“communications device operatively coupled to the processor, to enable the apparatus to be wirelessly linked to a network to access Web content”). (EX1004, 7, 10, 39.)

- v) **[42.c] “a display; and”**

41. Zaurus’s specifications disclose a built-in “Display 320 x 240 dots ... liquid crystal” (“display”). (EX1004, 18.)

- vi) **[42.d] “storage means, operatively coupled to the processor, in which a plurality of instructions, including instructions corresponding to a browser having a rendering engine and instructions corresponding to an operating system, are stored that when executed by the processor enable the apparatus to perform operations including,”**

42. Zaurus discloses main-body memory including flash memory (“storage means”) suitable for storing executable “instructions,” including the built-in Internet “browser.” (EX1004, 18, 40.) Zaurus discloses a general-purpose PDA software environment (multiple built-in applications and “MORE software”) with an Internet

browser (“rendering engine”), which a person of ordinary skill in the art would have understood to be provided by an “operating system” stored in the device’s memory and executed by the “processor.” (EX1004, 15, 18.) In the combined device, Pad also provides a “browser” with a “rendering engine” by teaching that when “html data is fetched, it is parsed and the HTML item is created which contains a method for rendering the page,” including URL fetching and rendering behavior. (EX1005, 60.)

- vii) [77.pre] **“A tangible non-transitory machine-readable medium having stored thereon a plurality of instructions comprising a browser including a rendering engine configured to be executed by a wireless hand-held device having a display and an operating system and configured to access Web content via a network, wherein execution of the instructions in conjunction with execution of the operating system enables the device to perform operations comprising:”**

43. Zaurus’s flash memory is a “tangible non-transitory machine-readable medium” suitable for storing the “instructions” for a “browser” and operable by an “operating system.” (EX1004, 18.) Zaurus’s Internet browser accesses Web content over Internet connections supported by PHS/digital cellular adapters (“access Web content via a network”) and displays that content on its integrated “display.” (EX1004, 10, 18, 40.) Pad supplies the “browser including a rendering engine” instruction set by teaching HTML fetching/parsing and a “method for rendering the

page,” executable in conjunction with the device’s software “operating system.”

(EX1005, 60.)

viii) **[1.a] “in response to a request to access a Web page made via a wireless hand-held device including a display, retrieving markup language-based Web content associated with the Web page having an original format defining an original page layout of content on the Web page, the content including text content, a plurality of hyperlinks, and a plurality of images;”**

[42.d.1] “in response to a request to access a Web page made via a user of the apparatus, retrieving markup language-based Web content associated with the Web page having an original format defining an original page layout, functionality and design of content on the Web page, the content including text content, a plurality of hyperlinks, and a plurality of images;”

[77.a] “in response to a request to access a Web page made via a user of the device, retrieving markup language-based Web content associated with the Web page having an original format defining an original page layout, functionality and design of content on the Web page, the content including text content, a plurality of hyperlinks, and a plurality of images;”

[108.a] “in response to a request to access a Web page made via a mobile hand-held device, retrieving HTML-based Web content associated with the Web page having an original format defining an original page layout, functionality, and design of content on the Web page, the content including text content, a plurality of hyperlinks, and a plurality of images;”

44. In the Zaurus-Pad-SVG combination, in response to a Web page request, the hand-held device retrieves Web content for that page in an original format that defines the page’s original layout (and where recited, its “functionality

and design”), and the retrieved content includes text, hyperlinks, and images. The retrieved HTML is obtained in its desktop-designed encoding and structure, including references to images and links that define the desktop page layout, functionality, and design (“in response to a request to access a Web page”; “retrieving ... Web content”; “original format”; “original page layout”; “functionality and design”; “text content”; “a plurality of hyperlinks”; “a plurality of images”). (EX1004, 40-42, 46; EX1005, 60, 63, 75; EX1007, 91; EX1043.)

45. Zaurus discloses the request-triggered retrieval: a user can connect to the Internet and open home pages / open another page by specifying a URL (“in response to a request to access a Web page”). (EX1004, 39, 46.) Zaurus also describes link-driven navigation, where touching underlined characters causes the linked page to be displayed, confirming retrieval responsive to user selection of hyperlinks. (EX1004, 10.) Zaurus further teaches that its browser “incorporates HTML 3.2 specifications,” and a person of ordinary skill in the art would have understood that HTML is a markup language (“retrieving HTML-based Web content”; “retrieving markup language-based Web content”). (EX1004, 40.) Zaurus also shows that the retrieved content’s HTML “specifications” define layout (e.g., “Pages composed of frames”) (“original format”; “defining an original page layout”). (EX1004, 40; EX1043.)

46. Zaurus also discloses the required content components: it discloses “text data,” link behavior via “underlined characters,” and images (e.g., GIF/JPEG), and it includes a setting indicating “image data is displayed” versus “only text data is displayed,” confirming that retrieved pages include text and images and link structures (“text content”; “a plurality of hyperlinks”; “a plurality of images”). (EX1004, 40, 42.)

47. Pad discloses the same pipeline at the browser/rendering engine level: “HTML items” can “automatically fetch a file from a URL” and also fetch in-line images, and when the html data is fetched, “it is parsed and the HTML item is created,” confirming retrieval and interpretation of the original-format HTML to generate the page representation/layout (“retrieving ... Web content”; “original format”; “defining an original page layout”). (EX1005, 60.) Pad further teaches that the HTML item specifies the page width and the “page will be re-laid out according to the new width,” and that the HTML browser can lay out sub-documents in columns, further confirming layout derived from the original-format content (“defining an original page layout”). (EX1005, 63, EX1007, 91.) Pad also discloses HTML “anchors” created as separate items (“hotwords” in the HTML browser), supporting that retrieved content includes multiple hyperlinks and associated navigation behavior, which corresponds to the recited “functionality” where included (“a plurality of hyperlinks”; “functionality and design”). (EX1005, 60, 63;

EX1007, 91.) SVG is designed to replace raster graphics with a vector format that supports “text data” with “precise control of text sizing [and] positioning... so that graphic artists have a way to ensure that the end user sees what was intended.” (EX1008, 1, 7-8.) SVG ensures “true scalability,” stating that “all objects and attributes (including such things as text and linewidths) should grow/shrink uniformly with the zoom level.” (EX1008, 10, 14.)

- ix) **[1.b] “processing, via the wireless hand-held device, the markup language-based Web content with a rendering engine to generate page layout information corresponding to the original page layout of content on the Web page as interpreted by the rendering engine including a width of the Web page;”**
[42.d.2] “processing the markup language-based Web content with the rendering engine to generate page layout information corresponding to the original page layout as interpreted by the rendering engine including a width of the Web page; and”
[77.b] “processing the markup language-based Web content with the rendering engine to generate page layout information corresponding to the original page layout as interpreted by the rendering engine including a width of the Web page;”
[108.b] “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;”

48. In the Zaurus-Pad-SVG combination, the hand-held device processes the retrieved markup/HTML Web content with a browser rendering engine to generate page layout information corresponding to the Web page’s original page layout (“as designed for a desktop computer”) as interpreted by the rendering engine,

including a page width where recited (“processing ... via the wireless hand-held device”; “processing ... Web content with a rendering engine”; “generate page layout information corresponding to the original page layout as interpreted by the rendering engine”; “including a width of the Web page”). (EX1004, 40; EX1005, 60, 63; EX1007, 91.)

49. Zaurus teaches device-side rendering engine processing of markup/HTML content by disclosing an “Internet browser” that “incorporates HTML 3.2 specifications,” and processes the retrieved HTML on the device; it contemplates that “a single page is divided into a certain number of boundaries” and supports layout features, reflecting the browser’s interpretation of the original-format markup to generate the displayed layout (“processing ... Web content with a rendering engine”; “page layout information corresponding to the original page layout ... as interpreted”). (EX1004, 40, 48.) Zaurus’s disclosure of processing “HTML 3.2” satisfies [108.b]’s “HTML-based Web content,” and HTML is also “markup language-based Web content” for [1.b]/[42.d.2]/[77.b]. (EX1004, 40, 48.)

50. Pad teaches the rendering engine processing and layout information generation by teaching that when “html data is fetched, it is parsed and the HTML item is created which contains a method for rendering the page” confirming conversion of the retrieved markup/HTML into a layout representation used for rendering (“processing ... Web content with a rendering engine”; “generate page

layout information”). (EX1005, 60.) Pad also teaches generating layout information for page components via bounding boxes (e.g., “bbox” returning item extents), and Pad’s discussion of layout processing (e.g., “calculating the bounding box ... and laying out nodes”) further reflects layout information generation as part of rendering (“page layout information”; “original page layout ... as interpreted by the rendering engine”). (EX1005, 17, 22, 60; EX1007, 41.)

51. For the “width” limitation in [1.b]/[42.d.2]/[77.b], Pad teaches page-width layout information by stating that the HTML item “specifies the width ... of the html page” and that “[t]he page will be re-laid out according to the new width,” showing the rendering engine determines and uses a page-width parameter in generating layout (“including a width of the Web page”; “page layout information corresponding to the original page layout”). (EX1005, 63.) Pad further confirms renderer-interpreted layout behavior (e.g., laying out sub-documents in columns) (“original page layout ... as interpreted by the rendering engine”). (EX1007, 91.)

- x) [1.c] “**employing the page layout information to generate scalable page layout information; and**”
[77.c] “**employing the page layout information to generate scalable page layout information; and**”
[108.c] “**employing scaled page layout information and/or data derived therefrom to generate scaled Web page content at multiple scale factors; and**”

52. In the Zaurus-Pad-SVG combination, Zaurus suggests multi-scale viewing by disclosing that pages may be displayed in different magnifications (e.g.,

“reduced display” versus “magnified display”), demonstrating multiple zoom levels (“multiple scale factors”). (EX1004, 40, 46.) SVG supplies the vector-based scalable graphics model used to represent page objects so the resulting “scalable page layout information” is scalable in the construed sense across zoom levels. (EX1008, 11, 14.)

53. Pad’s HTML processing creates an internal object-based representation from the fetched HTML (“html data is fetched, it is parsed and the HTML item is created”), and Pad generates/uses geometric layout information such as bounding boxes (“bbox” returns item extents) (“page layout information”). (EX1005, 60, 17.) Pad also teaches that the HTML item “specifies the width ... of the html page” and that “[t]he page will be re-laid out according to the new width,” showing the renderer employs layout parameters (including width-driven layout results) to produce a reusable layout representation. (EX1005, 63.) Because that representation includes object geometry (such as positions/sizes/bounding boxes) that can be re-rendered under different zoom values, it is graphic content that (1) is capable of being rendered at multiple zoom levels and (2) includes one or more vectors including vectors represented by (X2, Y2) relative to origin points (X1, Y1) that define page-layout locations of objects (“scalable page layout information”).

54. Pad further discloses the “scalable” nature of that layout representation by defining magnification as “the multiplication of the current view ... with the

object's size (from the -place ... option)," and defines the view to include "zoom" where "zoom specifies the magnification." This teaches that the stored object size/placement information (derived from the initial page layout) is capable of being rendered at multiple zoom levels by changing the zoom/magnification value ("scalable page layout information"). (EX1005, 29-30.)

55. For [108.c], Pad discloses a numeric zoom parameter where "zoom specifies the magnification," and magnification is defined as multiplication of the current view with the object's stored size and placement information. (EX1005, 29-30.) Applying different zoom and magnification values to the underlying layout geometry (for example, object sizes, positions, and bounding boxes) yields "scaled page layout information and/or data derived therefrom" for each selected scale factor, which Pad then uses to render correspondingly scaled output at multiple zoom levels. Pad confirms this scaled rendering by stating that "Text and images are scaled," which discloses generation of "scaled Web page content" at different zoom levels, and thus "at multiple scale factors." (EX1005, 35; EX1004, 46.) SVG is consistent with this approach by describing zoom-level scaling behavior, which suggests that applying scale factors to layout/geometry produces uniformly scaled renderings ("scalable page layout information"; "scaled Web page content at multiple scale factors"). (EX1008, 11, 14.)

- xi) [1.d] “employing scalable page layout information and/or data derived therefrom to generate views of the Web page on the display, including,”
[42.d.3] “employing scaled page layout information and/or data derived therefrom to generate views of the Web page on a display, including,”
[77.d] “employing the scalable page layout information and/or data derived therefrom to generate views of the Web page on the display, including,”
[108.d.1] “employing the scaled Web page content to enable a user to browse the Web page at multiple zoom levels on the mobile hand-held device”

56. In the Zaurus-Pad-SVG combination, the hand-held device uses scalable or scaled page layout information and/or data derived therefrom to render different views of the Web page on the device display, including enabling browsing at multiple zoom levels (“employing scalable page layout information and/or data derived therefrom to generate views of the Web page on the display”; “employing scaled page layout information and/or data derived therefrom to generate views of the Web page on a display”; “employing the scaled Web page content to enable a user to browse the Web page at multiple zoom levels”). (EX1004, 12, 40, 46; EX1005, 29, 30, 32, 35; EX1007, 91; EX1008, 14.)

57. Zaurus provides the display and user-facing “views” of Web pages by disclosing a “Screen (display portion)” and supporting multiple on-screen viewing modes such as switching “between reduced display and magnified display,” which corresponds to user selection among different zoom levels/views on the display

(“generate views of the Web page on the display”; “browse ... at multiple zoom levels”). (EX1004, 12, 40, 46.)

58. Pad teaches generating views from scalable/scaled layout data by defining the view to include “zoom” where “zoom specifies the magnification,” and defining magnification as “the multiplication of the current view ... with the object’s size (from the -place ... option),” so the displayed view is computed from underlying object/layout size and placement data, scaled by a zoom/magnification value (“employing ... page layout information and/or data derived therefrom”; “generate views”; “multiple zoom levels”). (EX1005, 29, 30.) Pad also provides view-manipulation commands such as the “moveto” view-change command to change what portion of the surface is visible, further evidencing generation of different views from the same underlying content representation (“generate views”). (EX1005, 30.) Pad’s rendering model places content on a surface for viewing/navigation (including zooming/panning), suggesting that these computed views are displayed to the user (“on the display”; “enable a user to browse”). (EX1007, 91.)

59. SVG suggests that scalable layout data is used to generate views across zoom levels by requiring “true scalability” such that “all objects and attributes (including ... text ...) should grow/shrink uniformly with the zoom level,” which

entails employing scalable geometry/attributes to render the view at the selected scale factor (“generate views”; “multiple zoom levels”). (EX1008, 14.)

- (1) [1.d.1] “a view under which the width of the Web page is fully displayed; and”
[31] “The method of claim 1, further comprising: generating a view under which the width of the Web page is fit across the display.”
[42.d.3.i] “a view under which the width of the Web page is fit across the display; and”
[77.d.1] “a view under which the width of the Web page is fully displayed; and”
[104] “The machine-readable medium of claim 77, wherein execution of the instructions performs further operations comprising: generating a view under which the width of the Web page is fit across the display.”

60. In the Zaurus-Pad-SVG combination, the device can generate a view where the entire width of the web page fits within the display. Zaurus provides a “Switch between reduced display and magnified display,” which supports generating an overview view in which more of the page, including the page width, is visible across the screen. (EX1004, 46-47.) The “reduced” view is an overview view that displays more of the page at once on the handheld screen, and therefore can present a view in which the page width is entirely visible within the display (“fully displayed”) and thus also “fit across the display.”

61. Pad defines the view in “xview yview zoom” form where “zoom specifies the magnification,” allowing selection of a zoom level that makes a page or object fit within the window. (EX1005, 30.) Pad also explains that clicking on

text causes Pad “to pan and zoom until that text fills the screen horizontally,” demonstrating that Pad can compute and apply a zoom that fits content to the screen width. (EX1007, 108; EX1007, 54.) Pad further ties “width” to the HTML page itself, stating that the HTML item specifies the width of the HTML page and that the page will be re-laid out according to the new width. (EX1005, 63.) Setting the HTML page width to the display width, and selecting the corresponding view/zoom, yields a view under which the width of the Web page is “fit across” the display.

62. Pad can compute an image item’s bounding box (“x1 y1 x2 y2”), then adjust the view so the item (or its bounding box) “fills the specified amount of screen” via a center command or set the view magnification using a move command for “magnification of zoom.” (EX1005, 17, 22, 30, 32.) A person of ordinary skill in the art would have understood that as applied to the Web page object (or to a group representing the page), this supports a generated view that fits the page within the window; and where the page width is the dimension for the desired overview, the command yields a view in which the page width is “fit across” the display. Pad’s tutorial example confirms that Pad uses zooming to fit content horizontally; a person of ordinary skill in the art would have applied the same view selection technique to the Web page width in the HTML browser context. (EX1007, 54.) Since the “Web page” is represented on the Pad surface as group of objects in Pad, applying these commands to the page object would result in the claimed “view.”

63. SVG supports “zoom and pan” and “transformations” that allow content to be scaled to fit any “viewport” or device resolution, and zoom/pan support are consistent with selecting a view where content is scaled to fit across the display. (EX1008, 11, 14.) A person of ordinary skill in the art would have been motivated to implement a fit “width” view in the Zaurus browser using Pad’s view selection logic (including use of a page width and zoom/magnification) to improve readability on the handheld device, avoiding horizontal scrolling. This reflects routine adaptation of known zoomable rendering and view-selection techniques to handheld display constraints.

- (2) [1.d.2] “views of the Web page generated in response to associated user inputs to enable a user to iteratively zoom in on a portion of the Web page while preserving the original page layout, functionality, and design of the content on the Web page defined by the markup language-based Web content,”
[42.d.3.ii] “views of the Web page generated in response to associated user inputs to enable a user to iteratively zoom in on a portion of the Web page while preserving the original page layout, functionality, and design of the content on the Web page defined by the markup language-based Web content,”
[77.d.2] “views of the Web page generated in response to associated user inputs to enable a user to iteratively zoom in on a portion of the Web page while preserving the original page layout, functionality, and design of the content on the Web page defined by the markup language-based Web content,”

[108.d.2] “while preserving the original page layout, functionality, and design of the content on the Web page defined by the HTML-based Web content as interpreted by the rendering engine,”

[108.e.2] “wherein the user is enabled to iteratively zoom in on a portion of the Web page.”

64. In the Zaurus-Pad-SVG combination, the device responds to user inputs by changing the view scale (zoom) and re-rendering the same underlying page representation to produce “views of the Web page,” so the user can iteratively zoom in on a selected portion while the page’s layout relationships, interactive behavior, and visual design remain preserved across the different zoom levels. (EX1004, 12, 15-16, 46; EX1007, 6, 13, 54, 87-93; EX1005, 30; EX1008, 14.)

65. Zaurus provides the user-input-driven view generation on a hand-held display by disclosing a “Screen (display portion)” and touch-based interaction, including switching “between reduced display and magnified display” (“views ... generated in response to associated user inputs”). (EX1004, 12, 15-16, 46.) Pad teaches iterative zooming behavior, stating that “clicking and holding on the middle mouse button performs a smooth animated zoom in,” and that “this process can be repeated indefinitely” (“iteratively zoom in”). (EX1007, 4, 6.) Pad further teaches zooming in on a portion of content by allowing the user to click specific items (e.g., text or images) so that they fill the screen (“zoom in on a portion of the Web page”). (EX1007, 54.)

66. Pad ties zoom to a view change rather than changing the underlying page structure: the view includes a numeric “zoom” where “zoom specifies the magnification,” so the rendering system can compute different displayed views by applying different zoom values to the same underlying layout produced from the interpreted markup. (EX1005, 30.) Pad also describes how its HTML browser lays out content (e.g., placing documents on the surface and laying out sub-documents in columns), supporting that the content’s layout is defined by the interpreted markup and remains coherent as the user changes the view/zoom (“preserving the original page layout”). (EX1007, 91.) This supports [108.d.2]’s “as interpreted by the rendering engine,” because the “interpreted” layout is the engine’s parsed/laid-out page representation and zoom alters how that representation is viewed.

67. SVG’s “true scalability” requirement states that “all objects and attributes ... should grow/shrink uniformly with the zoom level,” suggesting that scalable rendering preserves the page’s design/layout relationships at different zoom levels (“preserving the original page layout ... and design”). (EX1008, 14.) Because the interactions are attached to objects in the rendered representation (for example, link and event handlers), changing zoom to generate successive views does not eliminate interactivity (“preserving ... functionality”).

68. A person of ordinary skill in the art would have been motivated to combine Zaurus’s touch-driven mobile browsing with Pad’s interactive/iterative

zooming and scalable rendering behavior (suggested by SVG's "true scalability") to generate successive zoomed views in response to user inputs while preserving the interpreted layout relationships and the page's interactive behavior. (EX1004, 46; EX1007, 6, 13; EX1008, 14.)

- xii) [1.e] "wherein preservation of the functionality defined by the markup language-based content includes preservation of hyperlink functionality."
[42.e] "wherein preservation of the functionality defined by the markup language-based content includes preservation of hyperlink functionality."
[77.e] "wherein preservation of the functionality defined by the markup language-based content includes preservation of hyperlink functionality."
[108.e.1] "wherein preservation of the functionality includes preservation of hyperlink functionality, and"

69. In the Zaurus-Pad-SVG combination, preserving the page's functionality includes preserving link navigation defined by the markup/HTML; "hyperlink functionality" remains operative in the generated/zoomed views ("wherein preservation of the functionality ... includes preservation of hyperlink functionality"). (EX1004, 10, 40, 46; EX1005, 47, 60-64; EX1007, 89, 91; EX1008, 14.)

70. Zaurus teaches that hyperlink functionality is part of the functionality defined by the retrieved markup/HTML content by explaining that "A separate page which is linked is displayed by touching the underlined characters," which enables using a hyperlink to navigate to a linked page ("hyperlink functionality"). (EX1004,

10, 40.) Zaurus also supports link-like functionality via “client-side clickable maps” (clicking inside the figure jumps to a corresponding page) and provides browsing views that include switching “between reduced display and magnified display,” consistent with hyperlink navigation remaining part of the browsing functionality as the view changes. (EX1004, 40, 46.)

71. Pad teaches that hyperlinks are represented as interactive objects and that their link behavior is preserved during interaction: for HTML items, Pad specifies that (by default) events go through the HTML object to its anchors (“-divisible” true), meaning user interaction is delivered to the anchor objects rather than being blocked at the HTML object (“preservation of hyperlink functionality”). (EX1005, 61-64.) Pad also teaches that HTML anchors are special items (“htmlanchor”) automatically grouped with the HTML object, and that anchors expose the URL addressed by the anchor, confirming that the link target remains associated with the rendered anchor object (“preservation of hyperlink functionality”; “functionality defined by the markup language-based content”). (EX1005, 63-64.)

72. Pad further confirms that link interaction remains available in a zoomed view: “Here is a zoomed in view of the document. Hotwords are shown in blue ...” and “When you follow a link, the relevant document is loaded ...,” demonstrating operable hyperlink navigation while zoomed (“preservation of hyperlink”).

functionality”). (EX1007, 89, 91.) A person of ordinary skill in the art would have understood that because the anchors remain anchor items with URLs and receive interaction events through the HTML object, the hyperlink functionality remains operative as the user changes view and zoom.

73. SVG further suggests hyperlink preservation in scalable graphics by requiring support for “Links” and stating that it should be possible to assign a link to any graphic object, and by requiring that objects/attributes grow/shrink uniformly with zoom; consistent with link targets remaining attached to their corresponding rendered objects across zoom levels (“hyperlink functionality”). (EX1008, 14.)

74. A person of ordinary skill in the art would have been motivated to combine these teachings so that, as the hand-held device renders and scales Web content across different views/zoom levels, the link behavior defined by the markup remains active (“preservation of the functionality ... includes preservation of hyperlink functionality”). (EX1004, 10; EX1005, 60-64; EX1007, 89, 91; EX1008, 14.)

4. Claims 2, 43, and 78

- i) [2] “The method of claim 1, wherein re-rendering views of the Web page is performed in real-time.”**
[43] “The apparatus of claim 42, wherein zooming views of the Web content is enabled to be performed in real-time.”
[78] “The machine-readable medium of claim 77, wherein zooming the views of the Web content is enabled to be performed in real-time.”

75. In the Zaurus-Pad-SVG combination, zoom/pan views are re-rendered in real time by maintaining an interactive frame rate during view changes (“re-rendering views ... performed in real-time”; “zooming views ... enabled to be performed in real-time”). (EX1004, 15, 46; EX1007, 4, 142; EX1005, 11; EX1008, 1, 5, 6.)

76. Pad teaches real-time re-rendering during zoom/pan by disclosing adaptive rendering control: it “monitors render times, and adapts the rendering algorithm to maintain a constant frame rate during zooms and pans,” which is an express “real-time” rendering objective during view changes (“re-rendering / zooming ... views ... in real-time”). (EX1007, 142.) Pad further supports the meaning of real-time frame-by-frame updates by specifying a target frame rate parameter for animation (e.g., a default of 20 frames per second), which requires generating each updated view in a fraction of a second (“performed in real-time”). (EX1005, 11.) Pad also characterizes zooming as a “smooth animated zoom,” reflecting continuously updated rendering responsive to user interaction (“zooming ... in real-time”). (EX1007, 4.)

77. Zaurus provides the user-facing mobile browsing context for real-time view changes by disclosing touch-driven display operations and zoom-related viewing modes such as switching between “reduced display and magnified display,” which a person of ordinary skill in the art would have understood must be rendered

responsively on the device display to be usable as an interactive browsing feature (“real-time”). (EX1004, 15, 46.)

78. SVG suggests that SVG is suitable for “interactive” and “dynamic” Web pages and listing performance-oriented requirements such as “Reasonable display performance” and being “Streamable (i.e., support progressive rendering),” consistent with real-time updating while content is manipulated or viewed (“performed in real-time”). (EX1008, 1, 5, 6.)

79. A person of ordinary skill in the art would have been motivated to combine Zaurus’s mobile Web access with Pad’s adaptive rendering techniques (suggested by SVG’s performance/scalability requirements) so that as the user zooms or pans, the device re-renders views at interactive frame rates. (EX1007, 142; EX1005, 11; EX1004, 46.)

5. Claims 3 and 71

- i) [3] “The method of claim 1, wherein the display of the wireless hand-held device comprises a touch-sensitive display configured to receive user inputs to control display of the Web page.”
[71] “The apparatus of claim 42, wherein the display comprises a touch-sensitive display configured to receive user inputs to control display of the Web page.”

80. In the Zaurus-Pad-SVG combination, Zaurus’s touch-sensitive display can receive user inputs that control Web page display (including zoom/pan view changes taught by Pad) (“touch-sensitive display configured to receive user inputs”).

to control display of the Web page”). (EX1004, 12, 15, 46; EX1007, 4, 71, 141; EX1008, 1, 14.)

81. Zaurus discloses a “Screen” and explains user interaction by “Touch[ing] the screen with a pen,” evidencing a touch-sensitive display configured to receive user inputs (“touch-sensitive display configured to receive user inputs”). (EX1004, 12, 15.) Zaurus further shows those touch inputs control Web page display, including switching “between reduced display and magnified display” (zoom control) and other display/navigation functions while browsing (“control display of the Web page”). (EX1004, 46.)

82. Pad suggests that user inputs are used to control display of content (including Web/HTML content) through interactive navigation such as “smooth animated zoom” and panning/zooming to another part of a presentation; inputs which a person of ordinary skill in the art would have understood are implemented via tapping/dragging on a touch-sensitive handheld display (“receive user inputs to control display”). (EX1007, 4, 71.) Pad also notes applicability to “handheld computers (i.e. PDA’s),” reinforcing that the contemplated input modality for such devices includes touch/stylus interaction. (EX1007, 141.)

83. SVG further suggests user-input control of displayed content by supporting event handling (e.g., click-type events) in an interactive graphics context that includes zoom/pan behavior, which a person of ordinary skill in the art would

have understood maps naturally to touch events on a small mobile device display (“receive user inputs to control display”). (EX1008, 1, 14.)

84. A person of ordinary skill in the art would have been motivated to use Zaurus’s touch-sensitive screen as the input mechanism for the combined browser so the user can directly control Web page display (e.g., zooming/panning/scrolling) via the display. (EX1004, 15, 46; EX1007, 4, 71.)

6. Claims 4, 45, and 79

- i) **[4.a] “The method of claim 1, further comprising:”**
[45.a] “The apparatus of claim 42, wherein execution of the instructions performs further operations comprising:”
[79.a] “The machine-readable medium of claim 77, wherein execution of the instructions performs further operations comprising:”
[4.b] “enabling the user to select a hyperlink; and, in response thereto,”
[45.b] “enabling the user to select a hyperlink; and, in response thereto,”
[79.b] “enabling a user to select a hyperlink; and, in response thereto,”

85. In the Zaurus-Pad-SVG combination, hyperlinks in the rendered Web page remain selectable (e.g., touching/clicking a linked word triggers loading/display of the linked content) (“enabling the user to select a hyperlink”). (EX1004, 10, 15, 40; EX1007, 60, 89, 108; EX1005, 60; EX1008, 14.)

86. Zaurus teaches hyperlink selection via touch input: it explains that “A separate page which is linked is displayed by touching the underlined characters”

(“select a hyperlink”). (EX1004, 10, 40.) Zaurus also teaches touch interaction (“Touch[ing] the screen with a pen”), suggesting that the hyperlink selection is performed via touch on the device’s display. (EX1004, 15.)

87. Pad suggests hyperlink selection in a rendered HTML view by describing hyperlinks (“hotwords ... shown in blue”) and stating that “When you click on the word, the appropriate document is placed on the Pad surface” (“enabling ... to select a hyperlink; and, in response thereto”). (EX1007, 89, 108.) Pad also teaches that “HTML anchors are created as separate items that can have events bound to them,” evidencing that hyperlinks are implemented as interactive objects selectable by the user (“select a hyperlink”). (EX1005, 60.)

88. SVG further suggests hyperlink selection as a standard capability for scalable Web content by requiring support for “Links,” including that it should be possible to assign a link to any graphic object (“select a hyperlink”). (EX1008, 14.)

89. A person of ordinary skill in the art would have been motivated to combine Zaurus’s touch-based hyperlink interaction with Pad’s anchor/event model (suggested by SVG link support) so that the user can select hyperlinks in the rendered Web page on the hand-held device. (EX1004, 10; EX1005, 60; EX1007, 89; EX1008, 14.)

ii) [4.b.1] “retrieving and processing markup language-based Web content associated with the hyperlink to

**produce additional scalable page layout information;
and”**

**[45.b.1] “retrieving and processing the markup
language-based Web content associated with the
hyperlink to produce additional page layout
information; and”**

**[79.c] “retrieving and processing the markup
language-based Web content associated with the
hyperlink to produce additional scalable page layout
information; and”**

90. In the Zaurus-Pad-SVG combination, Zaurus teaches that selecting a hyperlink causes the linked page to be retrieved and displayed: “A separate page which is linked is displayed by touching the underlined characters” (“retrieving ... Web content associated with the hyperlink”). (EX1004, 10.)

91. Pad teaches both retrieval and processing of the linked markup to create a rendered representation. Pad explains that HTML items “know about the internet” and can “automatically fetch a file from a URL,” and that “When the html data is fetched, it is parsed and the HTML item is created which contains a method for rendering the page” (“retrieving and processing ... Web content”). (EX1005, 60.) Pad’s confirms the hyperlink-follow behavior at the UI level: “When you follow a link, the relevant document is loaded into Pad.” (EX1007, 91.)

92. SVG suggests that links and zoom/pan are intended to work with true scalability: “all objects and attributes ... should grow/shrink uniformly with the zoom level” and that “It should be possible to assign a link to any graphic object or group,” consistent with scalable, link-bearing layout objects. (EX1008, 14.)

93. A person of ordinary skill in the art would have understood that this newly loaded/created HTML item provides “additional page layout information” (claim 45) because it is a new rendered page object with its own layout/structure produced by the parsing/rendering step. (EX1004, 10; EX1005, 60; EX1007, 91.) As for the “additional scalable page layout information” recited in claims 4 and 79, Pad’s object model is scalable: “pathName scale ... multiplies the size of items by a scale amount.” Thus, once the linked-page HTML item is created, it is an item whose layout representation can be scaled. (EX1005, 36; EX1008, 14.)

- iii) **[4.b.2] “employing the additional scalable page layout information and/or data derived therefrom to generate a view of the Web content associated with the hyperlink.”**
[45.b.2] “employing the additional page layout information and/or data derived therefrom to generate a view of the Web content associated with the hyperlink.”
[79.d] “employing the additional scalable-page layout information and/or data derived therefrom to generate a view of the Web content associated with the hyperlink.”

94. In the Zaurus-Pad-SVG combination, once the linked document is retrieved and processed into an additional (scalable) layout representation, the device uses that additional layout information to render/display the linked page (“employing the additional [scalable] page layout information ... to generate a view of the Web content associated with the hyperlink”). (EX1004, 10; EX1007, 91, 93; EX1004, 10.)

95. Pad teaches generating a view of the linked-page content from the newly created linked-page representation: after following a link, “the relevant document is loaded,” and the linked document is then “placed on the surface to the right of the original document, at a smaller scale.” (“additional [scalable] page layout information”; “generate a view of the Web content associated with the hyperlink”). (EX1007, 91.) Pad further teaches that the user can “zoom in on the sub-document to read it” (“additional [scalable] page layout information”; “generate a view”). (EX1007, 91, 93.)

96. Zaurus suggests that link selection results in “separate page which is linked is displayed” (“generate a view of the Web content associated with the hyperlink”). (EX1004, 10.) SVG suggests the scalability for such views (objects/attributes grow/shrink uniformly with zoom), consistent with using scalable layout information to generate rendered views of linked content across scale factors. (EX1008, 14.)

7. Claims 5, 46, and 80

- i) [5.a] “The method of claim 1, further comprising:”**
[46.a] “The apparatus of claim 42, wherein execution of the instructions performs further operations comprising:”
[80.a] “The machine-readable medium of claim 77, wherein execution of the instructions performs further operations comprising:”
[5.b] “parsing and processing markup language code with the rendering engine to determine the original page layout of display content within the Web page as

interpreted by the rendering engine, wherein the original page layout defines a layout location for a plurality of objects, including text objects associated with the text content and image objects associated with the plurality of images included in the Web page; and”

[46.b] “parsing and processing markup language code with the rendering engine to determine the original page layout of display content within the Web page as interpreted by the rendering engine, wherein the original page layout defines a layout location for a plurality of objects, including text objects associated with the text content and image objects associated with the plurality of images included in the Web page; and”

[80.b] “parsing and processing markup language code with the rendering engine to determine the original page layout of display content within the Web page as interpreted by the rendering engine, wherein the original page layout defines a layout location for a plurality of objects, including text objects associated with the text content and image objects associated with the plurality of images included in the Web page; and”

97. In the Zaurus-Pad-SVG combination, Zaurus suggests that the markup being processed is HTML and that the original markup-defined layout/content is what the browser interprets for display, stating the browser “incorporates HTML 3.2 specifications,” supports “Pages composed of frames,” and includes “text data” and image data (e.g., GIF/JPEG). (EX1004, 40, 42.)

98. Pad teaches that, after Web content is obtained, the rendering engine parses and processes markup-language code to construct a representation: “HTML items ... automatically fetch a file from a URL ... [and] When the html data is

fetches, it is parsed and the HTML item is created which contains a method for rendering the page” (“parsing and processing markup language code with the rendering engine to determine the original page layout ... as interpreted by the rendering engine”). (EX1005, 60.)

99. That renderer-determined layout defines layout locations for the resulting objects. Pad explains that its HTML browser “will lay out sub-documents in two columns next to the parent document,” reflecting that the layout process places multiple objects at specific locations in the page (“wherein the original page layout defines a layout location for a plurality of objects”). (EX1007, 91.) Pad further teaches that “Every object has a -place which specifies the anchor point of that object,” evidencing that objects resulting from the layout have location data (coordinates/anchor points) derived from the layout process (“defines a layout location”). (EX1005, 50.)

100. Pad also identifies the relevant object types created/used by the rendering engine: “Text Items ... displays a string of characters” and “Image Items ... appear on the display as color images” (“objects ... including text objects associated with the text content and image objects associated with the plurality of images”). (EX1005, 75, 64.)

- ii) **[5.c] “for each object, defining an object datum corresponding to the layout location for the object on the original page layout; and”**

[46.c] “for each object, defining an object datum corresponding to the layout location for the object on the original page layout; and”

[80.c] “for each object, defining an object datum corresponding to the layout location for the object on the original page layout; and”

101. In the Zaurus-Pad-SVG combination, the device defines a reference point (datum) for each object that corresponds to its position in the layout. Pad discloses that for every item (object), the system defines an “anchor” (“object datum”) which “tells how to position the object relative to the positioning point.” (EX1005, 47.) Pad also calculates a “bounding box” for every object. (EX1005, 17.) Pad maps coordinates to these boxes: “A frame is a bounding box, and all coordinates are specified within the unit square where the unit square is mapped to the frame.” (EX1005, 34.) A person of ordinary skill in the art would have understood that this anchor/coordinate definition corresponds to the “layout location” determined during the parsing of the “original page layout” (e.g., the position defined by the HTML stream).

102. SVG discloses defining “local user coordinates” for objects and using “Transformations” (e.g., translation matrices) to define the object’s position. (EX1008, 11.) SVG further discloses utility methods to “query an object’s bounds” (“layout location”). (EX1008, 15.)

103. A person of ordinary skill in the art would have been motivated to use Pad’s anchor and bounding box model in the Zaurus browser to manage the scalable

vector objects in view of SVG. For each text or image object parsed from the HTML, the engine defines an “anchor” (“object datum”) that corresponds to its calculated X/Y position (“layout location”) on the page. (EX1005, 47, 65.)

- iii) [5.d] “associating the object and its object datum.”
[46.d] “associating the object and its object datum.”
[80.d] “associating the object and its object datum.”

104. In the Zaurus-Pad-SVG combination, once the original page layout is determined, each text/image object is assigned an anchor/positioning point (and associated derived bounds) that corresponds to the object’s layout location. (EX1005, 47, 50; EX1005, 17, 32, 34; EX1008, 4, 11, 15.)

105. Pad explains that “[e]very object has a -place” which “specifies the anchor point of that object” and that the “place specifies the object’s position and size” (“for each object, defining an object datum corresponding to the layout location for the object on the original page layout”). (EX1005, 50.) Pad commands like “itemconfigure” allow modifying this datum for a specific object ID, confirming the association. (EX1005, 32.) Pad further explains that the -anchor setting “tells how to position the object relative to the positioning point for the item (see -place),” confirming that the object’s positioning/anchor point is the reference datum used to place the object at its layout location. (EX1005, 47.)

106. Pad also provides derived location data for each object via bounding boxes: “bbox” returns an item’s bounding box (“x1 y1 x2 y2”) such that the drawn

area is within the returned coordinates, which is data derived from the object's placement/location and usable to represent the object's layout extent. (EX1005, 17.) Pad's coordinate-frame discussion likewise ties layout to bounding regions ("A frame is a bounding box ... [with] the unit square ... mapped to the frame"), suggesting that object placement/location is represented using defined coordinate mappings anchored to bounding regions. (EX1005, 34.)

107. SVG suggests the same per-object datum/position concept in scalable graphics by supporting "local user coordinates" and nested 2×3 matrix transformations (including translation); coordinate/transform data that positions each object in the page layout, and by describing utility methods to query an object's bounds. (EX1008, 4, 11, 15.)

8. Claims 6, 47, and 81

- i) [6] "The method of claim 5, further comprising translating at least one image object from an original format into a scalable format."
[47] "The apparatus of claim 46, wherein execution of the instructions performs further operations comprising translating at least one image object from an original format into a scalable format."
[81] "The machine-readable medium of claim 80, wherein execution of the instructions performs further operations comprising translating at least one image object from an original format into a scalable format."

108. In the Zaurus-Pad-SVG combination, at least one retrieved image file in an "original format," such as GIF or JPEG (each a particular way that image

information is encoded), is translated (converted) from that encoded file format into an internal Pad “image” object representation that is capable of being rendered at multiple zoom levels (“translating ... from an original format into a scalable format”). (EX1004, 40; EX1005, 17, 36, 64.) Zaurus teaches that the browser can display images “not only in GIF format, but also ... JPEG format,” (“image object”; “original format”). (EX1004, 40.) A person of ordinary skill in the art would have understood GIF and JPEG identify encoded image file formats, which Pad reads and converts for display.

109. Pad teaches importing raster images and converting them into Pad “image” objects: “allocimage” reads image files (including GIF, and optionally JPEG) and creates an image token used to create an “image item,” and Pad explains that “Items of type image appear on the display as color images” (“translating ... image object”; “scalable format”). (EX1005, 17, 64.) Pad further teaches that these image items reside in a scalable layout model because items have vector placement/size controls (e.g., “-place” specifies an object’s anchor point and controls its position/size), and because Pad scales items via view magnification and scaling commands: “Magnification is defined as the multiplication of the current view ... with the object’s size,” and “scale...multipl[ies] the size of the item.” (EX1005, 29, 35, 36, 50.) Thus, once the encoded image file is translated (converted) into a Pad image item, the image is present in a “scalable format”

because the image item is rendered within Pad's scalable coordinate and zoom system and is capable of being rendered at multiple zoom levels by changing the view magnification and or scaling the item's size ("translating ... into a scalable format").

110. Pad confirms the effect of this conversion because images become zoomable or scalable objects in the interface (users can zoom in to see images "in greater detail"), consistent with the image being capable of being rendered at multiple zoom levels after translation from its original file format. (EX1007, 80.) SVG further confirms that images are intended to participate in scalable/zoomable rendering, stating that "images and text should scale along with the sizes of the shapes," consistent with translating image content into a scalable rendering model. (EX1008, 11.)

9. Claims 7, 48, and 82

- i) [7.a] "The method of claim 5, further comprising:"
[48.a] "The apparatus of claim 46, wherein execution of the instructions performs further operations comprising:"
[82.a] "The machine-readable medium of claim 80, wherein execution of the instructions performs further operations comprising:"
[7.b] "associating attribute information including a color and typefont for each text object; and"
[48.b] "associating attribute information including a color and typefont for each text object; and"
[82.b] "associating attribute information including a color and typefont for each text object; and"**

111. In the Zaurus-Pad-SVG combination, the rendering engine associates per-text-object formatting attributes such as font/typefont and color with the text objects created from the markup/HTML (“associating attribute information including a color and typefont for each text object”). (EX1005, 8, 75, 78; EX1007, 89; EX1008, 10, 13.)

112. Pad teaches that the rendering system creates text objects as distinct items: “Text Items” are items that “display[] a string of characters on the screen” (“text object”). (EX1005, 75.) Pad further teaches that each such text item (and related item types including HTML) supports per-item configuration options that include both font/typefont and color: “-font fontname...Specifies the font to be used for rendering text for this item,” including “Adobe Type 1 font,” and “-pen color” specifies the pen color for the item (“color”; “typefont”; “for each text object”). (EX1005, 78.) Because these are item configuration options, a person of ordinary skill in the art would have understood the system necessarily associates (stores/maintains) the specified font and pen/color attributes with the particular text item to render it accordingly (“associating attribute information ... for each text object”). (EX1005, 78.) Pad also illustrates color association by configuring items with pen colors (e.g., “-pen red”) (“color”). (EX1005, 8.) Pad’s HTML browsing example further demonstrates per-text color behavior for hyperlinks/hotwords (“Hotwords are shown in blue ... [and] changes its color to red”), suggesting that

text objects are associated with color attributes during rendering and interaction (“associating ... a color ... for each text object”). (EX1007, 89.)

113. SVG suggests that text/font attributes and color are standard, renderer-applied attributes for text objects, stating that “All CSS text/font attributes ... should be honored,” and describing fill/color support applicable to “shape or text object,” consistent with associating font and color attributes with text objects during rendering (“typefont”; “color”). (EX1008, 13, 10.)

- ii) **[7.c] “employing the attribute information to render text content corresponding to text objects shown in views of the Web page.”**
[48.c] “employing the attribute information to render text content corresponding to text objects shown in views of the Web page.”
[82.c] “employing the attribute information to render text content corresponding to text objects shown in views of the Web page.”

114. In the Zaurus-Pad-SVG combination, the rendering engine uses the previously associated per-text-object attributes (e.g., font/typefont and color) when drawing text items on the display (“employing the attribute information to render text content corresponding to text objects shown in views of the Web page”). (EX1005, 75, 78; EX1007, 89; EX1008, 8, 10, 13.)

115. Pad teaches that a “Text item displays a string of characters on the screen,” and that the rendering engine uses the item’s stored attributes when rendering: “-font...Specifies the font to be used for rendering text for this item,” and

“-pen...specifies a color to use for drawing the text characters” (“employing the attribute information to render text content corresponding to text objects shown in views of the Web page”). (EX1005, 75, 78.) Pad’s HTML browsing example further suggests active use of text-color attributes during rendering of a view (e.g., “Hotwords are shown in blue ... [and] changes its color to red”), demonstrating that the renderer applies the stored attribute information to draw/update the visible text objects. (EX1007, 89.)

116. SVG suggests that conforming processors must apply text/font and paint/color attributes when rendering text, stating under CSS support that “All CSS text/font attributes ... should be honored,” and describing rendering/painting of text using specified font and paint properties (e.g., fill). (EX1008, 8, 10, 13.)

10. Claim 8

- i) [8.a] “The method of claim 5, further comprising effecting a zoom operation by, for each of a plurality of objects to be included in a zoomed view of the Web page to be rendered on the display,”**

117. In the Zaurus-Pad-SVG combination, a zoom operation is effected by (1) determining which page objects are included in the zoomed view (i.e., which objects are visible in the new view) and then (2) rendering those objects on the display, which necessarily entails iterating for each object in that included set (“effecting a zoom operation by, for each of a plurality of objects to be included in”).

a zoomed view ... to be rendered on the display”). (EX1004, 46; EX1007, 141; EX1005, 59; EX1008, 14.)

118. Pad teaches view-dependent selection of objects to render, stating under “Spatial Indexing” that “Pad++ uses an R-Tree internally to quickly determine which objects are visible in a given view,” which identifies the “plurality of objects to be included in a zoomed view” after the view changes during zoom. (EX1007, 141.) Pad further explains that “When things become too small to see, Pad++ ceases to render them,” confirming object-by-object decisions tied to the current (zoomed) view. (EX1007, 141.) Pad also suggests the required per-object rendering iteration by explaining that objects are rendered from an internal ordered structure: “Group members are rendered sequentially in the display list,” evidencing that rendering proceeds object-by-object for the objects included in the (zoomed) view. (EX1005, 59.)

119. Zaurus provides the handheld display context and the user-triggered zoom operation, teaching the user can “Switch between reduced display and magnified display.” (EX1004, 46.) SVG suggests zooming and rendering of multiple object types by requiring that “An SVG processor should support zoom and pan on graphics,” and explaining that “all objects and attributes ... should grow/shrink uniformly with the zoom level,” consistent with rendering the included objects for the zoomed view. (EX1008, 14.)

ii) [8.b] “scaling page layout information associated with the object using a scale factor corresponding to a zoom level associated with the zoom operation to determine a scaled datum; and”

120. In the Zaurus-Pad-SVG combination, the rendering engine determines a scaled datum for each object by applying the zoom’s scale factor to that object’s page-layout information (its stored placement/size/coordinates), thereby computing the object’s scaled placement for the zoomed view (“scaling page layout information ... using a scale factor corresponding to a zoom level ... to determine a scaled datum”). (EX1005, 29, 30, 35 (“Text and images are scaled”), 36 (“scaleAmount”), 50, 60 (“html data is fetched”); EX1008, 11, 14.)

121. Pad teaches that each object carries page layout information that includes its placement and size: “Every object has a -place which specifies the anchor point of that object,” and the place “specifies the object’s position and size” (“layout information associated with the object”). (EX1005, 50.) Pad then teaches scaling by a view zoom factor: “Magnification is defined as the multiplication of the current view ... with the object’s size,” and defines the view as “xview yview zoom” where “zoom specifies the magnification” (“scale factor corresponding to a zoom level”). (EX1005, 29, 30.) A person of ordinary skill in the art would have understood this multiplication applies the zoom scale factor to the object’s layout coordinates/size to produce the object’s scaled placement (“scaled datum”) for rendering in the zoomed view. (EX1005, 29, 30, 50.)

122. Pad suggests the resulting transformed coordinate concept by distinguishing between an object's local coordinates and the global coordinate system used after applying transforms, consistent with determining a scaled/transformed datum for the object under the current view. (EX1005, 23.) SVG suggests the same operation by requiring that "all objects and attributes ... should grow/shrink uniformly with the zoom level," which entails applying the zoom scale factor to the object's layout attributes. (EX1008, 14.) SVG further describes coordinate and transformation behavior using nested 2x3 matrix transformations, including scaling, which mathematically produces scaled coordinates for object placement ("scaled datum"). (EX1008, 11.)

iii) [8.c] "scaling content associated with the object using the scale factor and rendering the scaled content at a location corresponding to the scaled datum."

123. In the Zaurus-Pad-SVG combination, after the scaled datum is determined for an object (e.g., scaled placement coordinates under the current zoom), the rendering engine scales the object's associated content (text, images, shapes) ("scaling content ... using the scale factor and rendering the scaled content at a location corresponding to the scaled datum"). (EX1005, 25, 26, 29, 36; EX1008, 11, 14.)

124. Pad defines magnification as applying the view factor to the object ("Magnification is defined as the multiplication of the current view ... with the

object's size"), which means the object's drawn content is scaled under the zoom factor. (EX1005, 29.) Pad also supports scaling of items via the scale operation (multiplying the size of items), which includes text and image items. (EX1005, 36.)

125. Pad further teaches rendering the scaled content at specific coordinates such as at locations corresponding to the "scaled datum." For text, drawtext draws text at "xloc yloc"; for images, drawimage draws the image at "(x, y)." These rendering commands place content at specified coordinates that, in a zoomed view, correspond to the "scaled datum" produced by the view transform. (EX1005, 26, 25.)

126. SVG suggests that content is scaled and rendered under zoom: it states that "images and text should scale along with the sizes of the shapes," and that an SVG processor should support "zoom and pan ... with true scalability," consistent with scaling object content by the zoom factor and rendering it at the corresponding transformed coordinates. (EX1008, 11, 14.)

11. Claim 9

- i) [9.a] "The method of claim 8, further comprising effecting a zoom operation combined with a pan operation by,"**

127. In the Zaurus-Pad-SVG combination, the rendering system implements a combined zoom-and-pan by applying a single view change that updates both (i) the magnification/scale factor (zoom) and (ii) the view's center/position coordinates

(pan) (“effecting a zoom operation combined with a pan operation”). (EX1004, 46; EX1005, 30, 46; EX1004, 46; EX1008, 14.)

128. Pad defines the view as “xview yview zoom” and provides moveto to “Set the view position and magnification” in one operation to simultaneously change the view center (pan) and the magnification (zoom) (“zoom operation combined with a pan operation”). (EX1005, 30.) Pad also teaches zooming about a specified point (zoom around a point), which combines zoom with a pan so the selected point remains at the intended screen location during zoom (“combined with a pan operation”). (EX1005, 46.)

129. Zaurus suggests the handheld browsing context by teaching user-selectable reduced/magnified display modes (“Switch between reduced display and magnified display”), consistent with zooming behavior, and user navigation of displayed content. (EX1004, 46.) SVG suggests that the graphics processor should support “zoom and pan on graphics,” consistent with combining zoom and pan in interactive view changes. (EX1008, 14.)

ii) [9.b] “determining an offset corresponding to the pan operation; and”

130. In the Zaurus-Pad-SVG combination, a pan changes the view position by translating the view center from its current coordinates to new coordinates (“determining an offset corresponding to the pan operation”). (EX1005, 30, 38; EX1008, 11.)

131. Pad teaches that a view is defined by “xview yview zoom,” where “xview, yview specifies the point at the center of the window,” meaning a pan operation is accomplished by changing the view-center coordinates. (EX1005, 30.) Pad’s “moveto” sets the view position (center point) (and magnification) based on supplied coordinates, which determines what change in view position is needed for the offset/translation (“determining an offset”). (EX1005, 30.)

132. Pad also suggests translation offsets by adding “dx and dy” to coordinates as a computed offset vector for a translation operation. (EX1005, 38.) SVG similarly teaches translation components as part of its “2x3 matrix transformations,” where translation terms represent the offset used to pan/translate content within the coordinate system. (EX1008, 11.)

133. Accordingly, whether expressed as (a) new xview, yview center coordinates in Pad or (b) translation terms in an SVG transform matrix, the combined system determines a translation offset corresponding to the pan operation. (EX1005, 30; EX1008, 11.)

iii) [9.c] “employing the offset to at least a portion of the content that has been scaled and rendered to produce a view of the web page on the display.”

134. In the Zaurus-Pad-SVG combination, after content is scaled for the zoom level, the rendering engine applies the pan translation (“offset”) so that the scaled content is positioned correctly in the display window, thereby producing the

final view on the display (“employing the offset to at least a portion of the content that has been scaled and rendered to produce a view of the web page on the display”).

(EX1005, 29, 30, 32; EX1008, 11.)

135. Pad teaches that the view transform used for rendering includes both zoom (scaling) and view position (translation). Pad defines magnification as “the multiplication of the current view ... with the object’s size,” and defines the current view as “xview yview zoom,” where “xview, yview specifies the point at the center of the window.” Thus, the renderer’s view transform uses both the scale factor (zoom) and the view’s position coordinates (“offset”) when rendering objects. (EX1005, 29, 30.)

136. Pad further suggests applying the offset to produce the view by explaining that “moveto...Change[s] the view” using specified coordinates; updating the view center used to render the scene shifts at least the newly exposed portion of the already-scaled content into the window to produce the displayed view (“employing the offset ... to produce a view ... on the display”). (EX1005, 32.)

137. SVG suggests that translation offsets are applied during rendering via transforms by describing nested “2x3 matrix transformations” that are “sufficient to achieve any types of translation.” (EX1008, 11.)

12. Claim 10

- i) **[10.a] “The method of claim 8, wherein rendering scaled content associated with a text object comprises:”**
[10.b] “employing presentation attributes for the text object, the presentation attributes including a font typeface, size and color;”

138. In the Zaurus-Pad-SVG combination, rendering scaled content for a text object uses presentation attributes associated with that text object (including typeface/font, size, and color) when drawing the text in the zoomed view (“employing presentation attributes for the text object, the presentation attributes including a font typeface, size and color”). (EX1005, 75, 77, 78, 37, 50; EX1008, 13.)

139. Pad teaches that “Text items” are text objects. (EX1005, 75.) Pad further teaches that text items have a “font” option: “Specifies the font to be used for rendering text,” and that the font file may be an “Adobe Type 1 font” (“typeface”). (EX1005, 78.) Pad also teaches a color attribute for text items via -pen color, which “specifies a color to use for drawing the text characters” (“color”). (EX1005, 78.) Pad discloses “setfontheight,” which sets the text size used in rendering (“size”). (EX1005, 37.) Pad also ties item size to placement via -place, which “specifies the object’s position and size,” and supports changing size through scaling commands. (EX1005, 50.)

140. SVG teaches that conforming processors should honor “CSS text/font attributes,” suggesting that text rendering employs font/typeface and size attributes derived from the content’s styling, and that such attributes are treated as presentation attributes for text objects. (EX1008, 13.)

- ii) **[10.c] “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 by the operating system relative to a location associated with the scaled datum for the text object, and wherein the scale applied to the scalable font is a function of at least one of the scale factor and the font size.”**

141. In the Zaurus-Pad-SVG combination, the rendering engine uses a scalable font for the selected typeface and applies the text-color attribute when drawing the text object; the text is rendered at a screen location determined from the text object’s scaled datum under the current view transform; and the applied font scale depends on the zoom scale factor and/or the specified font size (“employing a scalable font ... render ... in a color ... relative to a location associated with the scaled datum ... scale ... is a function of ... the scale factor and the font size”). (EX1005, 12, 29, 30, 77, 78; EX1004, 18; EX1008, 14.)

142. Zaurus discloses the device includes an operating system environment used to run applications such as the browser (“wherein the text is rendered by the operating system”). (EX1004, 18.)

143. Pad teaches that a text item's "-font fontname...Specifies the font to be used for rendering text," and that the font file "contains an Adobe Type 1 font." (EX1005, 78.) Pad "remembers what size, font and bitmap it used" when it last drew a character and, if the character is drawn again "at the same size and font," Pad "reuses the last bitmap image ... rather than generating the bitmap ... from its outline description." (EX1005, 12.) Pad also explains that if font caching is disabled, "characters are always drawn from their outline descriptions," confirming that Type 1 text rendering is based on outline descriptions that would be "scalable." (EX1005, 12.) Pad also teaches rendering text in the specified color using "-pen color," which "specifies a color to use for drawing the text characters." (EX1005, 78.)

144. Pad defines the view as "xview yview zoom" and defines "Magnification" as "the multiplication of the current view ... with the object's size," meaning an object's placement is transformed by the view (including zoom and view position) to determine its rendered placement in the window ("relative to a location associated with the scaled datum for the text object"). (EX1005, 29, 30.) Pad also provides drawing at specific coordinates (e.g., "drawtext" draws text at "xloc yloc"). (EX1005, 26.)

145. Pad's view includes "zoom" (the scale factor) and applies that factor in its magnification definition ("the multiplication of the current view ... with the object's size"). (EX1005, 29, 30.) Pad also provides font-size control via

“setfontheight” height, evidencing font size as an input to rendering scale. (EX1005, 37.) Consistent with font scale being a function of at least one of zoom and font size, Pad’s Type 1 text rendering caches bitmaps keyed by “size” and “font” and reuses the bitmap only when the character is drawn again at the same size and font; otherwise it regenerates the bitmap from the glyph’s outline description, which demonstrates the applied glyph scale varies with rendering size (as driven by zoom and/or specified font height) (“wherein the scale applied to the scalable font is a function of at least one of the scale factor and the font size”). (EX1005, 12, 29-30, 37.) SVG states that “all objects and attributes (including ... text ...) should grow/shrink uniformly with the zoom level,” suggesting that font scaling is a function of the zoom scale factor. (EX1008, 14.)

13. Claims 11, 49, and 83

- i) [11] “The method of claim 1, further comprising enabling the Web content to be displayed at different resolutions by scaling scalable page layout information to determine corresponding layout locations for the Web content on at least a portion of a scaled Web page for each of a plurality of scale factors.”**
[49] “The apparatus of claim 42, wherein execution of the instructions performs further operations comprising enabling the Web content to be displayed at different resolutions by scaling the page layout information to determine corresponding layout locations for the Web content on at least a portion of a scaled Web page for each of a plurality of scale factors.”

[83] “The machine-readable medium of claim 77, wherein execution of the instructions performs further operations comprising enabling the Web content to be displayed at different resolutions by scaling the scalable page layout information to determine corresponding layout locations for the Web content on at least a portion of a scaled Web page for each of a plurality of scale factors.”

146. In the Zaurus-Pad-SVG combination, the device displays Web content at different effective “resolutions” (i.e., at different zoom/magnification levels) by scaling the layout representation using multiple scale factors and, for each scale factor, determining the corresponding on-screen layout locations used to render the content (“displayed at different resolutions by scaling ... [scalable] page layout information to determine corresponding layout locations...on at least a portion of a scaled Web page for each of a plurality of scale factors”). (EX1005, 29, 30, 50; EX1007, 4, 54; EX1008, 1, 14.) Also see my explanations for Claims 8 and 9. (Sections IX.A.10 (scaling content/rendering), 11 (scale factor/offset via Pad view).)

147. Pad teaches scaling the layout representation by a scale factor associated with the view: it defines “zoom” as the magnification parameter of the current view, and defines “Magnification” as multiplying the current view by the object’s size; the stored object size/placement (layout representation) is scaled by the zoom factor (“scaling ... page layout information”; “for each of a plurality of scale factors”). (EX1005, 29, 30.) Pad also teaches that each object has an anchor point and size via the “-place” option, meaning the system has a stored layout

representation that, when scaled by the current zoom, yields the corresponding display-space positions/sizes for rendering (“determine corresponding layout locations”). (EX1005, 50.)

148. Pad discloses “smooth animated zoom,” evidencing view updates across many intermediate zoom values, and it provides a “maxZoom” option giving “16 orders of magnitude of zooming” (“for each of a plurality of scale factors”; “displayed at different resolutions”). (EX1007, 4; EX1005, 13.) Pad also teaches zooming on a page by zooming in so selected text “fills the screen horizontally” which a person of ordinary skill in the art would have understood requires scaling the layout and recomputing the corresponding layout locations for that portion at the selected zoom factor (“at least a portion of a scaled Web page”; “determine corresponding layout locations”). (EX1007, 54.)

149. SVG suggests that this scaled rendering enables display across different device resolutions and zoom levels: SVG is designed to be “displayable on a wider range of ... device resolutions,” and it requires “true scalability” such that “all objects and attributes ... should grow/shrink uniformly with the zoom level,” consistent with recomputing layout locations from scaled layout data for each zoom factor. (“displayed at different resolutions”; “scaled ... for each of a plurality of scale factors”). (EX1008, 1, 14.)

150. Accordingly, scaling Pad's stored layout representation (page layout information / scalable page layout information) by multiple zoom/scale factors determines the corresponding on-screen layout locations for rendering Web content, which enables the content to be displayed at different resolutions for at least portions of a scaled Web page. (EX1005, 29, 30, 50; EX1007, 4, 54; EX1008, 1, 14.)

14. Claims 12, 50, and 84

- i) **[12.a] "The method of claim 1, further comprising:"**
[50.a] "The apparatus of claim 42, wherein execution of the instructions performs further operations comprising:"
[84.a] "The machine-readable medium of claim 77, wherein execution of the instructions performs further operations comprising:"
[12.b] "enabling a user of the wireless hand-held device to select an area of the display to zoom in on via the display; and"
[84.b] "enabling a user of the wireless hand-held device to select an object on the display to zoom in on via the display; and"
[50.b] "enabling a user of the apparatus to select an area of a view to zoom in on via the apparatus; and"
[84.c] "in response thereto, re-rendering the display such that content associated with the object is rendered to fit across a width of the display."

151. In the Zaurus-Pad-SVG combination, the hand-held device enables a user to select (i) an on-screen area (or (ii) an on-screen object) to zoom into, and in response re-renders so the selected content fits across the display width. ("select an [area/object] [of the display/of a view] to zoom in on via the [display/apparatus] ...

[and] re-rendering ... to fit across a width of the display”). (EX1004, 12, 15; EX1007, 54; EX1005, 22.)

152. Zaurus discloses stylus/touch interaction on its screen: “Touch the screen with a pen” (“select ... via the display / via the apparatus”). (EX1004, 12, 15.) Pad discloses object-targeted zoom based on user selection: “When you click on a piece of text, Pad++ zooms in so that that text fills the screen horizontally...” (“select an [area/object] ... to zoom in on”). (EX1007, 54.)

153. Pad discloses zooming so the selected text “fills the screen horizontally” (“re-rendering ... rendered to fit across a width of the display”). (EX1007, 54.) Pad further suggests this fit-to-screen behavior via its view command that centers a selected item and sets zoom so “the largest dimension of its bounding box fills the specified amount of screen.” (EX1005, 22.)

154. A person of ordinary skill in the art would have applied Pad’s object/area-based click-to-zoom (fit-to-width) behavior to Zaurus’s touch-screen inputs so a user can tap/select a region or object on the display and the device re-renders the view to fit the selected content across the display width. (EX1004, 15; EX1007, 54; EX1005, 22.)

- ii) **[12.c] “in response thereto, re-rendering the display such that content in the area that is selected is enlarged.”**
[50.c] “in response thereto, regenerating the view such that content in the area that is selected is enlarged.”

155. In the Zaurus-Pad-SVG combination, the device responds to the selection by zooming the view, which causes the rendering engine to redraw the content in the selected area at a larger scale. (EX1005, 22, 29; EX1004, 46-47; EX1007, 54; EX1008, 14.)

156. Pad states that in response to clicking the text, “Pad++ zooms in so that that text fills the screen horizontally” (“re-rendering the display”). (EX1007, 54.) Because the text previously occupied only a portion of the screen and now fills the screen, it becomes “enlarged.” Pad confirms that “Magnification is defined as the multiplication of the current view... with the object’s size,” and that the system renders objects based on this calculated view. (EX1005, 29.) When the “center” command is executed, it updates the view transform, causing the display to be re-rendered with the selected bounding box magnified to fit the window. (EX1005, 22.)

157. SVG mandates that “all objects and attributes... should grow/shrink uniformly with the zoom level,” ensuring that the content within the selected area is enlarged with high quality “true scalability” during the re-rendering process. (EX1008, 14.)

158. A person of ordinary skill in the art would have understood that in the combined device, tapping an area on the screen triggers the rendering engine to

calculate a new zoom factor that fits the selected area to the display and re-renders the content at that enlarged scale.

15. Claims 13, 51, 86, and 114

- i) [13] “The method of claim 1, further comprising enabling a user to pan views of the Web page in response to a corresponding user input.”
[51] “The apparatus of claim 42, 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 input.”
[86] “The machine-readable medium of claim 77, 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.”
[114.a] “The method of claim 108, further comprising:”
[114.b] “in response to a user input to pan a current view of the Web page to a new view displaying a portion of the Web page that is entirely not shown in the current view,”

159. In the Zaurus-Pad-SVG combination, Zaurus supplies user-driven navigation inputs on a hand-held display, while Pad and SVG supply the panning/view-update mechanics (including re-rendering newly exposed portions). (EX1004, 43; EX1005, 11, 30; EX1007, 71.) SVG supports “zoom and pan on graphics,” suggesting panning of scalable content in response to user interaction (“pan views”). (EX1008, 14.) The user can pan the currently displayed view of a Web page in response to user input, including panning from a current view to a new view that displays a different portion of the page not shown in the current view

(“enabling a user to pan [a view/views] of the Web page ... in response to [a corresponding] user input [to pan a current view ... to a new view displaying a portion ... entirely not shown in the current view]”).

160. Zaurus teaches user-controlled scrolling/navigation of content that exceeds the screen, including touch-driven cursor/scroll operations (i.e., user input causes the displayed portion to shift), which a person of ordinary skill in the art would have understood as panning the view of the page content (“pan ... in response to a corresponding user input”). (EX1004, 43.)

161. Pad explains that “Pad++ normally does fast pans ... re-rendering the new portion,” (“pan views”; “to a new view displaying a portion ... not shown”). (EX1005, 11.) Pad also defines the view in terms of view coordinates and supports changing the view position without changing zoom (e.g., moving the view location), which is panning (“pan a current view ... to a new view”). (EX1005, 30.) Pad ties this to user input, stating that “at any time you can use the mouse to pan and zoom to another part of the presentation,” which encompasses panning far enough that the new view can show a portion not previously displayed (“in response to a corresponding user input...to a new view displaying a portion ... entirely not shown”). (EX1007, 71.) Pad’s “fastPan” explains it copies the unchanged screen portion for “re-rendering the new portion” of the screen. (EX1005, 11.)

16. Claims 14, 52, 87, 114, and 115

- i) [14] “The method of claim 13, further comprising enabling the views of the Web page to be panned in real-time.”
[52] “The apparatus of claim 51, wherein execution of the instructions performs further operations comprising enabling the view of the Web content to be panned in real-time.”
[87] “The machine-readable medium of claim 86, wherein execution of the instructions performs further operations comprising enabling the view of the Web page to be panned in real-time.”
[114.c] “generating scaled Web page content in real-time.”
[115] “The method of claim 114, wherein the scaled Web page content is generated in a fraction of a second.”

162. In the Zaurus-Pad-SVG combination, Pad’s fast-pan re-rendering and adaptive render scheduling (applied in the hand-held browsing context of Zaurus and consistent with SVG’s performance expectations) enable the view to be panned in real-time and enable generation of scaled content in real-time as the user pans to new views. (EX1005, 11; EX1007, 142; EX1004, 43; EX1008, 6.)

163. Pad explains that it “normally does fast pans,” which involve “copying the portion of the screen that doesn’t change, and re-rendering the new portion,” so as the user pans to a new view, the system generates (and renders) the newly exposed content immediately as part of the pan update (“panned / generating in real-time”). (EX1005, 11.) Pad further teaches that this panning/zooming is kept interactive via adaptive rendering control: it uses “Adaptive render scheduling,” where the “system monitors render times, and adapts the rendering algorithm to maintain a constant

frame rate during zooms and pans” (“panned / generating in real-time”). (EX1007, 142.) Pad also discloses a configurable desired frame rate for animation (e.g., 20 frames/second), suggesting that view updates during panning occur fast enough to be perceived as real-time (“real-time”). (EX1005, 11.)

164. Zaurus suggests the user-interactive context for real-time panning on a hand-held device by disclosing touch-driven scrolling/panning behavior (e.g., scrolling is carried out by user touch interaction), which a person of ordinary skill in the art would have understood requires responsive, real-time feedback for usability during browsing (“real-time”). (EX1004, 43.) SVG further suggests that real-time performance is an expected requirement for interactive scalable Web graphics by listing performance-related requirements such as “Reasonable display performance” and related performance expectations for rendering during user pan/zoom (“real-time”). (EX1008, 6.)

165. For claim 115, Pad targets a specific frame rate, stating it maintains the “desired frame rate” and the “default is 20 frames/second.” (EX1005, 11.) A person of ordinary skill in the art would have understood that 20 frames per second requires generating each frame of scaled content in 0.05 seconds (“fraction of a second”). (See also EX1003, 344-346.)

17. Claims 15, 53, and 88

- i) [15] “The method of claim 1, further comprising enabling a user to zoom on a column of the Web**

content via an associated user input, wherein in response to the user input, the display is re-rendered such that content corresponding to the selected column is displayed across the display.”

[53] “The apparatus of claim 42, wherein execution of the instructions performs further operations comprising enabling a user to zoom on a column of the Web content via an associated user input, wherein in response to the user input, the view is re-rendered such that content corresponding to the selected column is displayed across the display.”

[88] “The machine-readable medium of claim 77, wherein execution of the instructions performs further operations comprising enabling a user to zoom on a column of the Web content via an associated user input, wherein in response to the user input, the view is re-rendered such that content corresponding to the selected column is displayed across the display.”

166. In the Zaurus-Pad-SVG combination, Pad states that the “HTML browser will lay out sub-documents in two columns next to the parent document,” and then instructs the user: “Simply zoom in on the sub-document to read it” (“enabling a user to zoom on a column of the Web content via an associated user input”). (EX1007, 91, 93.)

167. Pad states: “When you click on a piece of text, Pad++ zooms in so that that text fills the screen horizontally.” (EX1007, 54.) Consistent with this fit-to-width behavior, Pad’s center command changes the view so “the largest dimension of its bounding box fills the specified amount of screen,” which a person of ordinary skill in the art would have understood, when applied to a selected column (e.g., the sub-document column or its bounding box), re-renders the view so that column spans

the display width (“in response to the user input, the display / view is re-rendered such that content corresponding to the selected column is displayed across the display”). (EX1005, 22.)

168. Zaurus discloses “Touch[ing] the screen with a pen” (“user input”). (EX1004, 15.) A person of ordinary skill in the art would have been motivated to combine Pad’s column-based HTML layout and click-to-zoom fit-to-width behavior with Zaurus’s touch interface to facilitate reading multi-column Web layouts on a small display. (EX1007, 91, 54; EX1004, 15.)

18. Claims 16, 54, and 89

- i) [16] “The method of claim 15, wherein the user input comprises tapping the column.”**
[54] “The apparatus of claim 53, wherein the user input comprises tapping the column.”
[89] “The machine-readable medium of claim 88, wherein the user input comprises tapping the column.”

169. In the Zaurus-Pad-SVG combination, Zaurus discloses touch-based interaction on its screen to “Touch the screen with a pen” and further explains “Easy key touch ... Can be operated by a light touch,” which a person of ordinary skill in the art would have understood as a tap gesture on a touch-sensitive display (“tapping the column”). (EX1004, 15.)

170. Pad’s column-zoom interaction is triggered by a user “click” on displayed content: “When you click on a piece of text, Pad++ zooms in...”). In the

combination, that same user selection is performed on a touchscreen via a tap on the displayed column (or text within the column) (“user input comprises tapping the column”). (EX1007, 54; EX1004, 15.)

19. Claims 17, 55, and 90

- i) [17] “The method of claim 1, further comprising enabling a user to zoom on an image via an associated user input, wherein in response to the user input, the view is re-rendered such that the image is displayed to fit across a width of the display.”
[55] “The apparatus of claim 42, wherein execution of the instructions performs further operations comprising enabling a user to zoom on an image via an associated user input, wherein in response to the user input, the display is re-rendered such that the image is displayed to fit across a width of the display.”
[90] “The machine-readable medium of claim 77, wherein execution of the instructions performs further operations comprising enabling a user to zoom on an image, wherein in response to the user input, the view is re-rendered such that the image is displayed to fit across at least one of a width and height of the display.”

171. In the Zaurus-Pad-SVG combination, the user taps an on-screen image with the Zaurus pen input (“associated user input”). (EX1004, 15, 40.) Pad supports directing a user action to the specific object under interaction, so the tap is associated with the selected image item (“zoom on an image via an associated user input”). (EX1005, 18, 64.) In response to that input, Pad changes the view parameters and re-renders at the new view (“in response to the user input, the view / display is re-rendered”). (EX1005, 17, 22, 30, 32.)

172. The center command “change[s] the view” to center an item “so the largest dimension of its bounding box fills the specified amount of screen” (“displayed to fit across at least one of a width [and height] of the display”) because center makes that largest dimension fill the screen. (EX1005, 22.) To implement width-specific fitting when needed, Pad also computes the image’s width from its bounding box and set the view’s zoom accordingly (“displayed to fit across a width of the display”). (EX1005, 17, 30, 32.)

173. Pad demonstrates user-driven zooming on images, stating that after viewing “pictures,” “you could zoom in further to see the images in greater detail” (“enabling a user to zoom on an image”). (EX1007, 80.) Zaurus confirms the web-image context, including GIF and JPEG support, so the tapped object can be a displayed web image. (EX1004, 40.) SVG further suggests that images scale with zoom under “true scalability,” consistent with the system re-rendering at a new scale so the image fits the display. (EX1008, 11, 14.)

20. Claims 18, 56, and 91

- i) **[18] “The method of claim 17, wherein the user input comprises tapping the image.”**
[56] “The apparatus of claim 55, wherein the user input comprises tapping the image.”
[91] “The machine-readable medium of claim 90, wherein the user input comprises tapping the image.”

174. In my opinion, in the Zaurus-Pad-SVG combination, the “tapping” input is satisfied for the same reasons as claim 3 (touch display) and claim 17 (zoom-

on-image interaction), with Pad's click selection implemented as a tap on Zaurus's touch screen. (See Sections IX.A.5, 19; EX1004, 15; EX1005, 47, 64; EX1007, 54, 80.)

175. Zaurus teaches touch-based user input on the device display, including "touching the screen with a pen" and "Easy key touch... Can be operated by a light touch," which corresponds to a tap gesture on the image ("tapping the image"). (EX1004, 15.)

176. Pad teaches that users zoom into images via user interaction, explaining in the file-browser example that the user can "zoom in further to see the images in greater detail," and Pad further treats images as "Items of type image." (EX1007, 80; EX1005, 64.) Pad also describes click-driven zooming behavior (e.g., clicking a displayed item to zoom), and Pad's supports applying user-selection interactions to item objects on the surface, including image items. (EX1007, 54; EX1005, 47, 64.)

177. Accordingly, a person of ordinary skill in the art would implement Pad's click-to-zoom interaction on the Zaurus touch display such that the user input is a tap on the image. (EX1004, 15; EX1007, 54, 80; EX1005, 47, 64.)

21. Claims 19, 57, and 92

- i) [19] "The method of claim 1, further comprising enabling a user to zoom on a paragraph of the Web content via an associated user input, wherein in response to the user input, the view is re-rendered"**

such that content corresponding to the selected paragraph is displayed to fit across the display.”

[57] “The apparatus of claim 42, wherein execution of the instructions performs further operations comprising enabling a user to zoom on a paragraph of the Web content via an associated user input, wherein in response to the user input, the view is re-rendered such that content corresponding to the selected paragraph is displayed across the display.”

[92] “The machine-readable medium of claim 77, wherein execution of the instructions performs further operations comprising enabling a user to zoom on a paragraph of the Web content via an associated user input, wherein in response to the user input, the view is re-rendered such that content corresponding to the selected paragraph is displayed across the display.”

178. In the Zaurus-Pad-SVG combination, Pad teaches user-input zooming on a selected block of text and re-rendering to fit the selected text across the screen: “When you click on a piece of text, Pad++ zooms in so that that text fills the screen horizontally” (“zoom on a paragraph ... via an associated user input”; “in response ... the view is re-rendered”; “displayed ... across the display”). (EX1007, 54.) A person of ordinary skill in the art would have understood that a “piece of text” in a Web context corresponds to a “paragraph” (e.g., an HTML <p> tag rendered as a text object). (EX1042, 1-3; EX1043, 2, 10.)

179. Pad further suggests the fit-to-width re-rendering mechanism by teaching center, which changes the view so “the largest dimension of its bounding box fills the specified amount of screen,” meaning that when applied to the selected

text object corresponding to the paragraph, the view is regenerated so that paragraph content is enlarged to fit across the display (“re-rendered ... displayed to fit across the display”). (EX1005, 22.) Pad’s also supports treating rendered text as discrete text objects (“text items”) created during HTML parsing, providing the selectable paragraph text object on which the zoom/re-render operates. (EX1005, 60, 75.)

180. Zaurus supplies the handheld/touch input context for the “associated user input” on a mobile display, teaching interaction by “touching the screen with a pen,” which a person of ordinary skill in the art would implement as the selection input to trigger Pad’s text-zoom behavior on a paragraph of HTML-rendered text. (EX1004, 15.)

22. Claims 20, 58, and 93

- i) [20] “The method of claim 19, wherein the user input comprises tapping the paragraph.”
[58] “The apparatus of claim 57, wherein the user input comprises tapping the paragraph.”
[93] “The machine-readable medium of claim 92, wherein the user input comprises tapping the paragraph.”

181. In the Zaurus-Pad-SVG combination, the user input used to select the paragraph for the paragraph-zoom operation is a tap on the touch-sensitive display (“wherein the user input comprises tapping the paragraph”). (EX1004, 15.)

182. Zaurus teaches tap-like touch input on the display, instructing the user to “Touch the screen with a pen” and explaining “Easy key touch” that “Can be

operated by a light touch,” supporting that the device receives user inputs via touch/tapping on displayed content. (EX1004, 15.)

183. Pad teaches the corresponding selection gesture for triggering the zoom on a block of text: “When you click on a piece of text, Pad++ zooms in ...,” and in the Zaurus touch-display, a person of ordinary skill in the art would implement that selection input as a tap on the paragraph text (“tapping the paragraph”). (EX1007, 54.)

23. Claims 21, 59, and 94

- i) **[21] “The method of claim 19, wherein the content of the paragraph is reformatted when the view is re-rendered.”**
[59] “The apparatus of claim 57, wherein the content of the paragraph is reformatted when the view is re-rendered.”
[94] “The machine-readable medium of claim 92, wherein the content of the paragraph is reformatted when the view is re-rendered.”

184. In the Zaurus-Pad-SVG combination, when the user zooms on a paragraph and the view is re-rendered to fit that paragraph across the display (as in claim 19), the paragraph’s text is reformatted (e.g., reflowed/rewrapped) as part of the re-render (“the content of the paragraph is reformatted when the view is re-rendered”). (EX1005, 22, 63.)

185. Pad explains that an HTML item “specifies the width ... of the html page” and that “[t]he page will be re-laid out according to the new width,” which is

reformatting of the text content (including paragraph wrapping/line breaks) when the page/view is re-rendered under a new width. (EX1005, 63.) A person of ordinary skill in the art would have understood that HTML paragraphs are designed to be “reformatted” (reflowed/wrapped) based on their width. (EX1043, 2, 10.) Applied to a paragraph text object, this changes the effective width used for rendering and therefore causes the HTML/text layout to be recomputed under the new width, resulting in paragraph reformatting upon re-render. (EX1005, 22, 63.) Pad further describes “Semantic zooming” in which objects can appear differently at different sizes, consistent with re-render-time changes in how text content is presented as the view scale changes. (EX1007, 85.)

24. Claims 22, 60, and 95

- i) [22.a] “The method of claim 1, further comprising:”
[60.a] “The apparatus of claim 42, wherein execution of the instructions performs further operations comprising:”
[95.a] “The machine-readable medium of claim 77, wherein execution of the instructions performs further operations comprising:”
[22.b] “generating a display list derived, at least in part, via use of the scalable page layout information; and”
[60.b] “generating a display list derived, at least in part, via use of the scalable page layout information; and”
[95.b] “generating a display list; and”

186. In the Zaurus-Pad-SVG combination, the renderer maintains an internal list/tree of drawing items (a “display list”) including object/layout information used

to render at different scales. A person of ordinary skill in the art would have understood that a “display list” is a well-known, conventional computer graphics term for the internal list of commands or objects used to define the output. (EX1052, 195-196; *see also* EX1001, 19:10-22.) Pad discloses this “display list” during rendering (“Group members are rendered sequentially in the display list”) and confirms it is an internal hierarchical item tree (via “printtree” printing the “internal tree structure”). (EX1005, 6, 34, 59.) Pad further suggests the list structure by providing commands that reorder items “in the display list,” confirming that objects/items occupy positions in an internal ordered list used by the renderer. (EX1005, 32.) Generating that structure uses scalable layout information because items are created/placed using coordinates and size/position parameters (e.g., create uses “coordinates,” and “-place” sets “position and item size”) (“derived ... via use of the scalable page layout information”). (EX1005, 24; 50.) SVG’s “implicit layering” (objects drawn bottom-to-top in input order) further discloses the same display-list/scene-graph rendering paradigm motivating this architecture in the combination. (EX1008, 9.)

- ii) [22.c] “employing the display list to generate views of the Web page.”
[60.c] “employing the display list to generate views of the Web page.”
[95.c] “employing the display list to generate views of the Web page.”

187. In the Zaurus-Pad-SVG combination, the rendering engine handles the internal list/tree of graphical objects and their attributes (“employing the display list”) by traversing it to render the page content to the screen (“generating views of the Web page”). Pad ties rendering to use of the display list: “Group members are rendered sequentially in the display list,” and operations such as raising/lowering a group act on all group members within that display-list structure. (EX1005, 59.) Pad further explains that it supports view changes such as fast panning by “re-rendering the new portion” of the scene based on the underlying object data, which would be done by re-traversing the display list/scene graph to generate the updated view. (EX1005, 11.) Consistent with standard graphics principles, a person of ordinary skill in the art would have understood that the purpose of a display list/scene graph is to enable rendering by generating views of the content by traversing the stored drawing objects/commands. (EX1052, 195-196.)

25. Claims 23, 61, and 96

- i) [23.a] “The method of claim 1, further comprising:”
[61.a] “The apparatus of claim 42, wherein execution of the instructions performs further operations comprising:”
[96.a] “The machine-readable medium of claim 77, wherein execution of the instructions performs further operations comprising:”
[23.b] “parsing and processing, with the rendering engine, markup language code corresponding to the received Web content to determine page layout information corresponding to an original page layout of the content on the Web page as interpreted by the

rendering engine, the content including a plurality of objects comprising text objects associated with the text content and image objects associated with the plurality of images, wherein for each object the layout information includes object layout information identifying a coordinate location of the object on the page layout; and”

[61.b] “parsing and processing, with the rendering engine, markup language code corresponding to the received Web content to determine page layout information corresponding to an original page layout of the content on the Web page as interpreted by the rendering engine, the content including a plurality of objects comprising text objects associated with the text content and image objects associated with the plurality of images, wherein for each object the layout information includes object layout information identifying a coordinate location of the object on the page layout; and”

[96.b] “parsing and processing, with the rendering engine, markup language code corresponding to the received Web content to determine page layout information corresponding to an original page layout of the content on the Web page as interpreted by the rendering engine, the content including a plurality of objects comprising text objects associated with the text content and image objects associated with the plurality of images, wherein for each object the layout information includes object layout information identifying a coordinate location of the object on the page layout; and”

188. In the Zaurus-Pad-SVG combination, Zaurus teaches a handheld Internet browser that incorporates “HTML 3.2” and can display images on a page (GIF and JPEG), evidencing processing of HTML markup to render Web content that includes text and images. (EX1004, 40.) Pad likewise teaches HTML parsing

and rendering: “When the html data is fetched, it is parsed and the HTML item is created which contains a method for rendering the page” (“parsing and processing, with the rendering engine, markup language code”). (EX1005, 60.)

189. Pad explains that the HTML page layout is computed and can be recomputed based on layout constraints: “The page will be re-laid out according to the new width, and the length of the page may change dependent on the new width” (“page layout information corresponding to an original page layout of the content on the Web page as interpreted by the rendering engine”). (EX1005, 63.) A person of ordinary skill in the art would have understood that the rendering engine interprets the markup to derive the layout information for the page. (EX1005, 63.)

190. Pad explains that parsed HTML yields multiple discrete items: “HTML anchors are created as separate items.” (EX1005, 60.) Pad also defines “Text Items” (a text item displays “a string of characters”) and “Image Items” (items of type image appear as “color images”) (“content including a plurality of objects comprising text objects associated with the text content and image objects associated with the plurality of images”). (EX1005, 64, 75.)

191. Pad discloses that “Every object has a -place” which specifies the anchor point, and “The place specifies the object’s position and size,” which provides coordinate location information for each object (“for each object the layout information includes object layout information identifying a coordinate location of

the object on the page layout”). (EX1005, 50.) Pad also teaches that “bbox” returns an item’s bounding box “in the form ‘x1 y1 x2 y2’,” which identifies the object’s location and extent in coordinates on the page layout. (EX1005, 17.) SVG further suggests coordinate-based layout by specifying “Coordinate systems” and positioning concepts, consistent with representing page objects using coordinates. (EX1008, 11.)

192. A person of ordinary skill in the art would have been motivated to combine Zaurus’s HTML browser context with Pad’s parsed, object-based rendering and coordinate-based item placement to obtain the claimed “page layout information” for “a plurality of objects” (text and images) each having “a coordinate location,” consistent with SVG’s coordinate system expectations.

- ii) **[23.c] “associating each object and its object layout information.”**
[61.c] “associating each object and its object layout information.”
[96.c] “associating each object and its object layout information.”

193. In the Zaurus-Pad-SVG combination, the rendering engine maintains, for each rendered item (such as a text item or image item), an object identifier and the object’s stored layout parameters (position, size, bounds) that can be retrieved for that specific object. Pad discloses that when an item is created, the system assigns it an identifier because the create command “returns the id for the new item.” (EX1005, 24.) Pad further shows that the system tracks item-specific data associated

to that item because it refers to “item: The id of the item being configured,” and its “itemconfigure” command “modifies item-specific options for the items given by tagOrId” (“associating each object and its object layout information”). (EX1005, 16, 32.) In combination, the object (identified by id or tagOrId) is linked to its position/size attributes and bounds coordinates within the rendering engine’s object model. (EX1005, 16, 17, 24, 32, 46, 50, 59.)

26. Claims 24, 62, and 97

- i) [24] **“The method of claim 23 further comprising: generating page layout information including an object bounding box for each object.”**
[62] **“The apparatus of claim 61 wherein execution of the instructions performs further operations comprising: generating page layout information including an object bounding box for each object, the object bounding box representing a portion of the page layout occupied by the object’s associated content.”**
[97] **“The machine-readable medium of claim 96 wherein execution of the instructions performs further operations comprising: generating page layout information including an object bounding box for each object, the object bounding box representing a portion of the page layout occupied by the object’s associated content.”**

194. In the Zaurus-Pad-SVG combination, the renderer supports “generating page layout information including an object bounding box for each object” by generating, for, a bounding region that encloses the content drawn for that item. Pad’s “bbox” command returns an item’s bounding box as “x1 y1 x2 y2” and

specifies that “the drawn areas” of the named items are within the region bounded by those coordinates (“object bounding box representing a portion of the page layout occupied by the object’s associated content”). (EX1005, 17.) Pad provides bounding box generation for text content through “gettextbbox,” which returns the text’s bounding coordinates (“x1 y1 x2 y2”) and states the text lies within that bounded region. (EX1005, 30.) For each item (text, image, or other object), Pad computes a bounding box for the item via the “-bb” option, which evaluates a script “to compute the bounding box of this item” and returns lower-left and upper-right corners (“generating page layout information including an object bounding box for each object”). (EX1005, 65, 74-75.) Pad further suggests that the system uses these per-object bounding boxes as the occupied region on the page layout because the damage command can redraw “just the bounding box of each of the objects specified,” reflecting that each object has a bounding box the renderer relies on for updates. (EX1005, 24.)

27. Claims 25, 63, and 98

- i) **[25] “The method of claim 23, further comprising: for each of the plurality of objects, mapping the coordinate location associated with the object layout information for the object to a corresponding object datum coordinate on a virtual display area employing a resolution-independent coordinate system.”**
[63] “The apparatus of claim 61, wherein execution of the instructions performs further operations comprising: for each of the plurality of objects, mapping the coordinate location associated with the

object layout information for the object to a corresponding object datum coordinate on a virtual display area employing a resolution-independent coordinate system.”

[98] “The machine-readable medium of claim 96, wherein execution of the instructions performs further operations comprising: for each of the plurality of objects, mapping the coordinate location associated with the object layout information for the object to a corresponding object datum coordinate on a virtual display area employing a resolution-independent coordinate system.”

195. In the Zaurus-Pad-SVG combination, the rendering engine can “for each of the plurality of objects” map the object’s “coordinate location ... to a corresponding object datum coordinate on a virtual display area employing a resolution-independent coordinate system” by storing each object’s anchor position in Pad++ “pad” coordinates on the Pad++ surface (and converting between window and pad coordinates as needed).

196. Pad explains “all of the data is placed directly on the Pad++ surface, and you navigate through the data by panning and zooming,” in a “zoomable coordinate system” where objects have an “absolute physical location.” (EX1007, 73.) Pad explains that this “surface” coordinate space is distinct from the visible device window: for “bbox,” if “the item is the Pad++ surface ... then bbox will refer to the bounding box of the portion of the surface that is currently visible (based on the view and window size)” (“virtual display area”). (EX1005, 17.)

197. Pad defines a per-object datum (an anchor coordinate) via “-place,” stating: “Every object has a -place which specifies the anchor point of that object ... The place specifies the object’s position and size” (“mapping the coordinate location ... to a corresponding object datum coordinate”). (EX1005, 50-51.) Pad also provides a coordinate mapping operation between the device/window and the surface: “padxy ... Given a window x-coordinate ... and y-coordinate ... returns the pad x-coordinate and y-coordinate that is displayed at that location” (“mapping the coordinate location”). (EX1005, 33.)

198. Pad supports resolution-independent coordinate handling by interpreting coordinates relative to coordinate frames rather than fixed pixel positions: “pushcoordframe ... When any coordinate frames are on the stack, all coordinates are interpreted relative to the frame ... all coordinates are specified within the unit square where the unit square is mapped to the frame” (“resolution-independent coordinate system”). (EX1005, 34.) Pad further notes that size “may also be specified as a percentage of the view,” which implies resolution-independent specification tied to view geometry rather than a pixel grid. (EX1005, 48.) SVG reinforces that scalable web content uses resolution-independent coordinates, requiring support for “local user coordinates” and “real number values ... for all appropriate attributes and coordinates” (“resolution-independent coordinate system”). (EX1008, 11-12.)

28. Claims 26, 64, and 99

- i) [26.a] “The method of claim 25, further comprising:”
[64.a] “The apparatus of claim 63, wherein execution of the instructions performs further operations comprising:”
[99.a] “The machine-readable medium of claim 98, wherein execution of the instructions performs further operations comprising:”
[26.b] “determining a first scale factor and offset in response to one or more corresponding user inputs defining a user-selectable zoom level and pan position corresponding to a view of the Web content desired by a user;”
[64.b] “determining a first scale factor and offset in response to one or more corresponding user inputs defining a user-selectable zoom level and pan position corresponding to a view of the Web content desired by a user;”
[99.b] “determining a first scale factor and offset in response to one or more corresponding user inputs defining a user-selectable zoom level and pan position corresponding to a view of the Web content desired by a user;”

199. In the Zaurus-Pad-SVG combination, Pad states that “clicking and holding on the middle mouse button performs an smooth animated zoom in” (“in response to one or more ... user inputs defining a user-selectable zoom level”). (EX1007, 4.) Pad also explains that “at any time you can use the mouse to pan and zoom to another part of the presentation” (“user-selectable ... pan position”) and a user-selected (“view ... desired by a user”). (EX1007, 71.) As an example, Pad notes “Clicking on the ‘there lived a king’ line performs an animated pan/zoom” (“in response to ... user inputs”). (EX1007, 113.)

200. As I discussed for Claims 8 and 9, Pad implements user-driven zoom and pan through view parameters defined as “xview yview zoom,” where “zoom specifies the magnification” (“scale factor”) and the view center (xview, yview) defines the pan position (“offset”). (See Sections IX.A.10, 11; EX1005, 30, 32.) Pad’s “moveto” then “Change[s] the view so that the point ‘xview yview’ is at the center of the screen with a magnification of zoom” to achieve the pan position and zoom level selected through the user’s inputs (“determining a first scale factor and offset”). (EX1005, 32.)

- ii) **[26.c] “determining a virtual display bounding box for the virtual display area associated with the first scale factor and offset;”**
[64.c] “determining a virtual display bounding box for the virtual display area associated with the first scale factor and offset;”
[99.c] “determining a virtual display bounding box for the virtual display area associated with the first scale factor and offset;”

201. In the Zaurus-Pad-SVG combination, the “virtual display bounding box” corresponds to the bounding box of the visible portion of the Pad surface for the selected view/window size, as I discussed for Claim 24. (See Section IX.A.26; EX1005, 17.) Pad determines the bounding box of the portion of the surface that is currently visible, explaining that when “the item is the Pad++ surface (item #1), then bbox will refer to the bounding box of the portion of the surface that is currently visible (based on the view and window size)” (“determining a virtual display

bounding box for the virtual display area associated with the first scale factor and offset”). (EX1005, 17.)

- iii) [26.d] “identifying objects having at least a portion falling within the virtual display bounding box; and”
[64.d] “identifying objects having at least a portion falling within the virtual display bounding box; and”
[99.d] “identifying objects having at least a portion falling within the virtual display bounding box; and”

202. In the Zaurus-Pad-SVG combination, as I discussed for Claim 8, Pad identifies objects at least partly within the visible region (e.g., via overlapping/visible-object determination, including its R-tree mechanism). (See Section IX.A.13; EX1005, 27; EX1007, 141.) Pad identifies items within a rectangular region using “overlapping x1 y1 x2 y2,” which “Returns all the items that overlap or are enclosed within the rectangular region given by x1, y1, x2, and y2” (“identifying objects having at least a portion falling within the virtual display bounding box”). (EX1005, 27.)

- iv) [26.e] “rendering a view of the Web page corresponding to the zoom level and pan position on the display including the at least a portion of the objects falling within the virtual display bounding box.”
[64.e] “rendering a view of the Web page corresponding to the zoom level and pan position on the display including the at least a portion of the objects falling within the virtual display bounding box.”
[99.e] “applying an applicable scale factor to the object’s content to render a view corresponding to the zoom level and pan position on the display.”

203. In the Zaurus-Pad-SVG combination, rendering the view corresponding to the selected zoom level and pan position follows from Pad's view-change and re-rendering mechanics I discussed for Claims 8, 9, and 14, where updated view parameters (zoom and center coordinates) are applied to the underlying layout representation to render the portion of objects falling within the visible region. (See Sections IX.A.11, 12, 16; EX1005, 11, 30-32; EX1007, 108.)

204. Pad renders the updated display based on the changed view during pan operations, explaining that it “does fast pans, i.e., copying the portion of the screen that doesn't change, and re-rendering the new portion” (“render[ing] a view [of the Web page] corresponding to the zoom level and pan position on the display [including the at least a portion of the objects falling within the virtual display bounding box]”). (EX1005, 11.) In the Pad HTML browser context, Pad explains that “some words are shown in blue” as “hotwords,” and “[w]hen you click on the word, the appropriate document is placed on the Pad++ surface” (“view of the Web page”). (EX1007, 108.) Pad's view includes “zoom” where “zoom specifies the magnification,” and “moveto” changes the view to use that zoom (“applying an applicable scale factor to the object's content”). (EX1005, 30, 32.)

29. Claims 27, 65, and 100

- i) [27] “The method of claim 25, wherein the virtual display area is associated with a fixed reference frame.”**

[65] “The apparatus of claim 63, wherein the virtual display area is associated with a fixed reference frame.”

[100] “The machine-readable medium of claim 98, wherein the virtual display area is associated with a fixed reference frame.”

205. In the Zaurus-Pad-SVG combination, Pad explains that “objects have an absolute physical location” and that “all of the data is placed directly on the Pad++ surface,” and the user “navigate[s] through the data by panning and zooming” (“virtual display area is associated with a fixed reference frame”). (EX1007, 73.) Pad defines the view relative to that surface in “xview yview zoom” form, where “(xview,yview) specifies the point at the center of the window, and zoom specifies the magnification,” so panning and zooming adjust the view over the same underlying coordinate space (“fixed reference frame”). (EX1005, 30.) Pad further confirms that object coordinates are expressed in a stable coordinate space by returning coordinates in either the item’s “local coordinate system (i.e., as they were originally specified)” or in the “global coordinate system,” where coordinates “are transformed by that item’s translation and scale parameters” (“fixed reference frame”). (EX1005, 23.) As additional confirmation, Pad’s “bbox” supports returning geometry “independent of the current view...as if the view were ‘0 0 1.’” (EX1005, 23.)

30. Claims 28, 66, and 101

- i) **[28.a] “The method of claim 1, wherein the content includes a plurality of text objects having associated text content, the method further comprising:”**
[66.a] “The apparatus of claim 42, wherein the content includes a plurality of text objects having associated text content, and wherein execution of the instructions performs further operations comprising:”
[101.a] “The machine-readable medium of claim 77, wherein the content includes a plurality of text objects having associated text content, and wherein execution of the instructions performs further operations comprising:”

206. In the Zaurus-Pad-SVG combination, Pad provides “text” and “textfile” item types that each display a text string, and each such item stores that string as the item’s content. Pad states that a text item “displays a string of characters on the screen,” and that for text items, “String specifies the characters to be displayed in the text item,” which corresponds to “text objects” with “associated text content.” (EX1005, 75, 78.) A person of ordinary skill in the art would have understood that rendering a document or page using multiple such text items yields “a plurality of text objects” each linked to its respective string content (“wherein the content includes a plurality of text objects having associated text content”).

- ii) **[28.b] “for at least a portion of the plurality of text objects, scaling a scalable font to render the text content associated with the text object.”**
[66.b] “for at least a portion of the plurality of text objects, scaling a scalable font to render the text content associated with the text object.”

[101.b] “for at least a portion of the plurality of text objects, scaling a scalable font to render the text content associated with the text object.”

207. As I discussed for limitation [10.c], Pad renders text using scalable Type 1 fonts and scales text rendering as a function of zoom/magnification and/or font height, consistent with SVG’s “true scalability.” (EX1005, 12, 30, 62, 75, 78; EX1008, 14; Section IX.A.12.ii.)

208. In the Zaurus-Pad-SVG combination, Pad’s per-item “-font” option (available for text-related item types) requires “an Adobe Type 1 font” (“scaling a scalable font”). (EX1005, 62.) Pad explains that when drawing text in Type 1 fonts it may “generat[e] the bitmap for the character from its outline description,” and it can cache and reuse the bitmap for the same “size” and “font” (“scaling a scalable font to render the text content associated with the text object”). (EX1005, 12.) Pad explains: “Text and images are scaled” via the coordinate-frame and item placement mechanisms, and text items note that their “size can be changed with the scale command or the -place itemconfigure option” (“for at least a portion of the plurality of text objects”). (EX1005, 35, 75.) Pad also represents the view with “zoom” where “zoom specifies the magnification,” which further supports scaling rendered text as zoom changes. (EX1005, 30.) SVG suggests this expected zoom behavior, stating that with true scalability “text ... should grow/shrink uniformly with the zoom level,” consistent with scaling fonts when rendering zoomed views. (EX1008, 10,

14.) SVG is designed to replace raster graphics with a vector format that supports “text data” with “precise control of text sizing [and] positioning... so that graphic artists have a way to ensure that the end user sees what was intended.” (EX1008, 1, 7-8.)

31. Claims 29, 67, and 102

- i) [29.a] “The method of claim 1, wherein the content includes a plurality of text objects having associated text content, the method further comprising:”
[67.a] “The apparatus of claim 42, wherein the content includes a plurality of text objects having associated text content, and wherein execution of the instructions performs further operations comprising:”
[102.a] “The machine-readable medium of claim 77, wherein the content includes a plurality of text objects having associated text content, and wherein execution of the instructions performs further operations comprising:”**

209. The Zaurus-Pad-SVG combination discloses and/or suggests this limitation for the same reasons as I discussed for limitation [28.a]. (Section IX.A.30.i.)

- ii) [29.b] “for at least a portion of the plurality of text objects, determining an applicable font size based on an applicable scale factor associated with a zoom level; and”
[67.b] “for at least a portion of the plurality of text objects, determining an applicable font size based on an applicable scale factor associated with a zoom level; and”
[102.b] “for at least a portion of the plurality of text objects, determining an applicable font size based on**

**an applicable scale factor associated with a zoom level;
and”**

210. In the Zaurus-Pad-SVG combination, Pad provides a per-item “-font” option “available only for HTML, portal, text and textfile item types,” which “Specifies the font to be used for rendering text for this item” (“for at least a portion of the plurality of text objects”). (EX1005, 62, 78, 81.) Pad also supports user-selectable zoom: “clicking and holding on the middle mouse button performs an smooth animated zoom in” (EX1007, 4), and Pad defines the view in “xview yview zoom” form where “zoom specifies the magnification” (“an applicable scale factor associated with a zoom level”) (EX1005, 30). Pad determines font size for rendering with “setfontheight,” where “[h]eight is specified in the current pad units” (“determining an applicable font size based on an applicable scale factor associated with a zoom level”). (EX1005, 30, 37, 81.) Pad also provides “getmag,” which returns the current “magnification” for a specific render and defines magnification in terms of the current view combined with the object’s size parameters. (EX1005, 28-29.)

- (1) [29.c] “selecting a corresponding bitmapped font and employing the bitmapped font to render the text content associated with the text object.”
[67.c] “selecting a corresponding bitmapped font and employing the bitmapped font to render the text content associated with the text object.”
[102.c] “selecting a corresponding bitmapped font and employing the bitmapped font to render the text content associated with the text object.”

211. In the Zaurus-Pad-SVG combination, Pad explains that it employs a caching mechanism when drawing text in Type 1 fonts that “remembers what size, font and bitmap it used when it last drew a particular character,” and when the character is drawn again at the same size and font, Pad “reuses the last bitmap image for that character” rather than regenerating it from the outline (“selecting a corresponding bitmapped font and employing the bitmapped font to render the text content associated with the text object”). (EX1005, 12.)

32. Claims 30 and 68

- i) **[30] “The method of claim 1, further comprising reformatting a text object so as to fit the text object on the display when it is rendered.”**
[68] “The apparatus of claim 42, wherein execution of the instructions performs further operations comprising reformatting a text object so as to fit the text object on the display when it is rendered.”

212. In the Zaurus-Pad-SVG combination, Pad provides a “-width” option for HTML pages, explaining that it “Specifies the width (in the current units) of the html page,” and that “[t]he page will be re-laid out according to the new width, and the length of the page may change dependent on the new width” so the page fits within the target display width when rendered (“reformatting a text object so as to fit the text object on the display when it is rendered”). (EX1005, 63.)

33. Claims 32, 70, and 106

- i) **[32] “The method of claim 1, wherein in a view at least one image is scaled.”**

[70] “The apparatus of claim 42, wherein in a generated view at least one image from⁴ is scaled.”

[106] “The machine-readable medium of claim 77, wherein⁵ at least one view at least one image is scaled.”

213. In the Zaurus-Pad-SVG combination, Pad discloses “image” items (i.e., objects that render image content) by providing that “Items of type image appear on the display as color images,” created via a “create image” command and configured with an “-image” option that specifies the “imagetoken” the item will render. (EX1005, 64-65.) Pad further discloses that images are scaled in the displayed view by explaining that, when using Pad’s coordinate frames and zoomable placement, “Text and images are scaled” such that “the height of the image is scaled to the height of the coordinate frame ... (using the -place or -z itemconfigure options).” (EX1005, 35.) Pad likewise states that after viewing “pictures,” the user can “zoom in further to see the images in greater detail” (“wherein in a view at least one image is scaled”) (EX1007, 80).

34. Claim 33: “The method of claim 1, further comprising maintaining at least one instance of the page layout information in a manner that is independent of zoom levels used to view the web page.”

⁴ In view of claims 32 and 106, “from” appears to be a stray extra word.

⁵ In view of claims 32 and 70, the word “in” appears to be omitted.

214. In the Zaurus-Pad-SVG combination, Pad discloses that “objects have an absolute physical location,” and that “all of the data is placed directly on the Pad++ surface,” while the user “navigate[s] through the data by panning and zooming” (“maintaining at least one instance of the page layout information in a manner that is independent of zoom levels”). (EX1007, 73.) Pad also represents the current view as “xview yview zoom,” where “zoom specifies the magnification,” a person of ordinary skill in the art would have understood that changing “zoom” changes only the view transform applied during rendering, leaving the maintained layout instance unchanged across zoom levels. (EX1005, 30.) Pad further provides a view-independent form of object geometry by explaining that when “-sticky is specified, then the bounding box returned is independent of the current view (i.e., it returns the bounding box as if the view was ‘0 0 1’),” reflecting retention of at least one instance of layout information that does not change with zoom. (EX1005, 17.) Pad further suggests view-independent object coordinates as a “local coordinate system” as originally specified or in the “global coordinate system” after applying the item’s translation and scale parameters. (EX1005, 23.)

35. Claim 34: “The method of claim 1, wherein the scalable page layout information comprises scalable vector-based page layout information.”

215. In the Zaurus-Pad-SVG combination, SVG is a “Scalable Vector Graphics” model, and it describes applying transformations uniformly to “shapes,

images and text,” including “true scalability” where “linewidths, images and text should scale along with the sizes of the shapes.” (EX1008, 1, 5, 11.) Pad likewise represents page objects on a surface in a coordinate space and “Text and images are scaled” using view magnification and scaling transformations (e.g., “zoom specifies the magnification” and the “scale” command multiplies item size by a scale amount). (EX1005, 30, 35, 36; EX1007, 73.) Accordingly, the combination’s scalable layout representation is vector-based, such that “the scalable page layout information comprises scalable vector-based page layout information.” In particular, the layout information includes vectors in the form of (X, Y) translation offsets (e.g., SVG transform translation terms) from a page/frame reference point to each object’s anchor/reference point, satisfying the requirement that the scalable vector-based page layout information “includes one or more vectors.” (EX1008, 11; EX1005, 50.)

36. Claim 35

- i) **[35.a] “The method of claim 34, wherein scaling operations are effected by applying a mathematical transformation to a plurality of points in a two-dimensional coordinate system comprising X and Y axes,”**

216. In the Zaurus-Pad-SVG combination, SVG supports “scaling operations are effected by applying a mathematical transformation to a plurality of points in a two-dimensional coordinate system comprising X and Y axes” because

it requires geometric transformations expressed as matrices and ties those transformations to a coordinate system with numeric coordinate values. SVG states: “Arbitrarily nested 2x3 matrix transformations should be available,” and “2x3 matrices are sufficient to achieve any types of translation, scaling, rotation and skewing.” (EX1008, 11.) SVG also specifies “local user coordinates” and real-number values for “attributes and coordinates,” reflecting that graphics content is represented using 2D coordinate values that are transformed by the matrix operations. (EX1008, 11.) A person of ordinary skill in the art would have understood that applying a 2x3 transform matrix to 2D user coordinates effects scaling by transforming the (X, Y) coordinate pairs of multiple points that define objects.

- ii) **[35.b] “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,”**

217. In the Zaurus-Pad-SVG combination, SVG’s vector-based layout information includes coordinate-defined geometry and applies the same coordinate framework to shapes, text, and images. SVG supports “local user coordinates” and arbitrarily nested “2x3 matrix transformations” (including translation), and a person of ordinary skill in the art would understand that the translation terms provide an (X, Y) offset vector from a page/frame reference point (“primary datum”) to an object’s

reference point (“object datum”) (i.e., a vector represented by (X2, Y2) relative to an origin point (X1, Y1), and thus a mathematical expression representing a length and a direction in the two-dimensional coordinate system), thereby defining page-layout locations for text and image objects within the two-dimensional coordinate system. (EX1008, 11.) Pad likewise defines per-object anchor points (“-place”) and object bounds, providing the corresponding object datums and coordinate locations to which those (X, Y) offset vectors apply in the combined system. (EX1005, 50; EX1005, 17.)

218. SVG also defines vector graphics shapes using “path data” consisting of commands such as “Straight line segments” and “Cubic bezier curved segments,” which are specified by coordinate points that define the geometry of the shape. (EX1008, 7.) A person of ordinary skill in the art would have understood that these coordinate-based segments consist of the claimed “datum points having corresponding vectors” because such segments are defined by datum points (e.g., start, end, and control points) and the corresponding vectors between those points; vectors represented by (X2, Y2) relative to an origin point (X1, Y1) that encode the length and direction from a first datum point to a second datum point. Standard industry references further confirm that cubic Bézier segments use control points that determine curve direction at the endpoints (e.g., tangency relative to the line segment between specified points), consistent with those point-to-point vectors

representing direction and length in the two-dimensional coordinate system. (EX1053, 578-579.) By defining these geometric relationships through start, end, and control points, the SVG “path data” discloses the “datum points and corresponding vectors.” SVG further discloses that the transformations “should also be available for text and image objects,” placing text and image objects into the same point-based 2D coordinate system (“defining page layout locations of corresponding text and image objects mapped to the two-dimensional coordinate system”). (EX1008, 6, 11.)

219. Pad suggests that “objects have an absolute physical location” on the surface while the user navigates by “panning and zooming.” (EX1007, 73.) Pad represents text and image items as objects placed at X/Y locations on the surface and then scales those objects via view magnification and scaling operations. (EX1005, 30, 36.) Pad also assigns each object an anchor point through “-place,” where “The place specifies the object’s position and size,” consistent with page-layout locations for text and image objects. (EX1005, 50.) Pad includes both “image” items and “text” items, which are consistent with mapping text and image objects to 2D page-layout locations. (EX1005, 65, 78.) A person of ordinary skill in the art would view SVG’s vector point data and Pad’s object placement as complementary in the combined page layout representation.

- iii) **[35.c] “wherein the mathematical transformation comprises, $X'=XSF$; $Y'=YSF$; 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.”**

220. In the Zaurus-Pad-SVG combination, SVG requires support for “2x3 matrix transformations” for objects to “scale uniformly,” and a person of ordinary skill in the art would have understood that uniform 2D scaling involves applying SF to both coordinate components (“ $X'=XSF$; $Y'=YSF$... SF is the scale factor”). (EX1008, 11.) A person of ordinary skill in the art would have understood that the transform maps an input point (“ X, Y is the location of a point prior to transformation”) to an output point (“ X', Y' is the location of the point after transformation”) by multiplying each coordinate component by the scale factor (“ SF is the scale factor”). A person of ordinary skill in the art would have understood this coordinate-multiplication relationship based on the standard graphics library OpenGL, which states that the “Scale” command produces a scaling matrix with the scale values on the diagonal, which a person of ordinary skill in the art would have understood means coordinate components are multiplied by the scale values. (EX1054, 39-40.) Similarly, PostScript defines “scale $s_x s_y$ ” and states it “scales the units of the user coordinate space by a factor of s_x ... and s_y ...,” and provides the scaling matrix for that transformation (“ $X'=XSF$; $Y'=YSF$ ”). (EX1053, 668.)

221. Pad's scaling operation is consistent with this same mathematical mechanism because its "scale" command multiplies an item's size by a scale amount ("SF is the scale factor"). (EX1005, 36.) In view of SVG's matrix-based scaling and Pad's multiplication-by-scale-factor scaling, a person of ordinary skill in the art would have understood the scaling is performed by multiplying each point's X and Y coordinates by the scale factor to define the rendered objects ("wherein the mathematical transformation comprises, $X'=XSF$; $Y'=YSF$; 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").

37. Claim 36

- i) [36.a] "The method of claim 35, wherein the mathematical transformation is applied to points in a first coordinate system comprising a resolution independent coordinate system associated with a virtual display area onto which page layout information is mapped"**

222. In the Zaurus-Pad-SVG combination, Pad places objects on the Pad surface and the user navigates that surface by panning and zooming, with the content residing on a "virtual display area" independent of what portion is currently visible ("a virtual display area onto which page layout information is mapped"). (EX1007, 73.) Pad further confirms that this surface uses a resolution-independent coordinate system because it allows the coordinate "units" for "all coordinates used by Pad" to be specified as physical units such as "points," "mm," or "inches," rather than being

tied to device pixels (“a first coordinate system comprising a resolution independent coordinate system”). (EX1005, 13.) Pad also discloses “pushcoordframe” that provide “all coordinates are specified within the unit square” mapped to the frame, which defines a coordinate system independent of display pixel resolution. (EX1005, 34.) A person of ordinary skill in the art would have understood that this understanding is suggested by Adobe’s PostScript, which explains that coordinates are expressed in “user space,” may be integers or real numbers, and that the “resolution of coordinates in user space is not related in any way to the resolution of pixels in device space” (“a resolution independent coordinate system”). (EX1053, 197-198.)

- ii) **[36.b] “to a second coordinate system comprising a device coordinate system corresponding to a pixel resolution of the display, wherein points are mapped from the first coordinate system to the second coordinate system using the mathematical transformation.”**

223. In the Zaurus-Pad-SVG combination, Pad distinguishes “window” coordinates from Pad coordinates by providing “padxy,” which “Convert[s] a window point (x, y) to pad coordinates,” demonstrating that the window provides a device/display coordinate system and that coordinates are converted between the two spaces using the current view transformation (“a second coordinate system comprising a device coordinate system corresponding to a pixel resolution of the

display”). (EX1005, 33.) Pad also specifies the Pad widget width “in pixels.” (EX1005, 13.)

224. Pad defines the view as “xview yview zoom,” where “(xview,yview) specifies the point at the center of the window” and “zoom specifies the magnification,” and “moveto” changes the view to place the specified center point on screen “with a magnification of zoom.” (EX1005, 30, 32.) A person of ordinary skill in the art would have understood that rendering applies this view transformation to map Pad surface points (“first coordinate system”) into window pixel coordinates (“second coordinate system”), and that “padxy” reflects the inverse of that same mathematical transformation (“points are mapped from the first coordinate system to the second coordinate system using the mathematical transformation”). Adobe’s PostScript also explains that the interpreter “automatically transforms user space coordinates into device space,” and further notes that “Pixel boundaries always fall on integer coordinates in device space,” confirming a pixel-based device coordinate system distinct from a resolution-independent user space. (EX1053, 197, 516.)

38. Claim 37: “The method of claim 1, wherein the markup language-based Web content includes cascading style sheet content defining layout and presentation attributes for the Web page.”

225. In the Zaurus-Pad-SVG combination, the retrieved markup-language Web content may include CSS content defining layout and presentation attributes, and SVG teaches CSS support (including inheritance/cascading) for embedded SVG

fragments within an XML/HTML Web page, including CSS-based control of presentation attributes. SVG is “an application of XML” compatible with CSS within markup-language web content: “SVG should use CSS positioning for establishing a viewport for an outermost SVG element ... or an SVG fragment within an XML Web page” (“markup language-based Web content includes cascading style sheet content defining layout ... for the Web page”). (EX1008, 4, 11.) SVG references a parent grammar passing its “current style table” and discussing CSS inheritance, reflecting the cascading and inheritance framework associated with CSS (“cascading style sheet content”). (EX1008, 13.)

226. SVG identifies presentation characteristics (e.g., “Fill options ... gradients” and “Stroke options ... line width”), and states that attributes that make sense to be part of a style “should be expressed in CSS” (“presentation attributes”). (EX1008, 10, 4.) SVG further requires CSS-based presentation control, stating under “CSS support” that “All CSS text/font attributes ... should be honored by all conforming SVG processors” (“cascading style sheet content defining ... presentation attributes”). (EX1008, 13.) SVG also explains that CSS stylesheets may be used to apply properties to “both the text AND graphic objects/assemblies within an embedded SVG component (to provide/enforce ‘house style’...),” reinforcing that CSS defines presentation attributes across page content (“defining ... presentation attributes for the Web page”). (EX1008, 13.)

39. Claims 38 and 73

- i) [38] “The method of claim 1, wherein the markup language-based Web content comprises HTML-based Web content that is processed by the rendering engine to generate page layout information corresponding to the original page layout as interpreted by the rendering engine.”
[73] “The apparatus of claim 42, wherein the markup language-based Web content comprises HTML-based Web content that is processed by the rendering engine to generate page layout information corresponding to the original page layout as interpreted by the rendering engine.”

227. In the Zaurus-Pad-SVG combination, Zaurus discloses that the markup language-based Web content is HTML-based Web content, explaining that the device’s “Internet browser ... incorporates HTML 3.2 specifications” (“markup language-based Web content comprises HTML-based Web content”). (EX1004, 40.) Zaurus further discloses that this HTML-based content is processed by the Internet browser (“rendering engine”) for display, including handling “character fonts and sizes” and “images.” (EX1004, 41.) Pad states: “When the html data is fetched, it is parsed and the HTML item is created which contains a method for rendering the page,” and further explains that “[t]he page will be re-laid out according to the new width” (“processed by the rendering engine to generate page layout information corresponding to the original page layout as interpreted by the rendering engine”). (EX1005, 60, 63.)

40. Claims 39, and 109

- i) **[39.a] “The method of claim 1, wherein the Web page content includes a plurality of images, the method further comprising:”**

228. In the Zaurus-Pad-SVG combination, Zaurus discloses Web pages including multiple images by stating its Internet browser can display images in GIF and JPEG formats, and that multiple animated GIFs can be displayed on a single page (“includes a plurality of images”). (EX1004, 40.) Pad likewise discloses HTML content with “in-line images” fetched and rendered as part of HTML items (“Web page content includes ... images”). (EX1005, 60-61.)

- ii) **[39.b] “generating page layout information including a bounding box for each of the plurality of images;”**
[109.a] “The method of claim 108, wherein the page layout information that is generated includes a bounding box for each of the plurality of images, and”

229. In the Zaurus-Pad-SVG combination, Pad treats each rendered image as an “image” item, stating that “[i]tems of type image appear on the display as color images,” and that the “-image” option “specifies the image that this item will render,” meaning each image is represented as its own renderable object. (EX1005, 64-65.) Pad generates bounding boxes for each object via the “bbox” command, which “Returns ... the bounding box for all the items named,” that defines each item’s extents that bound the item’s “drawn areas” (“including a bounding box for each...image”). (EX1005, 17, 36.) Pad further explains a “coordinate frame” is

itself “a bounding box,” reinforcing bounding-box-based layout information for content objects (“page layout information including a bounding box”). (EX1005, 34-35.) Thus, for a Web page having a plurality of images, Pad represents each image as its own “image” item, and Pad’s “bbox” returns the bounding box for the specified item, so each image item has a corresponding bounding box included in the generated page layout information. (EX1005, 17, 36, 64-65.)

- iii) **[39.c] “scaling the bounding box associated with an image to produce a scaled bounding box; and”**
[109.b] “wherein scaled Web page content including one or more images is generated by scaling bounding boxes corresponding to the one or more images to produce scaled bounding boxes and,”

230. In the Zaurus-Pad-SVG combination, Pad discloses scaling an item’s bounding box via its “scale” command, which “Scale[s] ... by multiplying the size of the item with scaleAmount,” and Pad’s “bbox” reports the item’s extents (“x1 y1 x2 y2”) that bound the item’s “drawn areas” (“scaling the bounding box ... to produce a scaled bounding box”). (EX1005, 17, 36.) Pad also defines the view as including “zoom” where “zoom specifies the magnification,” tying the scaled bounding box to view zoom level (“scaled bounding box”). (EX1005, 30.) Relatedly, Pad explains that “Text and images are scaled” relative to a bounding-box coordinate frame, confirming that when an image item is scaled, the bounding region that encloses its drawn areas correspondingly scales (“scaling ... to produce a scaled bounding box”). (EX1005, 35.)

- iv) **[39.d] “rendering bitmap content associated with the image to fit within the scaled bounding box.”**
[109.c] “for each of the one or more images, scaling content associated with the image to produce a scaled image that fits within its corresponding scaled bounding box.”

231. In the Zaurus-Pad-SVG combination, Zaurus discloses that the displayed Web images are GIF/JPEG bitmaps. (EX1004, 40.) Pad’s images are bitmap-based: “allocimage” reads “gif” files and (with libraries) “jpeg,” and it describes internal pixel storage costs (for example “5 bytes per pixel”) (“rendering bitmap content associated with the image”). (EX1005, 17.) Pad further explains that a “coordinate frame” is a “bounding box,” and that within a coordinate frame “Text and images are scaled” such that “the height of the image is scaled to the height of the coordinate frame ... (using the -place or -z itemconfigure options),” meaning once the image item’s bounding region is set (including after scaling/zoom), the bitmap content is scaled/rendered to fit within that “scaled bounding region.” (EX1005, 34-36.)

232. Pad then renders that bitmap content through (1) image items whose “-image” option “specifies the image that this item will render,” and (2) the “drawimage” command, which “draws the image specified by imagetoken.” (EX1005, 65, 25.) Pad further states that under its zoomable coordinate-frame rendering, “Text and images are scaled” and that “the height of the image is scaled to the height of the coordinate frame ... (using the -place or -z itemconfigure

options)” (“for each of the one or more images, scaling content associated with the image to produce a scaled image”). (EX1005, 35-36.) Zaurus suggests that the displayed Web images in the combined system are GIF/JPEG bitmaps. (EX1004, 40.)

41. Claims 40, 75, and 107

- i) [40] “The method of claim 1, wherein text content in a view is rendered to have a resolution that is approximately proportional to the zoom level of the view, such that as the zoom level increases the resolution of text content increases approximately proportionally.”
[75] “The apparatus of claim 42, wherein text content in a view is rendered to have a resolution that is approximately proportional to the zoom level of the view, such that as the zoom level increases the resolution of text content increases approximately proportionally.”
[107] “The machine-readable medium of claim 77, wherein text content in a view is rendered to have a resolution that is approximately proportional to the zoom level of the view, such that as the zoom level increases the resolution of text content increases approximately proportionally.”

233. In the Zaurus-Pad-SVG combination, Pad defines the view as “xview yview zoom,” where “zoom specifies the magnification,” tying rendering to a numeric zoom level (“zoom level of the view”). (EX1005, 30.) Pad further defines “magnification” as “the multiplication of the current view ... with the object’s size,” showing that the rendered size of an object is “approximately proportional to the zoom level.” (EX1005, 29-30.)

234. As I explained for Claim 10, for scalable-font rendering and bitmap generation keyed to size/font, Pad's Type 1 outline-based rendering necessarily produces higher-resolution text bitmaps at higher zoom/magnification (and lower-resolution text at lower zoom), such that text resolution increases approximately proportionally with zoom level. (See Section IX.A.12; EX1005, 11-12, 29-30, 77.) Pad states that for text it supports "type 1 font" selection, and when drawing Type 1 text it "remembers what size, font and bitmap it used" for a character and "reuses the last bitmap image" only when asked to draw the same character again "at the same size and font," otherwise generating the bitmap from the character's "outline description." (EX1005, 12, 77.) A person of ordinary skill in the art would have understood that as zoom changes the rendered size, Pad generates a correspondingly sized bitmap from outlines where more pixels are used to represent each bitmap at the larger zoom-scaled size ("as the zoom level increases the resolution of text content increases approximately proportionally"). Pad further suggests variable-resolution rendering by explaining that under animation timing constraints "objects may be rendered at low resolution," confirming that Pad's renderer manages resolution as part of view rendering. (EX1005, 11.)

235. Pad provides an example confirmation that zoom changes the readability and detail of text, noting: "When you zoom in, the major headings become legible." (EX1007, 52, 59, 66.) SVG confirms this proportional zoom

behavior for text by requiring “with true scalability” where “text ... should grow/shrink uniformly with the zoom level,” consistent with the claimed proportional relationship between zoom and rendered text resolution. (EX1008, 14.) SVG is designed to replace raster graphics with a vector format that supports “text data” with “precise control of text sizing [and] positioning... so that graphic artists have a way to ensure that the end user sees what was intended.” (EX1008, 1, 7-8.)

42. Claims 41 and 76

- i) **[41] “The method of claim 1, wherein the layout of content for views associated with a zoom level is effected by scaling page layout information using a scale factor associated with the zoom level.”**
[76] “The apparatus of claim 42, wherein the layout of content for views associated with a zoom level is effected by scaling page layout information using a scale factor associated with the zoom level.”

236. In the Zaurus-Pad-SVG combination, Pad effects layout for a given zoom-level view by applying the view’s scale factor (zoom) to stored layout information. (See Sections IX.A.10, 11; EX1005, 32, 36.) Pad represents each “view” as “xview yview zoom,” where “zoom specifies the magnification” (“views associated with a zoom level”). (EX1005, 30.) Pad changes the view using “moveto,” which “Change[s] the view ... with a magnification of zoom” (“scale factor associated with the zoom level”). (EX1005, 32.) Pad’s layout geometry, including bounding boxes returned by “bbox,” is view-dependent because “the bounding box will be different when the view is different” (“the layout of content”).

for views associated with a zoom level is effected by scaling page layout information”). (EX1005, 17.) Pad confirms the scaling mechanism by defining “Magnification” as “the multiplication of the current view ... with the object’s size,” which is scaling stored layout and size information by the view’s zoom. (EX1005, 29.) Pad further discloses scaling by multiplication, because “scale” operates by “multiplying the size of the item with scaleAmount,” consistent with “scaling page layout information using a scale factor associated with the zoom level.” (EX1005, 36.)

43. Claims 44 and 74

- i) [44] “The apparatus of claim 42, wherein the hand-held mobile device comprises a mobile phone.”
[74] “The apparatus of claim 42, wherein the network comprises a mobile service provider network and the hand-held mobile device comprises a mobile phone.”

237. As I explained for Claims 8 and 9, in the Zaurus-Pad-SVG combination, Zaurus discloses a mobile handheld with cellular connectivity to access the Internet, including data communication “with base stations” (“the network comprises a mobile service provider network”; “the hand-held mobile device comprises a mobile phone”). (EX1004, 7-8 (adapters), 10 (portable), 16-17, 24 (modem).) Pad is compatible with “a small home computer” and is “especially attractive for systems which have small screens, such as handheld computers (i.e. PDA’s).” (EX1007, 139, 141.) SVG supports “small mobile devices.” (EX1008, 1, 4.) In the

combination, a person of ordinary skill in the art would have implemented the combined browsing/zooming system on a mobile phone (hand-held device with cellular network connectivity), given Zaurus's cellular-network adapters and the handheld/mobile targets of Pad and SVG.

44. Claims 69 and 105

- i) **[69] “The apparatus of claim 42, wherein the Web page includes text content and graphic layout objects, and views are generated through use of scaled text content and graphic layout objects.”**
[105] “The machine-readable medium of claim 77, wherein the Web page includes text content and graphic layout objects, and in generated views the text content and graphic layout objects are scaled.”

238. In the Zaurus-Pad-SVG combination, Zaurus's browser settings provide that either “image data is displayed” or “only text data is displayed” (“the Web page includes text content and graphic layout objects”). (EX1004, 42.) Zaurus further suggests graphic layout objects by explaining its Internet browser can display images in formats such as GIF and JPEG. (EX1004, 40.)

239. Pad then teaches that generated views scale both text and images/graphics (generated views of scaled “text content and graphic layout objects”). (EX1005, 64, 75.) Pad defines the view as “xview yview zoom” where “zoom specifies the magnification,” and defines magnification as multiplying the current view by the object's size. (EX1005, 29, 30.) Pad confirms this applies to both object types, stating “Text and images are scaled.” (EX1005, 35.) SVG

suggests the same scaling behavior for graphics and text under zoom with “true scalability,” where objects including text scale uniformly with zoom. (EX1008, 1, 7-8, 11, 14.)

45. Claim 72: “The apparatus of claim 42, wherein the storage means comprises a flash memory.”

240. In the Zaurus-Pad-SVG combination, Zaurus discloses flash memory in its specifications, listing “Main body memory ... flash memory” (e.g., 4MB or 8MB depending on model), which satisfies “storage means comprises a flash memory.” (EX1004, 18.) Zaurus further references using “main body flash” (and card flash) for storage (e.g., cache placement), confirming flash as a storage medium in the device. (EX1004, 43.)

46. Claim 85: “The machine-readable medium of claim 77, wherein execution of the instructions performs further operations comprising scaling the scalable page layout information and/or data derived therefrom to regenerate views of the Web content in response to associated user inputs to enable the user to iteratively zoom in and out a display of the Web page.”

241. In the Zaurus-Pad-SVG combination, Pad enables iterative zooming, stating that “this process can be repeated indefinitely” and “at any time you can use the mouse to pan and zoom” (“regenerate views of the Web content in response to associated user inputs to enable the user to iteratively zoom in and out a display of the Web page”). (EX1007, 6, 71.) Pad also discloses zooming out/overview during navigation (e.g., “zoom out to see an overview,” “Zoom out again,” or click a

“hotword” for the “appropriate document”), confirming iterative zoom in both directions. (EX1007, 10, 82, 108.) As I explained for Claims 8 and 9, the regenerated views correspond to updated scale factor/offset view parameters. (See Sections IX.A.10, 11.) Pad ties these regenerated views to a scale factor by defining the view as “xview yview zoom” where “zoom specifies the magnification,” and defining magnification as multiplication of the current view with the object’s size (“scaling the scalable page layout information and/or data derived therefrom”). (EX1005, 29, 30, 35 (“Text and images are scaled”), 36 (“scaleAmount”), 60 (“html data is fetched”).)

47. Claim 103: “The machine-readable medium of claim 77, wherein at least a portion of the instructions comprise Java-based instructions configured to be executed on a Java virtual machine.”

242. In the Zaurus-Pad-SVG combination, Zaurus discloses a “Personal Digital Assistant” that a person of ordinary skill in the art would have understood could execute various software instructions. (EX1004, 3.) Pad was “looking into connecting Pad++ to Java so that it will be possible to write zooming applications entirely in Java using the C++ core,” and it further discusses building Pad++ in Java and even a “simple zooming Java applet.” (EX1007, 140.) The Zaurus browser supported HTML 3.2 and SVG was defined by the W3C, which explained that HTML “adds widely deployed features such as ... applets,” and its APPLET element “is supported by all Java enabled browsers” for embedding a “Java applet into

HTML documents.” (EX1043, 2, 26.) A person of ordinary skill in the art would have known that Java was a standard platform for rendering interactive Web content on mobile devices in the relevant timeframe and implemented “at least a portion of” the claimed instructions using Java-based instructions to ensure cross-platform compatibility and security on the mobile device. (EX1055.)

48. Claim 110

- i) **[110.a] “The method of claim 108, further comprising:”**
[110.b] “processing the HTML-based Web content with the rendering engine to parse the text content into a plurality of text objects;”

243. In the Zaurus-Pad-SVG combination, Pad processes HTML content by parsing it: “When the html data is fetched, it is parsed and the HTML item is created,” and it also discloses that HTML anchors are created as separate items (“parse ... into a plurality of text objects”). (EX1005, 60.) Pad also defines “Text Items” that display strings, consistent with treating text as objects. (EX1005, 75.)

- ii) **[110.c] “generating page layout information including a bounding box for each of the plurality of text objects; and”**

244. In the Zaurus-Pad-SVG combination, Pad provides bounding boxes for items via “bbox,” and provides text-bounding functionality (e.g., “gettextbbox”). (EX1005, 17, 29.) Because Pad represents text as text items (“A text item displays a string of characters”), a person of ordinary skill in the art would have understood

that “bbox” provides a bounding box for each text object when queried per item (“including a bounding box for each of the plurality of text objects”). (EX1005, 17, 75.)

iii) [110.d] “employing page layout information associated with the bounding boxes to effect preservation of the original page layout of the text content.”

245. In the Zaurus-Pad-SVG combination, Pad employs this layout information to preserve the layout, explaining that the HTML item “specifies the width ... of the html page” and “[t]he page will be re-laid out according to the new width.” (EX1005, 63.) By using the bounding boxes and placement coordinates, the combined device would maintain the relative positions of the text objects as defined by the HTML (“effect preservation of the original page layout”).

246. Pad’s HTML item specifies the width of the html page and states the page “will be re-laid out according to the new width,” showing the rendering engine uses layout information (including extents/bounds) to lay out content while preserving the page’s relative arrangement as interpreted by the renderer. (EX1005, 63.) Once laid out, Pad maintains per-object layout geometry. (EX1005, 17, 50.) Pad then preserves that layout across zoom-level views by scaling the layout geometry using the view’s scale factor. (EX1005, 29-30, 50.) Because the zoom-level view is generated by scaling the stored object layout geometry, the relative arrangement of the text objects is preserved as the view is regenerated at different

zoom levels. (EX1007, 73; *see also* EX1008, 8 (SVG emphasizes control of text sizing/positioning).)

49. Claim 111

- i) **[111.a] “The method of claim 110, further comprising:”**
[111.b] “scaling bounding boxes for text objects corresponding to text content in a view; and”

247. As I explained for Claims 9 and 24, in the Zaurus-Pad-SVG combination, Pad’s per-item bounding boxes scale with the current view’s zoom parameter. (*See* Sections IX.A.11 (view parameters), IX.A.26 (“bbox”).) Pad explains that for sticky items, “bbox” returns the bounding box “as it appears for the current view” and differs when the view differs, showing bounding boxes scale with view changes. (EX1005, 17.) Pad’s view includes “zoom” where “zoom specifies the magnification.” (EX1005, 29-30.) Because Pad represents text as “text items,” a person of ordinary skill in the art would have understood that “bbox” applies to text items and that changing zoom correspondingly is “scaling bounding boxes for text objects corresponding to text content in a view.” (EX1005, 17, 75.)

- ii) **[111.c] “employing the scaled bounding boxes to lay out corresponding text content within the scaled bounding boxes.”**

248. In the Zaurus-Pad-SVG combination, Pad explains that “all coordinates are interpreted relative to the frame,” and that a “frame is a bounding box,” with coordinates specified within the unit square mapped to that bounding box. (EX1005,

34.) Pad explains text items have size that can be changed with the “scale” command or the “-place” option, and that “Text and images are scaled” (“employing the scaled bounding boxes to lay out corresponding text content within the scaled bounding boxes”). (EX1005, 29, 35, 36, 75.) SVG is designed to replace raster graphics with a vector format that supports “text data” with “precise control of text sizing [and] positioning... so that graphic artists have a way to ensure that the end user sees what was intended.” (EX1008, 1, 7-8.)

50. Claim 112: “The method of claim 111, further comprising: employing a scalable font to scale text content for a text object to fit within a scaled bounding box.”

249. As I discussed for limitation [10.c], Pad uses scalable Type 1 fonts whose rendered size varies with zoom/scale (and/or font height). (EX1005, 62; Section IX.A.12.ii.) As I discussed for Claims 5, 9, 23, and 24, when the text object’s bounding box is scaled for the zoom-level view, the text rendering is correspondingly scaled to fit within that scaled bounding box. (See Sections IX.A.7, 11, 25, 26.)

250. In the Zaurus-Pad-SVG combination, Pad discloses that a specified font for rendering text “must specify a filename which contains an Adobe Type 1 font” (“employing a scalable font”). (EX1005, 62.) Pad further explains its Type 1 text drawing generates “the bitmap for the character from its outline description” when needed, showing text is rasterized at the current rendered size (“scale text content”).

for a text object to fit within a scaled bounding box”). (EX1005, 12.) Pad also states that text items (with a “vector” font) are created at a default size and “can be changed with the scale command or the -place itemconfigure option,” such as “to the height of the coordinate frame.” (EX1005, 34-35, 75.) A person of ordinary skill in the art would have understood that when claim 111’s bounding box for a text object is scaled, Pad’s coordinate-frame and placement scaling correspondingly scales the text rendering so the text fits within the scaled bounding-box frame.

51. Claim 113

- i) [113.a] “The method of claim 111, further comprising:”
[113.b] “in conjunction with rendering a view of the Web page content,”
[113.c] “employing the bounding boxes and/or data derived therefrom to determine image and text objects to be rendered.”

251. In the Zaurus-Pad-SVG combination, Pad discloses text and image objects as distinct renderable item types (“text” items and “image” items) (“in conjunction with rendering a view of the Web page content”). (EX1005, 64, 75.) Pad provides per-item bounding information via “bbox,” and supports region-based selection of items via “overlapping,” which returns items overlapping a specified rectangular region. (EX1005, 17, 27.) Pad “uses an R-Tree internally to quickly determine which objects are visible in a given view.” (EX1007, 141.) A person of ordinary skill in the art would have understood that the renderer employs bounding

boxes (and derived data) to determine which text and image objects fall within the current view and therefore should be rendered (“employing the bounding boxes and/or data derived therefrom to determine image and text objects to be rendered”). (EX1056, 5 (“An R-tree is an index structure for n-dimensional spatial objects ... It is a height-balanced tree with records in the leaf nodes each containing an n-dimensional rectangle and a pointer to a data object having the rectangle as a bounding box.”).) Also see my explanations for Claims 8, 24, and 26. (Sections IX.A.10, 26, 28.)

X. CONCLUSION

252. I declare that all statements made herein of my knowledge are true, and that all statements made on information and belief are believed to be true, and that these statements were made with the knowledge that willful false statements and the like so made are punishable by fine or imprisonment, or both, under Section 1001 of Title 18 of the United States Code.

Dated: March 2, 2026

By: 
Dr. Benjamin Goldberg

Appendix A - Comparison to Unpatentable Claims

U.S. Patent No. 7,844,889	U.S. Patent No. 7,461,353	U.S. Patent No. 7,831,926	Comparison
[1.pre] A method comprising: [42.pre] An apparatus comprising a hand-held mobile device, including: [42.a] a processor, [42.b] a communications device operatively coupled to the processor, to enable the apparatus to be wirelessly linked to a network to access Web content; [42.c] a display; and [42.d] storage means, operatively coupled to the processor, in which a plurality of instructions, including instructions	[1.pre] A wireless device, comprising: [1.a] 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, [36.pre] A mobile hand-held device, comprising: [36.a] a processor, a	Appendix C (IPR2013-00004) [30.pre] A mobile phone, comprising: [30.a] a processor, [30.b] wireless communications means operatively coupled to the processor, to facilitate communication with a mobile service provider network via which Web content may be accessed; [30.c] a touch-sensitive display; [30.d] a memory, operatively coupled to the processor; and	A hand-held device or medium with a processor, wireless communications, a display, and stored browser instructions (including a rendering engine) for accessing Web content.

U.S. Patent No. 7,844,889	U.S. Patent No. 7,461,353	U.S. Patent No. 7,831,926	Comparison
corresponding to a browser having a rendering engine and instructions corresponding to an operating system, are stored that when executed by the processor enable the apparatus to perform operations including, [74] The apparatus of claim 42, wherein the network comprises a mobile service provider network and the hand-held mobile device comprises a mobile phone. [77.pre] A tangible non-transitory machine-readable medium having stored thereon a plurality of instructions comprising a browser including a rendering engine configured to be executed by a wireless hand-held device having a display	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, [252.pre] 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: [317.pre] A hand-held wireless device,	[30.e] 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, [52.pre] A mobile device, comprising: [52.a] a processor, [52.b] wireless communications means, to facilitate wireless communication with a network via which Web content may be accessed; [52.c] a touch-sensitive display; [52.d] flash memory, operatively coupled to the processor, in which a	

U.S. Patent No. 7,844,889	U.S. Patent No. 7,461,353	U.S. Patent No. 7,831,926	Comparison
and an operating system and configured to access Web content via a network, wherein execution of the instructions in conjunction with execution of the operating system enables the device to perform operations comprising:	comprising: [317.a] a processor, [317.b] a wireless communications interface, to facilitate wireless communication with a network that supports access to the Internet; [317.c] a display; and [317.d] 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,	plurality of instructions are stored that when executed by the processor enable the mobile device to perform operations including, Appendix D (90/009,995) [1.pre] A mobile device, comprising: [1.a] a processor, [1.b] a wireless communications device operatively coupled to the processor, to facilitate communication with a network via which Web content may be accessed; [1.c] a touch-sensitive display; [1.d] a memory, operatively coupled to the processor; and	

U.S. Patent No. 7,844,889	U.S. Patent No. 7,461,353	U.S. Patent No. 7,831,926	Comparison
		[1.e] 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, Appendix E (95/000,635) [79.pre] A mobile device, comprising: [79.a] processing means; [79.b] wireless communications means, to facilitate wireless communication with a network via which Web content may be accessed; [79.c] touch-sensitive display means, to facilitate user input and display rendered content;	

U.S. Patent No. 7,844,889	U.S. Patent No. 7,461,353	U.S. Patent No. 7,831,926	Comparison
		[79.d] programmed circuit means; and [79.e] storage means, in which a plurality of instructions are stored,	
[1.a] in response to a request to access a Web page made via a wireless hand-held device including a display, retrieving markup language-based Web content associated with the Web page having an original format defining an original page layout of content on the Web page, the content including text content, a plurality of hyperlinks, and a plurality of images; [42.d.1] in response to a request to access a Web page made via a user of the	[1.c] 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	Appendix C (IPR2013-00004) [30.f] 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; [30.g] retrieving HTML-based content associated with the Web page;	Retrieving Web page markup/HTML/markup language-based content in an original format (desktop-designed) that includes text, images, and hyperlinks.

U.S. Patent No. 7,844,889	U.S. Patent No. 7,461,353	U.S. Patent No. 7,831,926	Comparison
apparatus, retrieving markup language-based Web content associated with the Web page having an original format defining an original page layout, functionality and design of content on the Web page, the content including text content, a plurality of hyperlinks, and a plurality of images; [77.a] in response to a request to access a Web page made via a user of the device, retrieving markup language-based Web content associated with the Web page having an original format defining an original page layout, functionality and design of content on the Web page, the content including text content, a plurality of hyperlinks, and a plurality	original format when scaled and rendered; and [317.e] rendering a browser interface on the display via which a user is enabled to request access to a Web page including at least one image, at least one column, at least one hyperlink to an external reference and having a width and height; [317.f] retrieving the Web page via the wireless communications interface;	[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. [30.j] wherein preservation of the functionality defined by the HTML-based content includes	

U.S. Patent No. 7,844,889	U.S. Patent No. 7,461,353	U.S. Patent No. 7,831,926	Comparison
of images; [108.a] in response to a request to access a Web page made via a mobile hand-held device, retrieving HTML-based Web content associated with the Web page having an original format defining an original page layout, functionality, and design of content on the Web page, the content including text content, a plurality of hyperlinks, and a plurality of images;		preservation of hyperlink functionality. Appendix D (90/009,995) [1.f] 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; [1.g] retrieving HTML-based Web content associated with the Web page; [6.a] 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:	

U.S. Patent No. 7,844,889	U.S. Patent No. 7,461,353	U.S. Patent No. 7,831,926	Comparison
		[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. Appendix E (95/000,635) [79.g] rendering a browser interface via which a user is enabled to request to access to a Web page comprising HTML-based Web content	

U.S. Patent No. 7,844,889	U.S. Patent No. 7,461,353	U.S. Patent No. 7,831,926	Comparison
		defining an original page layout, functionality, and design of content on the Web page; [79.h] retrieving and processing the HTML-based Web content to produce scalable content; and [58.a] 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: [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	

U.S. Patent No. 7,844,889	U.S. Patent No. 7,461,353	U.S. Patent No. 7,831,926	Comparison
		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.	
[1.b] processing, via the wireless hand-held device, the markup language-based Web content with a rendering engine to generate page layout information corresponding to the original page layout of content on the Web page as interpreted by the rendering engine including a width of the Web page; [42.d.2] processing the	[1.b] 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;	Appendix C (IPR2013-00004) [52.f] retrieving and processing the HTML-based Web content to produce scalable content; and [52.i] 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	Processing markup with a rendering engine to generate page layout information corresponding to the original page layout as interpreted by the rendering engine.

U.S. Patent No. 7,844,889	U.S. Patent No. 7,461,353	U.S. Patent No. 7,831,926	Comparison
markup language-based Web content with the rendering engine to generate page layout information corresponding to the original page layout as interpreted by the rendering engine including a width of the Web page; and [77.b] processing the markup language-based Web content with the rendering engine to generate page layout information corresponding to the original page layout as interpreted by the rendering engine including a width of the Web page; [108.b] processing the HTML-based Web content with a rendering engine to generate page layout information corresponding	[36.c] retrieving the Web page via the wireless communications device, and processing HTML-based Web content to produce scalable content; and [58.b] parsing HTML-based code corresponding to the retrieved Web page to determine the original page layout of the content on 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, [75.a] 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: Appendix D (90/009,995) [21.b] processing the HTML-based Web content with a rendering engine to generate page layout information corresponding to the original page layout	

U.S. Patent No. 7,844,889	U.S. Patent No. 7,461,353	U.S. Patent No. 7,831,926	Comparison
to the original page layout as interpreted by the rendering engine;		as interpreted by the rendering engine; and [25.a] 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 [25.b] determining an applicable scale factor to fit the width of the Web page across a display area of the touch-sensitive display; and [25.c] employing the scale factor that is determined as the first scale factor. Appendix E (95/000,635)	

U.S. Patent No. 7,844,889	U.S. Patent No. 7,461,353	U.S. Patent No. 7,831,926	Comparison
		[79.i.2] 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,	
[1.c] employing the page layout information to generate scalable page layout information; and [77.c] employing the page layout information to generate scalable page layout information; and [108.c] employing scaled page layout information	[1.c] 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	Appendix C (IPR2013-00004) [30.h] 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	Generating scalable content or scalable page layout information that is capable of being rendered at multiple zoom levels, and supporting a scalable, resolution-independent content or layout information from the Web page.

U.S. Patent No. 7,844,889	U.S. Patent No. 7,461,353	U.S. Patent No. 7,831,926	Comparison
and/or data derived therefrom to generate scaled Web page content at multiple scale factors; and	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 [36.c] retrieving the Web page via the wireless communications device, and processing HTML-based Web content to produce scalable content; and	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 Appendix D (90/009,995) [1.h] translating the HTML-based Web content to produce scalable vector-based page layout information; Appendix E (95/000,635) [79.h] retrieving and processing the HTML-based Web content to produce scalable content; and	
[1.d] employing scalable page layout information and/or data derived therefrom to generate	[149.c] employing at least one of the scalable content	Appendix C (IPR2013-00004)	Employing scalable page layout information

U.S. Patent No. 7,844,889	U.S. Patent No. 7,461,353	U.S. Patent No. 7,831,926	Comparison
views of the Web page on the display, including, [42.d.3] employing scaled page layout information and/or data derived therefrom to generate views of the Web page on a display, including, [77.d] employing the scalable page layout information and/or data derived therefrom to generate views of the Web page on the display, including, [108.d.1] employing the scaled Web page content to enable a user to browse the Web page at multiple zoom levels on the mobile hand-held device	or data derived therefrom to, [149.d] render the Web page on a display of the hand-held device; and [149.e] 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.	[52.g] employing the scalable content and/or data derived therefrom to, [52.h] render a view of the Web page on the touch-sensitive display; and [52.i] 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, Appendix D (90/009,995) [1.i] employing the scalable vector-based page layout	to render displays of the Web page.

U.S. Patent No. 7,844,889	U.S. Patent No. 7,461,353	U.S. Patent No. 7,831,926	Comparison
		information and/or data derived therefrom to, [1.i.1] render a view of at least a portion of the Web page on the touch-sensitive display using a first scale factor; and [1.i.2.a] 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, Appendix E (95/000,635)	

U.S. Patent No. 7,844,889	U.S. Patent No. 7,461,353	U.S. Patent No. 7,831,926	Comparison
		[79.i] employing the scalable content and/or data derived therefrom to, [79.i.1] render a view of the Web page on the touch-sensitive display; and [79.i.2] 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,	
[1.d.1] a view under which the width of the Web page is fully displayed; and [31] The method of claim	[118.c] scaling the scalable content to render the Web page on the display such that a width of the Web page is	Appendix C (IPR2013-00004) [75.a] The mobile device of claim 52, wherein the	Rendering a width-fit view using an applicable scale factor.

U.S. Patent No. 7,844,889	U.S. Patent No. 7,461,353	U.S. Patent No. 7,831,926	Comparison
1, further comprising: generating a view under which the width of the Web page is fit across the display. [42.d.3.i] a view under which the width of the Web page is fit across the display; and [77.d.1] a view under which the width of the Web page is fully displayed; and [104] The machine-readable medium of claim 77, wherein execution of the instructions performs further operations comprising: generating a view under which the width of the Web page is fit across the display.	rendered to fit across the display.	original format of the Web page defines a width for the Web page, and wherein execution of the instructions performs further operations comprising: [75.b] determining an applicable scale factor to fit the width of the Web page across a display area of the touch-sensitive display; and [75.c] employing the scale factor to render the display area. Appendix D (90/009,995) [25.a] 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	

U.S. Patent No. 7,844,889	U.S. Patent No. 7,461,353	U.S. Patent No. 7,831,926	Comparison
		instructions performs further operations comprising [25.b] determining an applicable scale factor to fit the width of the Web page across a display area of the touch-sensitive display; and [25.c] employing the scale factor that is determined as the first scale factor. Appendix E (95/000,635) [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.	

U.S. Patent No. 7,844,889	U.S. Patent No. 7,461,353	U.S. Patent No. 7,831,926	Comparison
[1.d.2] views of the Web page generated in response to associated user inputs to enable a user to iteratively zoom in on a portion of the Web page while preserving the original page layout, functionality, and design of the content on the Web page defined by the markup language-based Web content, [42.d.3.ii] views of the Web page generated in response to associated user inputs to enable a user to iteratively zoom in on a portion of the Web page while preserving the original page layout, functionality, and design of the content on the Web page defined by the markup language-based Web content,	[36.e] 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.	Appendix C (IPR2013-00004) [52.i] 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, [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	Re-rendering in response to user input to support iterative zooming while preserving the original layout, functionality, and design defined by the HTML-based Web content (and associated design information).

U.S. Patent No. 7,844,889	U.S. Patent No. 7,461,353	U.S. Patent No. 7,831,926	Comparison
[77.d.2] views of the Web page generated in response to associated user inputs to enable a user to iteratively zoom in on a portion of the Web page while preserving the original page layout, functionality, and design of the content on the Web page defined by the markup language-based Web content, [108.d.1] employing the scaled Web page content to enable a user to browse the Web page at multiple zoom levels on the mobile hand-held device [108.d.2] while preserving the original page layout, functionality, and design of the content on the Web page defined by the HTML-based Web content		input made via the touch-sensitive display. Appendix D (90/009,995) [1.i.2.a] 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, [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	

U.S. Patent No. 7,844,889	U.S. Patent No. 7,461,353	U.S. Patent No. 7,831,926	Comparison
as interpreted by the rendering engine, [108.e.2] wherein the user is enabled to iteratively zoom in on a portion of the Web page.		input made via the touch-sensitive display. Appendix E (95/000,635) [79.i.2] 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, [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-	

U.S. Patent No. 7,844,889	U.S. Patent No. 7,461,353	U.S. Patent No. 7,831,926	Comparison
		selectable portion of a display of the Web page in response to a user interface input made via the touch-sensitive display.	
[1.e] wherein preservation of the functionality defined by the markup language-based content includes preservation of hyperlink functionality. [42.e] wherein preservation of the functionality defined by the markup language-based content includes preservation of hyperlink functionality. [77.e] wherein preservation of the functionality defined by the markup language-based content includes preservation of hyperlink functionality.	[317.i] 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;	Appendix C (IPR2013-00004) [30.j] wherein preservation of the functionality defined by the HTML-based content includes preservation of hyperlink functionality. [52.j] wherein preservation of the functionality defined by the HTML-based Web content includes preservation of hyperlink functionality. Appendix D (90/009,995) [1.i.2.b] wherein preservation of the functionality defined by the	Preserving hyperlink functionality across rendered views, including hyperlink activation.

U.S. Patent No. 7,844,889	U.S. Patent No. 7,461,353	U.S. Patent No. 7,831,926	Comparison
[108.e.1] wherein preservation of the functionality includes preservation of hyperlink functionality, and		HTML-based content includes preservation of hyperlink functionality. Appendix E (95/000,635) [79.i.2.a] wherein preservation of the functionality defined by the HTML-based Web content includes preservation of hyperlink functionality.	
[2] The method of claim 1, wherein re-rendering views of the Web page is performed in real-time. [43] The apparatus of claim 42, wherein zooming views of the Web content is enabled to be performed in real-time. [78] The machine-readable medium of claim 77, wherein zooming the views of the Web content is	[36.e] 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.	Appendix C (IPR2013-00004) [43.c] employing the display list to re-render the display at different scale factors to enable rapid zooming of the Web page. Appendix D (90/009,995) [5] The mobile device of claim 1, wherein the display of the Web page is	Responsive re-rendering at different scale factors to enable rapid zooming.

U.S. Patent No. 7,844,889	U.S. Patent No. 7,461,353	U.S. Patent No. 7,831,926	Comparison
enabled to be performed in real-time.		re-rendered in real-time to effect zooming operations. Appendix E (95/000,635) [57] The mobile device of claim 52, wherein the display of the Web page is re-rendered in real-time to effect zooming operations.	
[3] The method of claim 1, wherein the display of the wireless hand-held device comprises a touch-sensitive display configured to receive user inputs to control display of the Web page. [71] The apparatus of claim 42, wherein the display comprises a touch-sensitive display configured to receive user inputs to control display of the Web page.	[52] The mobile hand-held device of claim 51, wherein the corresponding user input comprises tapping on the image via the display. [317.j] zoom in on an image of the Web page by tapping on the image via the display; [317.k] zoom in on a column of the Web page by tapping on the column via the display; and	Appendix C (IPR2013-00004) [30.c] a touch-sensitive display; [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	Receiving user input via a touch-sensitive display to control zooming of displayed Web content.

U.S. Patent No. 7,844,889	U.S. Patent No. 7,461,353	U.S. Patent No. 7,831,926	Comparison
		input made via the touch-sensitive display. Appendix D (90/009,995) [1.c] a touch-sensitive display; Appendix E (95/000,635) [79.c] touch-sensitive display means, to facilitate user input and display rendered content;	
[4.a] The method of claim 1, further comprising: [4.b] enabling the user to select a hyperlink; and, in response thereto, [4.b.1] retrieving and processing markup language-based Web content associated with the hyperlink to produce additional scalable page	[317.i] 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;	Appendix C (IPR2013-00004) [30.g] retrieving HTML-based content associated with the Web page; [30.h] translating at least a portion of the HTML-based content from its original format to produce translated content including scalable vector-based	Selecting a hyperlink and retrieving and rendering linked Web content.

U.S. Patent No. 7,844,889	U.S. Patent No. 7,461,353	U.S. Patent No. 7,831,926	Comparison
layout information; and [4.b.2] employing the additional scalable page layout information and/or data derived therefrom to generate a view of the Web content associated with the hyperlink. [45.a] The apparatus of claim 42, wherein execution of the instructions performs further operations comprising: [45.b] enabling the user to select a hyperlink; and, in response thereto, [45.b.1] retrieving and processing the markup language-based Web content associated with the hyperlink to produce additional page layout	[36.c] retrieving the Web page via the wireless communications device, and processing HTML-based Web content to produce scalable content; and	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 [30.j] wherein preservation of the functionality defined by the HTML-based content includes preservation of hyperlink functionality. Appendix D (90/009,995) [6.a] The mobile device of claim 1, wherein the Web content includes at least one hyperlink, and wherein execution of the instructions performs	

U.S. Patent No. 7,844,889	U.S. Patent No. 7,461,353	U.S. Patent No. 7,831,926	Comparison
information; and [45.b.2] employing the additional page layout information and/or data derived therefrom to generate a view of the Web content associated with the hyperlink. [79.a] The machine-readable medium of claim 77, wherein execution of the instructions performs further operations comprising: [79.b] enabling a user to select a hyperlink; and, in response thereto, [79.c] retrieving and processing the markup language-based Web content associated with the hyperlink to produce additional scalable page		further operations comprising: [6.b] enabling the user to select the hyperlink via the touch-sensitive display; and, in response thereto, [6.c] retrieving and translating HMTL-based Web content associated with the hyperlink to produce additional scalable vector-based page layout information; and [6.d] 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. Appendix E (95/000,635)	

U.S. Patent No. 7,844,889	U.S. Patent No. 7,461,353	U.S. Patent No. 7,831,926	Comparison
layout information; and [79.d] employing the additional scalable-page layout information and/or data derived therefrom to generate a view of the Web content associated with the hyperlink.		[58.a] 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: [58.b] enabling the user to select the hyperlink via the touch-sensitive display; and, in response thereto, [58.c] retrieving and processing HMTL-based Web content associated with the hyperlink to produce additional scalable content; and [58.d] employing the additional scalable content and/or data derived therefrom to render the Web content associated	

U.S. Patent No. 7,844,889	U.S. Patent No. 7,461,353	U.S. Patent No. 7,831,926	Comparison
		with the hyperlink on the touch-sensitive display.	
[5.a] The method of claim 1, further comprising: [5.b] parsing and processing markup language code with the rendering engine to determine the original page layout of display content within the Web page as interpreted by the rendering engine, wherein the original page layout defines a layout location for a plurality of objects, including text objects associated with the text content and image objects associated with the plurality of images included in the Web page; and [5.c] for each object,	[138.b] parsing markup language code corresponding to the retrieved Web page to determine the original page layout of the content on the Web page; [138.c] logically grouping selected content into objects; [138.e.1] defining an object datum corresponding to a layout location datum for the object's associated display content; [138.e.3] creating a reference that links the object to the vector that is generated.	Appendix C (IPR2013-00004) [72.b] parsing HTML-based code corresponding to the received Web content to identify content on the Web page; [72.c] logically grouping selected content into objects; [72.e.1] defining an object datum corresponding to a layout location datum for the object's associated display content; [72.e.3] creating a reference that links the object to its corresponding vector. Appendix D (90/009,995)	Parsing content into objects including logically grouping selected content into objects, and associating each object with corresponding layout location data.

U.S. Patent No. 7,844,889	U.S. Patent No. 7,461,353	U.S. Patent No. 7,831,926	Comparison
defining an object datum corresponding to the layout location for the object on the original page layout; and [5.d] associating the object and its object datum. [46.a] The apparatus of claim 42, wherein execution of the instructions performs further operations comprising: [46.b] parsing and processing markup language code with the rendering engine to determine the original page layout of display content within the Web page as interpreted by the rendering engine, wherein the original page layout defines a layout location		[21.b] 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 [22.a] 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: [22.b] defining a primary datum corresponding to a page layout; and	

U.S. Patent No. 7,844,889	U.S. Patent No. 7,461,353	U.S. Patent No. 7,831,926	Comparison
for a plurality of objects, including text objects associated with the text content and image objects associated with the plurality of images included in the Web page; and [46.c] for each object, defining an object datum corresponding to the layout location for the object on the original page layout; and [46.d] associating the object and its object datum. [80.a] The machine-readable medium of claim 77, wherein execution of the instructions performs further operations comprising: [80.b] parsing and		[22.c] for each object, [22.c.1] defining an object datum corresponding to the layout location for the object on the page layout; [22.c.2] generating a vector from the primary datum to the object datum for the object; and [22.c.3] creating a reference that links the object to its corresponding vector. Appendix E (95/000,635) [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	

U.S. Patent No. 7,844,889	U.S. Patent No. 7,461,353	U.S. Patent No. 7,831,926	Comparison
processing markup language code with the rendering engine to determine the original page layout of display content within the Web page as interpreted by the rendering engine, wherein the original page layout defines a layout location for a plurality of objects, including text objects associated with the text content and image objects associated with the plurality of images included in the Web page; and [80.c] for each object, defining an object datum corresponding to the layout location for the object on the original page layout; and		rendered display page occupied by the object's associated group of content.	

U.S. Patent No. 7,844,889	U.S. Patent No. 7,461,353	U.S. Patent No. 7,831,926	Comparison
[80.d] associating the object and its object datum.			
[6] The method of claim 5, further comprising translating at least one image object from an original format into a scalable format. [47] The apparatus of claim 46, wherein execution of the instructions performs further operations comprising translating at least one image object from an original format into a scalable format. [81] The machine-readable medium of claim 80, wherein execution of the instructions performs further operations comprising translating at least one image object	[1.c] 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 [33] The wireless device of claim 1, wherein a portion of the scalable	Appendix C (IPR2013-00004) [30.h] 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 Appendix D (90/009,995) [1.h] translating the HTML-based Web content to	Translating or rendering image content in a scalable manner for display at different zoom levels.

U.S. Patent No. 7,844,889	U.S. Patent No. 7,461,353	U.S. Patent No. 7,831,926	Comparison
from an original format into a scalable format.	content comprises vector-based content.	produce scalable vector-based page layout information; Appendix E (95/000,635) [79.h] retrieving and processing the HTML-based Web content to produce scalable content; and	
[7.a] The method of claim 5, further comprising: [7.b] associating attribute information including a color and typefont for each text object; and [7.c] employing the attribute information to render text content corresponding to text objects shown in views of the Web page. [48.a] The apparatus of	[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. [36.d] employing at least one of the scalable content or data derived therefrom to, render the Web page on the display; and	Appendix C (IPR2013-00004) [30.f] 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; [30.i] employing the scalable vector-based	Preserving and using design attributes such as typeface and color when rendering text.

U.S. Patent No. 7,844,889	U.S. Patent No. 7,461,353	U.S. Patent No. 7,831,926	Comparison
claim 46, wherein execution of the instructions performs further operations comprising: [48.b] associating attribute information including a color and typefont for each text object; and [48.c] employing the attribute information to render text content corresponding to text objects shown in views of the Web page. [82.a] The machine-readable medium of claim 80, wherein execution of the instructions performs further operations comprising: [82.b] associating attribute information including a		content to render a view of at least a portion of the Web page on the display using a first scale factor, Appendix D (90/009,995) [24.a] The mobile device of claim 23, wherein rendering scaled content associated with a text object comprises [24.b] retrieving presentation attributes for the text object, the presentation attributes including a font typeface, size and color; [24.c] 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	

U.S. Patent No. 7,844,889	U.S. Patent No. 7,461,353	U.S. Patent No. 7,831,926	Comparison
color and typefont for each text object; and [82.c] employing the attribute information to render text content corresponding to text objects shown in views of the Web page.		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. Appendix E (95/000,635) [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. [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	

U.S. Patent No. 7,844,889	U.S. Patent No. 7,461,353	U.S. Patent No. 7,831,926	Comparison
		scalable font to render the scalable text content.	
[8.b] scaling page layout information associated with the object using a scale factor corresponding to a zoom level associated with the zoom operation to determine a scaled datum; and [8.c] scaling content associated with the object using the scale factor and rendering the scaled content at a location corresponding to the scaled datum.	[1.d] 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. [36.e] 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. [58.e.1] defining an object datum corresponding to a layout location datum for the object's associated display content;	Appendix C (IPR2013-00004) [30.i] 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, [43.c] employing the display list to re-render the display at different scale factors to enable rapid zooming of the Web page. Appendix D (90/009,995) [23.a] The mobile device of claim 22, wherein execution of the instructions performs further operations comprising effecting a	Scaling layout location information and rendering scaled content at corresponding scaled positions.

U.S. Patent No. 7,844,889	U.S. Patent No. 7,461,353	U.S. Patent No. 7,831,926	Comparison
		zoom operation combined with a pan operation by, [23.b] 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, [23.b.1] 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; [23.b.2] 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	

U.S. Patent No. 7,844,889	U.S. Patent No. 7,461,353	U.S. Patent No. 7,831,926	Comparison
		rendered on the panned view of the Web page; [23.b.3] scaling content associated with the display object using the scale factor; and [23.b.4] 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. Appendix E (95/000,635) [70.a] 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	

U.S. Patent No. 7,844,889	U.S. Patent No. 7,461,353	U.S. Patent No. 7,831,926	Comparison
		defining a user-selectable zoom level and pan corresponding to a rendered display of the Web page desired by a user; [70.b] determining a virtual display limit bounding box for the virtual display associated with the first scale factor and offset; [70.c] identifying object bounding boxes having at least a portion falling within the virtual display limit bounding box; and, [70.d] for each of such object bounding boxes, retrieving content associated with that object bounding box; and [70.e] applying an appropriate scale factor to	

U.S. Patent No. 7,844,889	U.S. Patent No. 7,461,353	U.S. Patent No. 7,831,926	Comparison
		the content to render the display.	
[9.a] The method of claim 8, further comprising effecting a zoom operation combined with a pan operation by, [9.b] determining an offset corresponding to the pan operation; and [9.c] employing the offset to at least a portion of the content that has been scaled and rendered to produce a view of the web page on the display.	[36.e] 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. [317.h] enabling the user to, zoom and pan a view of the Web page;	Appendix C (IPR2013-00004) [52.i] 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, Appendix D (90/009,995) [23.a] The mobile device of claim 22, wherein execution of the instructions performs further operations comprising effecting a	Generating zoomed and panned views by applying scale and offset to rendered content, including determining a pan offset and applying the offset to scaled, rendered content.

U.S. Patent No. 7,844,889	U.S. Patent No. 7,461,353	U.S. Patent No. 7,831,926	Comparison
		zoom operation combined with a pan operation by, [23.b] 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, [23.b.1] 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; [23.b.2] 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	

U.S. Patent No. 7,844,889	U.S. Patent No. 7,461,353	U.S. Patent No. 7,831,926	Comparison
		rendered on the panned view of the Web page; [23.b.3] scaling content associated with the display object using the scale factor; and [23.b.4] 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. Appendix E (95/000,635) [70.a] 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	

U.S. Patent No. 7,844,889	U.S. Patent No. 7,461,353	U.S. Patent No. 7,831,926	Comparison
		defining a user-selectable zoom level and pan corresponding to a rendered display of the Web page desired by a user; [70.b] determining a virtual display limit bounding box for the virtual display associated with the first scale factor and offset; [70.c] identifying object bounding boxes having at least a portion falling within the virtual display limit bounding box; and, [70.d] for each of such object bounding boxes, retrieving content associated with that object bounding box; and [70.e] applying an appropriate scale factor to	

U.S. Patent No. 7,844,889	U.S. Patent No. 7,461,353	U.S. Patent No. 7,831,926	Comparison
		the content to render the display.	
[10.a] The method of claim 8, wherein rendering scaled content associated with a text object comprises: [10.b] employing presentation attributes for the text object, the presentation attributes including a font typeface, size and color; [10.c] 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 by the operating system relative to a location associated with the scaled datum for the text object, and wherein the	[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. [36.e] 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.	Appendix C (IPR2013-00004) [52.i] 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, Appendix D (90/009,995) [24.a] The mobile device of claim 23, wherein rendering scaled content associated with a text object comprises [24.b] retrieving presentation attributes for	Rendering text using presentation attributes (including font typeface, size, and color) and employing a scalable font to render the text consistent with preserved design across zoom levels.

U.S. Patent No. 7,844,889	U.S. Patent No. 7,461,353	U.S. Patent No. 7,831,926	Comparison
scale applied to the scalable font is a function of at least one of the scale factor and the font size.		the text object, the presentation attributes including a font typeface, size and color; [24.c] 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. Appendix E (95/000,635) [29] The mobile device of claim 1, wherein the HTML-based Web content includes cascading style	

U.S. Patent No. 7,844,889	U.S. Patent No. 7,461,353	U.S. Patent No. 7,831,926	Comparison
		sheet content defining layout and presentation attributes for the Web page. [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.	
[11] The method of claim 1, further comprising enabling the Web content to be displayed at different resolutions by scaling scalable page layout information to determine corresponding layout locations for the Web content on at least a portion of a scaled Web page for each of a plurality of scale	[1.d] 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. [36.e] 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	Appendix C (IPR2013-00004) [43.c] employing the display list to re-render the display at different scale factors to enable rapid zooming of the Web page. Appendix D (90/009,995) [7] The mobile device of claim 1, wherein execution	Generating views at multiple scale factors using layout information to preserve relative positioning by scaling layout location information for objects and re-rendering at different zoom levels.

U.S. Patent No. 7,844,889	U.S. Patent No. 7,461,353	U.S. Patent No. 7,831,926	Comparison
factors. [49] The apparatus of claim 42, wherein execution of the instructions performs further operations comprising enabling the Web content to be displayed at different resolutions by scaling the page layout information to determine corresponding layout locations for the Web content on at least a portion of a scaled Web page for each of a plurality of scale factors. [83] The machine-readable medium of claim 77, wherein execution of the instructions performs further operations comprising enabling the Web content to be displayed at different resolutions by scaling the	while preserving the original page layout, functionality, and design of the Web page content at each zoom level and panned view.	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. Appendix E (95/000,635) [71.a] The mobile device of claim 52, further comprising dynamic memory having at least a portion employed for rendering purposes, [71.b] wherein execution of the instructions performs further operations comprising building a display list via use of the scalable content and	

U.S. Patent No. 7,844,889	U.S. Patent No. 7,461,353	U.S. Patent No. 7,831,926	Comparison
scalable page layout information to determine corresponding layout locations for the Web content on at least a portion of a scaled Web page for each of a plurality of scale factors.		rendering the display list on a virtual display area in the dynamic memory; and [71.c] scaling the display list to re-render the display of the Web page.	
[12.a] The method of claim 1, further comprising: [12.b] enabling a user of the wireless hand-held device to select an area of the display to zoom in on via the display; and [12.c] in response thereto, re-rendering the display such that content in the area that is selected is enlarged. [50.a] The apparatus of claim 42, wherein execution of the	[36.e] 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. [317.h] enabling the user to, zoom and pan a view of the Web page;	Appendix C (IPR2013-00004) [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. [52.i] re-render the Web page in response to	Selecting a user-chosen portion for zooming and re-rendering the selected content enlarged.

U.S. Patent No. 7,844,889	U.S. Patent No. 7,461,353	U.S. Patent No. 7,831,926	Comparison
instructions performs further operations comprising: [50.b] enabling a user of the apparatus to select an area of a view to zoom in on via the apparatus; and [50.c] in response thereto, regenerating the view such that content in the area that is selected is enlarged. [84.a] The machine-readable medium of claim 77, wherein execution of the instructions performs further operations comprising: [84.b] enabling a user of the wireless hand-held device to select an object on the display to zoom in on via the display; and		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, Appendix D (90/009,995) [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	

U.S. Patent No. 7,844,889	U.S. Patent No. 7,461,353	U.S. Patent No. 7,831,926	Comparison
[84.c] in response thereto, re-rendering the display such that content associated with the object is rendered to fit across a width of the display.		display of the Web page is re-rendered in real-time to effect zooming operations. Appendix E (95/000,635) [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.	
[13] The method of claim 1, further comprising enabling a user to pan views of the Web page in response to a corresponding user input. [51] The apparatus of claim	[36.e] 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,	Appendix C (IPR2013-00004) [52.i] re-render the Web page in response to associated user inputs to enable the user to iteratively zoom in and out	Panning views in response to user input during interactive browsing.

U.S. Patent No. 7,844,889	U.S. Patent No. 7,461,353	U.S. Patent No. 7,831,926	Comparison
42, 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 input. [86] The machine-readable medium of claim 77, 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. [114.a] The method of claim 108, further comprising: [114.b] in response to a user input to pan a current view of the Web page to a new view displaying a portion of the Web page	functionality, and design of the Web page content at each zoom level and panned view. [317.h] enabling the user to, zoom and pan a view of the Web page;	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, Appendix D (90/009,995) [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. Appendix E (95/000,635) [61] The mobile device of claim 52, wherein execution of the instructions performs	

U.S. Patent No. 7,844,889	U.S. Patent No. 7,461,353	U.S. Patent No. 7,831,926	Comparison
that is entirely not shown in the current view,		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.	
[14] The method of claim 13, further comprising enabling the views of the Web page to be panned in real-time. [52] The apparatus of claim 51, wherein execution of the instructions performs further operations comprising enabling the view of the Web content to be panned in real-time. [87] The machine-readable medium of claim 86, wherein execution of the instructions performs further operations	[36.e] 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.	Appendix C (IPR2013-00004) [43.c] employing the display list to re-render the display at different scale factors to enable rapid zooming of the Web page. Appendix D (90/009,995) [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.	Panning behavior during navigation across views.

U.S. Patent No. 7,844,889	U.S. Patent No. 7,461,353	U.S. Patent No. 7,831,926	Comparison
comprising enabling the view of the Web page to be panned in real-time. [114.c] generating scaled Web page content in real-time. [115] The method of claim 114, wherein the scaled Web page content is generated in a fraction of a second.		Appendix E (95/000,635) [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.	
[15] The method of claim 1, further comprising enabling a user to zoom on a column of the Web content via an associated user input, wherein in response to the user input, the display is re-rendered such that content corresponding to the selected column is displayed across the display.	[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	Appendix C (IPR2013-00004) [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	Tap-to-zoom on a column and re-rendering so the column fits across the display.

U.S. Patent No. 7,844,889	U.S. Patent No. 7,461,353	U.S. Patent No. 7,831,926	Comparison
[16] The method of claim 15, wherein the user input comprises tapping the column. [53] The apparatus of claim 42, wherein execution of the instructions performs further operations comprising enabling a user to zoom on a column of the Web content via an associated user input, wherein in response to the user input, the view is re-rendered such that content corresponding to the selected column is displayed across the display. [54] The apparatus of claim 53, wherein the user input comprises tapping the column.	selected column is rendered to fit across the display. [317.k] zoom in on a column of the Web page by tapping on the column via the display;	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. Appendix D (90/009,995) [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	

U.S. Patent No. 7,844,889	U.S. Patent No. 7,461,353	U.S. Patent No. 7,831,926	Comparison
[88] The machine-readable medium of claim 77, wherein execution of the instructions performs further operations comprising enabling a user to zoom on a column of the Web content via an associated user input, wherein in response to the user input, the view is re-rendered such that content corresponding to the selected column is displayed across the display. [89] The machine-readable medium of claim 88, wherein the user input comprises tapping the column.		displayed to fit across the touch-sensitive display. Appendix E (95/000,635) [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.	
[17] The method of claim 1, further comprising enabling a user to zoom on an image via an associated	[51] The mobile hand-held device of claim 36, wherein the Web content includes at least one	Appendix C (IPR2013-00004)	Tap-to-zoom on an image and re-rendering

U.S. Patent No. 7,844,889	U.S. Patent No. 7,461,353	U.S. Patent No. 7,831,926	Comparison
user input, wherein in response to the user input, the view is re-rendered such that the image is displayed to fit across a width of the display. [18] The method of claim 17, wherein the user input comprises tapping the image. [55] The apparatus of claim 42, wherein execution of the instructions performs further operations comprising enabling a user to zoom on an image via an associated user input, wherein in response to the user input, the display is re-rendered such that the image is displayed to fit across a width of the display. [56] The apparatus of claim	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.	[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. Appendix D (90/009,995) [13] The mobile device of claim 1, wherein the Web content includes at least one image, and wherein	so the image fits across the display.

U.S. Patent No. 7,844,889	U.S. Patent No. 7,461,353	U.S. Patent No. 7,831,926	Comparison
55, wherein the user input comprises tapping the image. [90] The machine-readable medium of claim 77, wherein execution of the instructions performs further operations comprising enabling a user to zoom on an image, wherein in response to the user input, the view is re-rendered such that the image is displayed to fit across at least one of a width and height of the display. [91] The machine-readable medium of claim 90, wherein the user input comprises tapping the image.		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. Appendix E (95/000,635) [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	

U.S. Patent No. 7,844,889	U.S. Patent No. 7,461,353	U.S. Patent No. 7,831,926	Comparison
		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.	
[19] The method of claim 1, further comprising enabling a user to zoom on a paragraph of the Web content via an associated user input, wherein in response to the user input, the view is re-rendered such that content corresponding to the selected paragraph is displayed to fit across the display. [20] The method of claim 19, wherein the user input	[36.e] 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. [317.k] zoom in on a column of the Web page by tapping on the column via the display; and	Appendix C (IPR2013-00004) [52.i] 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,	Zooming on a selected portion and re-rendering it enlarged to fit across the display, including any reformatting.

U.S. Patent No. 7,844,889	U.S. Patent No. 7,461,353	U.S. Patent No. 7,831,926	Comparison
comprises tapping the paragraph. [21] The method of claim 19, wherein the content of the paragraph is reformatted when the view is re-rendered. [57] The apparatus of claim 42, wherein execution of the instructions performs further operations comprising enabling a user to zoom on a paragraph of the Web content via an associated user input, wherein in response to the user input, the view is re-rendered such that content corresponding to the selected paragraph is displayed across the display. [58] The apparatus of claim 57, wherein the user input		[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. Appendix D (90/009,995) [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	

U.S. Patent No. 7,844,889	U.S. Patent No. 7,461,353	U.S. Patent No. 7,831,926	Comparison
comprises tapping the paragraph. [59] The apparatus of claim 57, wherein the content of the paragraph is reformatted when the view is re-rendered. [68] The apparatus of claim 42, wherein execution of the instructions performs further operations comprising reformatting a text object so as to fit the text object on the display when it is rendered. [92] The machine-readable medium of claim 77, wherein execution of the instructions performs further operations comprising enabling a user to zoom on a paragraph of the Web content via an associated user input,		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. Appendix E (95/000,635) [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	

U.S. Patent No. 7,844,889	U.S. Patent No. 7,461,353	U.S. Patent No. 7,831,926	Comparison
wherein in response to the user input, the view is re-rendered such that content corresponding to the selected paragraph is displayed across the display. [93] The machine-readable medium of claim 92, wherein the user input comprises tapping the paragraph. [94] The machine-readable medium of claim 92, wherein the content of the paragraph is reformatted when the view is re-rendered.		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.	
[22.a] The method of claim 1, further comprising: [22.b] generating a display list derived, at least in part, via use of the scalable page	[58.c] logically grouping selected content into objects; [36.e] re-render the display in response to associated user inputs to	Appendix C (IPR2013-00004) [43.b] generating a display list associated with the	Generating and using a display list to re-render views at different scale factors to enable rapid zooming.

U.S. Patent No. 7,844,889	U.S. Patent No. 7,461,353	U.S. Patent No. 7,831,926	Comparison
layout information; and [22.c] employing the display list to generate views of the Web page. [60.a] The apparatus of claim 42, wherein execution of the instructions performs further operations comprising: [60.b] generating a display list derived, at least in part, via use of the scalable page layout information; and [60.c] employing the display list to generate views of the Web page. [95.a] The machine-readable medium of claim 77, wherein execution of the instructions performs further operations	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.	scalable vector-based content; and [43.c] employing the display list to re-render the display at different scale factors to enable rapid zooming of the Web page. Appendix D (90/009,995) [15.b] generating a display list derived, at least in part, via use of the vector-based page layout information; and [15.c] employing the display list to re-render the display of the Web page. Appendix E (95/000,635) [71.a] The mobile device of claim 52, further comprising dynamic memory having at least a	

U.S. Patent No. 7,844,889	U.S. Patent No. 7,461,353	U.S. Patent No. 7,831,926	Comparison
comprising: [95.b] generating a display list; and [95.c] employing the display list to generate views of the Web page.		portion employed for rendering purposes, [71.b] 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 [71.c] scaling the display list to re-render the display of the Web page.	
[23.a] The method of claim 1, further comprising: [23.b] parsing and processing, with the rendering engine, markup language code corresponding to the received Web content to determine page layout information corresponding	[58.b] parsing HTML-based code corresponding to the retrieved Web page to determine the original page layout of the content on the Web page; [58.c] logically grouping selected content into objects;	Appendix C (IPR2013-00004) [72.b] parsing HTML-based code corresponding to the received Web content to identify content on the Web page;	Per-object layout information identifying coordinate locations and associating each object with that information.

U.S. Patent No. 7,844,889	U.S. Patent No. 7,461,353	U.S. Patent No. 7,831,926	Comparison
to an original page layout of the content on the Web page as interpreted by the rendering engine, the content including a plurality of objects comprising text objects associated with the text content and image objects associated with the plurality of images, wherein for each object the layout information includes object layout information identifying a coordinate location of the object on the page layout; and [23.c] associating each object and its object layout information. [61.a] The apparatus of claim 42, wherein execution of the instructions performs further operations	[58.e.1] defining an object datum corresponding to a layout location datum for the object's associated display content; [58.e.3] creating a reference that links the object to the vector that is generated.	[72.c] logically grouping selected content into objects; [72.e.1] defining an object datum corresponding to a layout location datum for the object's associated display content; [72.e.3] creating a reference that links the object to its corresponding vector. Appendix D (90/009,995) [16.b] parsing HTML-based code corresponding to the received Web content to logically group content into objects, the objects including a plurality of display objects; [16.d] for each display object, defining an object datum corresponding to a	

U.S. Patent No. 7,844,889	U.S. Patent No. 7,461,353	U.S. Patent No. 7,831,926	Comparison
comprising: [61.b] parsing and processing, with the rendering engine, markup language code corresponding to the received Web content to determine page layout information corresponding to an original page layout of the content on the Web page as interpreted by the rendering engine, the content including a plurality of objects comprising text objects associated with the text content and image objects associated with the plurality of images, wherein for each object the layout information includes object layout information identifying a coordinate location of the object on the page layout; and		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. [18.d] storing information that links each object with its corresponding page layout information; Appendix E (95/000,635) [70.a] 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	

U.S. Patent No. 7,844,889	U.S. Patent No. 7,461,353	U.S. Patent No. 7,831,926	Comparison
[61.c] associating each object and its object layout information. [96.a] The machine-readable medium of claim 77, wherein execution of the instructions performs further operations comprising: [96.b] parsing and processing, with the rendering engine, markup language code corresponding to the received Web content to determine page layout information corresponding to an original page layout of the content on the Web page as interpreted by the rendering engine, the content including a plurality of objects comprising text objects		zoom level and pan corresponding to a rendered display of the Web page desired by a user; [70.b] determining a virtual display limit bounding box for the virtual display associated with the first scale factor and offset; [70.c] identifying object bounding boxes having at least a portion falling within the virtual display limit bounding box; and, [70.d] for each of such object bounding boxes, retrieving content associated with that object bounding box; and [70.e] applying an appropriate scale factor to	

U.S. Patent No. 7,844,889	U.S. Patent No. 7,461,353	U.S. Patent No. 7,831,926	Comparison
associated with the text content and image objects associated with the plurality of images, wherein for each object the layout information includes object layout information identifying a coordinate location of the object on the page layout; and [96.c] associating each object and its object layout information.		the content to render the display. [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.	
[24] The method of claim 23 further comprising: generating page layout information including an object bounding box for each object. [62] The apparatus of claim 61 wherein execution of the instructions perform further operations	[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	Appendix C (IPR2013-00004) [72.c] logically grouping selected content into objects; Appendix D (90/009,995) [18.c] generating page layout information including a bounding box	Generating per-object bounding boxes and using bounding-box information in rendering and layout to support object based layout and rendering decisions.

U.S. Patent No. 7,844,889	U.S. Patent No. 7,461,353	U.S. Patent No. 7,831,926	Comparison
comprising: generating page layout information including an object bounding box for each object, the object bounding box representing a portion of the page layout occupied by the object's associated content. [97] The machine-readable medium of claim 96 wherein execution of the instructions perform further operations comprising: generating page layout information including an object bounding box for each object, the object bounding box representing a portion of the page layout occupied by the object's associated content.	associated group of content.	for each object, the bounding box defining width and height dimensions for the object; and Appendix E (95/000,635) [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.	

U.S. Patent No. 7,844,889	U.S. Patent No. 7,461,353	U.S. Patent No. 7,831,926	Comparison
[25] The method of claim 23, further comprising: for each of the plurality of objects, mapping the coordinate location associated with the object layout information for the object to a corresponding object datum coordinate on a virtual display area employing a resolution-independent coordinate system. [63] The apparatus of claim 61, wherein execution of the instructions perform further operations comprising: for each of the plurality of objects, mapping the coordinate location associated with the object layout information for the object to a corresponding object datum coordinate on a	[58.e.2] generating a vector from the primary datum to the object datum for the object; and	Appendix C (IPR2013-00004) [72.e.2] generating a vector from the primary datum to the object datum for the object; and Appendix D (90/009,995) [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. Appendix E (95/000,635) [71.a] The mobile device of claim 52, further comprising dynamic memory having at least a	Mapping object locations using primary and object data in a resolution-independent coordinate system.

U.S. Patent No. 7,844,889	U.S. Patent No. 7,461,353	U.S. Patent No. 7,831,926	Comparison
virtual display area employing a resolution-independent coordinate system. [98] The machine-readable medium of claim 96, wherein execution of the instructions perform further operations comprising: for each of the plurality of objects, mapping the coordinate location associated with the object layout information for the object to a corresponding object datum coordinate on a virtual display area employing a resolution-independent coordinate system.		portion employed for rendering purposes, [71.b] 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 [71.c] scaling the display list to re-render the display of the Web page.	
[26.a] The method of claim 25, further comprising: [26.b] determining a first	[36.e] re-render the display in response to associated user inputs to enable the Web page to be	Appendix C (IPR2013-00004)	Determining a scale factor and pan offset, defining a virtual-display bounding box

U.S. Patent No. 7,844,889	U.S. Patent No. 7,461,353	U.S. Patent No. 7,831,926	Comparison
scale factor and offset in response to one or more corresponding user inputs defining a user-selectable zoom level and pan position corresponding to a view of the Web content desired by a user; [26.c] determining a virtual display bounding box for the virtual display area associated with the first scale factor and offset; [26.d] identifying objects having at least a portion falling within the virtual display bounding box; and [26.e] rendering a view of the Web page corresponding to the zoom level and pan position on the display including the at least a portion of the objects falling within the	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. [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. [317.h] enabling the user to, zoom and pan a view of the Web page;	[52.i] 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, Appendix D (90/009,995) [23.a] 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, [23.b] for each of the plurality of display objects to be included in a panned	for the resulting view, identifying objects at least partly within the bounding box, and rendering the view accordingly.

U.S. Patent No. 7,844,889	U.S. Patent No. 7,461,353	U.S. Patent No. 7,831,926	Comparison
virtual display bounding box. [64.a] The apparatus of claim 63, wherein execution of the instructions perform further operations comprising: [64.b] determining a first scale factor and offset in response to one or more corresponding user inputs defining a user-selectable zoom level and pan position corresponding to a view of the Web content desired by a user; [64.c] determining a virtual display bounding box for the virtual display area associated with the first scale factor and offset; [64.d] identifying objects		view of the Web page to be rendered on the display, [23.b.1] 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; [23.b.2] 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; [23.b.3] scaling content associated with the display object using the scale factor; and	

U.S. Patent No. 7,844,889	U.S. Patent No. 7,461,353	U.S. Patent No. 7,831,926	Comparison
having at least a portion falling within the virtual display bounding box; and [64.e] rendering a view of the Web page corresponding to the zoom level and pan position on the display including the at least a portion of the objects falling within the virtual display bounding box. [99.a] The machine-readable medium of claim 98, wherein execution of the instructions perform further operations comprising: [99.b] determining a first scale factor and offset in response to one or more corresponding user inputs defining a user-selectable zoom level and pan		[23.b.4] 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. Appendix E (95/000,635) [70.a] 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; [70.b] determining a virtual display limit bounding box	

U.S. Patent No. 7,844,889	U.S. Patent No. 7,461,353	U.S. Patent No. 7,831,926	Comparison
position corresponding to a view of the Web content desired by a user; [99.c] determining a virtual display bounding box for the virtual display area associated with the first scale factor and offset; [99.d] identifying objects having at least a portion falling within the virtual display bounding box; and [99.e] applying an applicable scale factor to the object's content to render a view corresponding to the zoom level and pan position on the display.		for the virtual display associated with the first scale factor and offset; [70.c] identifying object bounding boxes having at least a portion falling within the virtual display limit bounding box; and, [70.d] for each of such object bounding boxes, retrieving content associated with that object bounding box; and [70.e] applying an appropriate scale factor to the content to render the display.	
[27] The method of claim 25, wherein the virtual display area is associated with a fixed reference	[58.d] defining a primary datum corresponding to the original page layout; and,	Appendix C (IPR2013-00004)	Using a fixed reference frame (primary datum) for a virtual display area / coordinate system to

U.S. Patent No. 7,844,889	U.S. Patent No. 7,461,353	U.S. Patent No. 7,831,926	Comparison
frame. [65] The apparatus of claim 63, wherein the virtual display area is associated with a fixed reference frame. [100] The machine-readable medium of claim 98, wherein the virtual display area is associated with a fixed reference frame.	[58.e.2] generating a vector from the primary datum to the object datum for the object; and	[72.d] defining a primary datum corresponding to the original page layout; and, Appendix D (90/009,995) [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	support consistent layout location mapping across zoom levels.

U.S. Patent No. 7,844,889	U.S. Patent No. 7,461,353	U.S. Patent No. 7,831,926	Comparison
		mathematical transformation. Appendix E (95/000,635) [71.a] The mobile device of claim 52, further comprising dynamic memory having at least a portion employed for rendering purposes, [71.b] 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 [71.c] scaling the display list to re-render the display of the Web page.	

U.S. Patent No. 7,844,889	U.S. Patent No. 7,461,353	U.S. Patent No. 7,831,926	Comparison
[28.a] The method of claim 1, wherein the content includes a plurality of text objects having associated text content, the method further comprising: [28.b] for at least a portion of the plurality of text objects, scaling a scalable font to render the text content associated with the text object. [66.a] The apparatus of claim 42, wherein the content includes a plurality of text objects having associated text content, and wherein execution of the instructions performs further operations comprising: [66.b] for at least a portion of the plurality of text	[36.c] retrieving the Web page via the wireless communications device, and processing HTML-based Web content to produce scalable content; and [36.e] 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. [33] The wireless device of claim 1, wherein a portion of the scalable content comprises vector-based content.	Appendix C (IPR2013-00004) [30.f] 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; [30.h] 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	Zoom-dependent text rendering by (a) scaling a scalable font and/or (b) selecting a size-appropriate font based on an applicable scale factor, while preserving original text design attributes (for example, CSS-defined typeface/size/color where applicable).

U.S. Patent No. 7,844,889	U.S. Patent No. 7,461,353	U.S. Patent No. 7,831,926	Comparison
objects, scaling a scalable font to render the text content associated with the text object. [101.a] The machine-readable medium of claim 77, wherein the content includes a plurality of text objects having associated text content, and wherein execution of the instructions performs further operations comprising: [101.b] for at least a portion of the plurality of text objects, scaling a scalable font to render the text content associated with the text object. [29.a] The method of claim 1, wherein the content includes a plurality of text objects having associated	[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.	portion of the HTML-based content when scaled and rendered; and Appendix D (90/009,995) [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. [24.a] The mobile device of claim 23, wherein rendering scaled content associated with a text object comprises [24.b] retrieving presentation attributes for the text object, the presentation attributes	

U.S. Patent No. 7,844,889	U.S. Patent No. 7,461,353	U.S. Patent No. 7,831,926	Comparison
text content, the method further comprising: [29.b] for at least a portion of the plurality of text objects, determining an applicable font size based on an applicable scale factor associated with a zoom level; and [29.c] selecting a corresponding bitmapped font and employing the bitmapped font to render the text content associated with the text object. [67.a] The apparatus of claim 42, wherein the content includes a plurality of text objects having associated text content, and wherein execution of the instructions performs further operations comprising:		including a font typeface, size and color; [24.c] 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. Appendix E (95/000,635) [74] The mobile device of claim 52, wherein the scalable content includes scalable text content, and wherein execution of the instructions performs	

U.S. Patent No. 7,844,889	U.S. Patent No. 7,461,353	U.S. Patent No. 7,831,926	Comparison
[67.b] for at least a portion of the plurality of text objects, determining an applicable font size based on an applicable scale factor associated with a zoom level; and [67.c] selecting a corresponding bitmapped font and employing the bitmapped font to render the text content associated with the text object. [102.a] The machine-readable medium of claim 77, wherein the content includes a plurality of text objects having associated text content, and wherein execution of the instructions performs further operations comprising:		further operations comprising scaling a scalable font to render the scalable text content.	

U.S. Patent No. 7,844,889	U.S. Patent No. 7,461,353	U.S. Patent No. 7,831,926	Comparison
[102.b] for at least a portion of the plurality of text objects, determining an applicable font size based on an applicable scale factor associated with a zoom level; and [102.c] selecting a corresponding bitmapped font and employing the bitmapped font to render the text content associated with the text object.			
[32] The method of claim 1, wherein in a view at least one image is scaled. [70] The apparatus of claim 42, wherein in a generated view at least one image from is scaled. [106] The machine-readable medium of claim 77, wherein at least one	[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	Appendix C (IPR2013-00004) [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	Scaling images for rendering in zoomed views.

U.S. Patent No. 7,844,889	U.S. Patent No. 7,461,353	U.S. Patent No. 7,831,926	Comparison
view at least one image is scaled.	thereto, the display is re-rendered such that the image is rendered to fit across the display.	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. Appendix D (90/009,995) [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,	

U.S. Patent No. 7,844,889	U.S. Patent No. 7,461,353	U.S. Patent No. 7,831,926	Comparison
		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. Appendix E (95/000,635) [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	

U.S. Patent No. 7,844,889	U.S. Patent No. 7,461,353	U.S. Patent No. 7,831,926	Comparison
		display area of the touch-sensitive display.	
[33] The method of claim 1, further comprising maintaining at least one instance of the page layout information in a manner that is independent of zoom levels used to view the web page	[252.e] 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,	Appendix C (IPR2013-00004) [52.i] 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, [43.b] generating a display list associated with the scalable vector-based content; and [43.c] employing the display list to re-render the display at different scale factors to enable rapid zooming of the Web page.	Resolution-independence and scalability and re-rendering across multiple zoom levels.

U.S. Patent No. 7,844,889	U.S. Patent No. 7,461,353	U.S. Patent No. 7,831,926	Comparison
		Appendix D (90/009,995) [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. Appendix E (95/000,635) [71.a] The mobile device of claim 52, further comprising dynamic memory having at least a portion employed for rendering purposes, [71.b] wherein execution of the instructions performs further operations comprising building a display list via use of the scalable content and	

U.S. Patent No. 7,844,889	U.S. Patent No. 7,461,353	U.S. Patent No. 7,831,926	Comparison
		rendering the display list on a virtual display area in the dynamic memory; and [71.c] scaling the display list to re-render the display of the Web page.	
[34] The method of claim 1, wherein the scalable page layout information comprises scalable vector-based page layout information.	[33] The wireless device of claim 1, wherein a portion of the scalable content comprises vector-based content. [43] The mobile hand-held device of claim 36, wherein at least a portion of the scalable content comprises scalable vector-based content.	Appendix C (IPR2013-00004) [30.h] 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	Scalable vector-based content supporting resolution-independent rendering using one or more vectors and capable of being rendered at multiple zoom levels.

U.S. Patent No. 7,844,889	U.S. Patent No. 7,461,353	U.S. Patent No. 7,831,926	Comparison
		Appendix D (90/009,995) [1.h] translating the HTML-based Web content to produce scalable vector-based page layout information; Appendix E (95/000,635) [79.h] retrieving and processing the HTML-based Web content to produce scalable content; and	
[35.a] The method of claim 34, wherein scaling operations are effected by applying a mathematical transformation to a plurality of points in a two-dimensional coordinate system comprising X and Y axes, [35.b] including points comprising datum points	[43] The mobile hand-held device of claim 36, wherein at least a portion of the scalable content comprises scalable vector-based content. [58.e.2] generating a vector from the primary datum to the object datum for the object; and	Appendix C (IPR2013-00004) [72.e.2] generating a vector from the primary datum to the object datum for the object; and Appendix D (90/009,995) [26.a] The mobile device of claim 1, wherein zooming	Applying a mathematical transformation using a scale factor to vector-defined page layout locations in a two-dimensional coordinate system.

U.S. Patent No. 7,844,889	U.S. Patent No. 7,461,353	U.S. Patent No. 7,831,926	Comparison
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, [35.c] wherein the mathematical transformation comprises, $X'=XSF$; $Y'=YSF$; 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.		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 [26.b] $X'=XSF$; [26.c] $Y'=YSF$; [26.d] wherein X, Y is the location of a point prior to transformation, X', Y' is the	

U.S. Patent No. 7,844,889	U.S. Patent No. 7,461,353	U.S. Patent No. 7,831,926	Comparison
		location of the point after transformation, and SF is the scale factor. Appendix E (95/000,635) [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.	
[36.a] The method of claim 35, wherein the mathematical transformation is applied to points in a first coordinate system comprising a resolution independent coordinate system associated with a virtual display area onto which page layout information is	[1.c] 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	Appendix C (IPR2013-00004) [30.h] 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-	Mapping a resolution-independent (virtual display area) coordinate system to device pixel coordinates when scaling and rendering views at different zoom levels.

U.S. Patent No. 7,844,889	U.S. Patent No. 7,461,353	U.S. Patent No. 7,831,926	Comparison
mapped [36.b] to a second coordinate system comprising a device coordinate system corresponding to a pixel resolution of the display, wherein points are mapped from the first coordinate system to the second coordinate system using the mathematical transformation.	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	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 Appendix D (90/009,995) [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	

U.S. Patent No. 7,844,889	U.S. Patent No. 7,461,353	U.S. Patent No. 7,831,926	Comparison
		the mobile device, wherein points are mapped from the first coordinate system to the second coordinate system using the mathematical transformation. Appendix E (95/000,635) [71.a] The mobile device of claim 52, further comprising dynamic memory having at least a portion employed for rendering purposes, [71.b] 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	

U.S. Patent No. 7,844,889	U.S. Patent No. 7,461,353	U.S. Patent No. 7,831,926	Comparison
		[71.c] scaling the display list to re-render the display of the Web page.	
[37] The method of claim 1, wherein the markup language-based Web content includes cascading style sheet content defining layout and presentation attributes for the Web page.	[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.	Appendix C (IPR2013-00004) [30.f] 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; Appendix D (90/009,995) [1.f] 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;	CSS-defined layout or presentation attributes used in rendering and preserved across views.

U.S. Patent No. 7,844,889	U.S. Patent No. 7,461,353	U.S. Patent No. 7,831,926	Comparison
		Appendix E (95/000,635) [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.	
[38] The method of claim 1, wherein the markup language-based Web content comprises HTML-based Web content that is processed by the rendering engine to generate page layout information corresponding to the original page layout as interpreted by the rendering engine. [73] The apparatus of claim 42, wherein the markup language-based Web content comprises HTML-	[58.b] parsing HTML-based code corresponding to the retrieved Web page to determine the original page layout of the content on the Web page;	Appendix C (IPR2013-00004) [52.i] 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, Appendix D (90/009,995)	Processing HTML-based content to generate page layout information as interpreted by a rendering engine.

U.S. Patent No. 7,844,889	U.S. Patent No. 7,461,353	U.S. Patent No. 7,831,926	Comparison
based Web content that is processed by the rendering engine to generate page layout information corresponding to the original page layout as interpreted by the rendering engine.		[21.b] 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 Appendix E (95/000,635) [79.i.2] 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,	

U.S. Patent No. 7,844,889	U.S. Patent No. 7,461,353	U.S. Patent No. 7,831,926	Comparison
[39.a] The method of claim 1, wherein the Web page content includes a plurality of images, the method further comprising: [39.b] generating page layout information including a bounding box for each of the plurality of images; [39.c] scaling the bounding box associated with an image to produce a scaled bounding box; [39.d] rendering bitmap content associated with the image to fit within the scaled bounding box. [109.a] The method of claim 108, wherein the page layout information that is generated includes a	[1.d] 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. [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. [59] The mobile hand-held device of claim 58, wherein execution of the instructions performs	Appendix C (IPR2013-00004) [30.i] 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, [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	For images, generating a bounding box per image, scaling the bounding box to produce a scaled bounding box, and scaling image content to fit within its corresponding scaled bounding box.

U.S. Patent No. 7,844,889	U.S. Patent No. 7,461,353	U.S. Patent No. 7,831,926	Comparison
bounding box for each of the plurality of images, and [109.b] wherein scaled Web page content including one or more images is generated by scaling bounding boxes corresponding to the one or more images to produce scaled bounding boxes and, [109.c] for each of the one or more images, scaling content associated with the image to produce a scaled image that fits within its corresponding scaled bounding box.	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.	least one of a width and height of a display area of the touch-sensitive display. Appendix D (90/009,995) [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.	

U.S. Patent No. 7,844,889	U.S. Patent No. 7,461,353	U.S. Patent No. 7,831,926	Comparison
		[18.c] generating page layout information including a bounding box for each object, the bounding box defining width and height dimensions for the object; and [23.a] 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	

U.S. Patent No. 7,844,889	U.S. Patent No. 7,461,353	U.S. Patent No. 7,831,926	Comparison
		with the zoom operation to determine a scaled datum; 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. Appendix E (95/000,635) [65] The mobile device of claim 52, wherein the Web content includes at least one	

U.S. Patent No. 7,844,889	U.S. Patent No. 7,461,353	U.S. Patent No. 7,831,926	Comparison
		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. [70.a] 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	

U.S. Patent No. 7,844,889	U.S. Patent No. 7,461,353	U.S. Patent No. 7,831,926	Comparison
		zoom level and pan corresponding to a rendered display of the Web page desired by a user; [70.b] determining a virtual display limit bounding box for the virtual display associated with the first scale factor and offset; [70.c] identifying object bounding boxes having at least a portion falling within the virtual display limit bounding box; and, [70.d] for each of such object bounding boxes, retrieving content associated with that object bounding box; and [70.e] applying an appropriate scale factor to	

U.S. Patent No. 7,844,889	U.S. Patent No. 7,461,353	U.S. Patent No. 7,831,926	Comparison
		the content to render the display. [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.	
[40] The method of claim 1, wherein text content in a view is rendered to have a resolution that is approximately proportional to the zoom level of the view, such that as the zoom level increases the resolution of text content increases approximately proportionally.	[36.e] 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	Appendix C (IPR2013-00004) [52.i] 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	Rendering text at a resolution that increases with zoom level, consistent with scaling and re-rendering at different scale factors

U.S. Patent No. 7,844,889	U.S. Patent No. 7,461,353	U.S. Patent No. 7,831,926	Comparison
[75] The apparatus of claim 42, wherein text content in a view is rendered to have a resolution that is approximately proportional to the zoom level of the view, such that as the zoom level increases the resolution of text content increases approximately proportionally. [107] The machine-readable medium of claim 77, wherein text content in a view is rendered to have a resolution that is approximately proportional to the zoom level of the view, such that as the zoom level increases the resolution of text content increases approximately proportionally.	each zoom level and panned view.	HTML-based Web content as interpreted by a rendering engine, Appendix D (90/009,995) [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. Appendix E (95/000,635) [74] The mobile device of claim 52, wherein the scalable content includes scalable text content, and wherein execution of the instructions performs further operations	

U.S. Patent No. 7,844,889	U.S. Patent No. 7,461,353	U.S. Patent No. 7,831,926	Comparison
		comprising scaling a scalable font to render the scalable text content.	
[41] The method of claim 1, wherein the layout of content for views associated with a zoom level is effected by scaling page layout information using a scale factor associated with the zoom level. [76] The apparatus of claim 42, wherein the layout of content for views associated with a zoom level is effected by scaling page layout information using a scale factor associated with the zoom level.	[36.e] 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.	Appendix C (IPR2013-00004) [43.c] employing the display list to re-render the display at different scale factors to enable rapid zooming of the Web page. Appendix D (90/009,995) [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	Applying a scale factor to layout information to generate views at a zoom level while preserving layout.

U.S. Patent No. 7,844,889	U.S. Patent No. 7,461,353	U.S. Patent No. 7,831,926	Comparison
		the display in response to associated user inputs. Appendix E (95/000,635) [70.a] 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; [70.b] determining a virtual display limit bounding box for the virtual display associated with the first scale factor and offset;	

U.S. Patent No. 7,844,889	U.S. Patent No. 7,461,353	U.S. Patent No. 7,831,926	Comparison
		[70.c] identifying object bounding boxes having at least a portion falling within the virtual display limit bounding box; and, [70.d] for each of such object bounding boxes, retrieving content associated with that object bounding box; and [70.e] applying an appropriate scale factor to the content to render the display.	
[44] The apparatus of claim 42, wherein the hand-held mobile device comprises a mobile phone. [74] The apparatus of claim 42, wherein the network comprises a mobile service provider network and the	[36.pre] A mobile hand-held device, comprising:	Appendix C (IPR2013-00004) [30.pre] A mobile phone, comprising: Appendix D (90/009,995)	The hand-held mobile device is a mobile phone, and communications are via a mobile service provider network.

U.S. Patent No. 7,844,889	U.S. Patent No. 7,461,353	U.S. Patent No. 7,831,926	Comparison
hand-held mobile device comprises a mobile phone.		[2] The mobile device of claim 1, wherein the device comprises a mobile phone. Appendix E (95/000,635) [53] The mobile device of claim 52, wherein the device comprises a mobile phone.	
[69] The apparatus of claim 42, wherein the Web page includes text content and graphic layout objects, and views are generated through use of scaled text content and graphic layout objects. [105] The machine-readable medium of claim 77, wherein the Web page includes text content and graphic layout objects, and in generated views the text	[58.c] logically grouping selected content into objects;	Appendix C (IPR2013-00004) [72.c] logically grouping selected content into objects; Appendix D (90/009,995) [22.a] 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	Generating views through use of scaled text content and graphic layout objects by scaling content and layout objects as zoom level changes.

U.S. Patent No. 7,844,889	U.S. Patent No. 7,461,353	U.S. Patent No. 7,831,926	Comparison
content and graphic layout objects are scaled.		included on the Web page, and wherein producing vector-based page layout information comprises: Appendix E (95/000,635) [68.a] 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:	
[72] The apparatus of claim 42, wherein the storage means comprises a flash memory.	[36.a] 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	Appendix C (IPR2013-00004) [52.d] 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,	Non-volatile memory (including flash memory) stores browser instructions for rendering and re-rendering web pages.

U.S. Patent No. 7,844,889	U.S. Patent No. 7,461,353	U.S. Patent No. 7,831,926	Comparison
	processor enable the mobile hand-held device to perform operations including,	Appendix D (90/009,995) [1.e] 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, Appendix E (95/000,635) [79.e] storage means, in which a plurality of instructions are stored,	
[85] The machine-readable medium of claim 77, wherein execution of the instructions performs further operations comprising scaling the scalable page layout information and/or data derived therefrom to regenerate views of the Web content in response to	[252.e] 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	Appendix C (IPR2013-00004) [52.i] 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,	Scaling page layout information (or data derived from it) and regenerating views in response to user inputs to enable iterative zooming.

U.S. Patent No. 7,844,889	U.S. Patent No. 7,461,353	U.S. Patent No. 7,831,926	Comparison
associated user inputs to enable the user to iteratively zoom in and out a display of the Web page.	panned view of the Web page,	and design defined by the HTML-based Web content as interpreted by a rendering engine, Appendix D (90/009,995) [1.i.2.a] 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, Appendix E (95/000,635) [79.i.2] re-render the Web page in response to associated user inputs made via the touch-sensitive display means to enable the	

U.S. Patent No. 7,844,889	U.S. Patent No. 7,461,353	U.S. Patent No. 7,831,926	Comparison
		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,	
[103] The machine-readable medium of claim 77, wherein at least a portion of the instructions comprise Java-based instructions configured to be executed on a Java virtual machine.	[252.pre] 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:	Appendix C (IPR2013-00004) [52.d] 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, Appendix D (90/009,995) [20] The mobile device of claim 1, wherein at least a portion of the instructions comprise Java-based	Implementing at least a portion of the browser instructions as Java-based instructions executable on a Java virtual machine is a logical implementation choice for the plurality of instructions limitations.

U.S. Patent No. 7,844,889	U.S. Patent No. 7,461,353	U.S. Patent No. 7,831,926	Comparison
		instructions configured to be executed on a Java virtual machine. Appendix E (95/000,635) [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.	
[110.a] The method of claim 108, further comprising: [110.b] processing the HTML-based Web content with the rendering engine to parse the text content into a plurality of text objects; [110.c] generating page layout information including a bounding box	[58.c] logically grouping selected content into objects; [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	Appendix C (IPR2013-00004) [72.c] logically grouping selected content into objects; [52.i] 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	Parsing text into text objects, generating a bounding box for each text object, and using bounding box layout information to preserve the original page layout of the text content.

U.S. Patent No. 7,844,889	U.S. Patent No. 7,461,353	U.S. Patent No. 7,831,926	Comparison
for each of the plurality of text objects; and [110.d] employing page layout information associated with the bounding boxes to effect preservation of the original page layout of the text content.	occupied by the object's associated group of content. [36.e] 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.	page layout, functionality, and design defined by the HTML-based Web content as interpreted by a rendering engine, Appendix D (90/009,995) [16.b] parsing HTML-based code corresponding to the received Web content to logically group content into objects, the objects including a plurality of display objects; [18.c] generating page layout information including a bounding box for each object, the bounding box defining width and height dimensions for the object; and [1.i.2.a] re-render the Web page in response to	

U.S. Patent No. 7,844,889	U.S. Patent No. 7,461,353	U.S. Patent No. 7,831,926	Comparison
		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, Appendix E (95/000,635) [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.	

U.S. Patent No. 7,844,889	U.S. Patent No. 7,461,353	U.S. Patent No. 7,831,926	Comparison
		[79.i.2] 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,	
[111.a] The method of claim 110, further comprising: [111.b] scaling bounding boxes for text objects corresponding to text content in a view; and [111.c] employing the scaled bounding boxes to lay out corresponding text	[1.d] 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. [59] The mobile hand-held device of claim 58, wherein execution of the instructions performs further operations comprising: generating a	Appendix C (IPR2013-00004) [43.c] employing the display list to re-render the display at different scale factors to enable rapid zooming of the Web page. [52.i] re-render the Web page in response to associated user inputs to enable the user to	Scaling bounding boxes for text objects and laying out corresponding text content within the scaled bounding boxes.

U.S. Patent No. 7,844,889	U.S. Patent No. 7,461,353	U.S. Patent No. 7,831,926	Comparison
content within the scaled bounding boxes.	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. [36.e] 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.	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, Appendix D (90/009,995) [18.c] generating page layout information including a bounding box for each object, the bounding box defining width and height dimensions for the object; and [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	

U.S. Patent No. 7,844,889	U.S. Patent No. 7,461,353	U.S. Patent No. 7,831,926	Comparison
		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. [24.a] The mobile device of claim 23, wherein rendering scaled content associated with a text object comprises [24.b] retrieving presentation attributes for the text object, the presentation attributes including a font typeface, size and color; [24.c] 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	

U.S. Patent No. 7,844,889	U.S. Patent No. 7,461,353	U.S. Patent No. 7,831,926	Comparison
		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. Appendix E (95/000,635) [70.a] 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; [70.b] determining a virtual display limit bounding box	

U.S. Patent No. 7,844,889	U.S. Patent No. 7,461,353	U.S. Patent No. 7,831,926	Comparison
		for the virtual display associated with the first scale factor and offset; [70.c] identifying object bounding boxes having at least a portion falling within the virtual display limit bounding box; and, [70.d] for each of such object bounding boxes, retrieving content associated with that object bounding box; and [70.e] applying an appropriate scale factor to the content to render the display.	
[112] The method of claim 111, further comprising: employing a scalable font to scale text content for a text object to fit within a scaled bounding box.	[36.e] 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	Appendix C (IPR2013-00004) [52.i] re-render the Web page in response to associated user inputs to enable the user to	Employing a scalable font to scale text content for a text object so the text fits within a scaled bounding box.

U.S. Patent No. 7,844,889	U.S. Patent No. 7,461,353	U.S. Patent No. 7,831,926	Comparison
	original page layout, functionality, and design of the Web page content at each zoom level and panned view. [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.	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, Appendix D (90/009,995) [24.a] The mobile device of claim 23, wherein rendering scaled content associated with a text object comprises [24.b] retrieving presentation attributes for the text object, the presentation attributes including a font typeface, size and color; [24.c] employing a scalable font associated with the font typeface to render text associated with the text	

U.S. Patent No. 7,844,889	U.S. Patent No. 7,461,353	U.S. Patent No. 7,831,926	Comparison
		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. Appendix E (95/000,635) [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.	
[113.a] The method of claim 111, further comprising:	[36.d] employing at least one of the scalable content or data derived therefrom	Appendix C (IPR2013-00004)	In conjunction with rendering a view, using bounding boxes (or data

U.S. Patent No. 7,844,889	U.S. Patent No. 7,461,353	U.S. Patent No. 7,831,926	Comparison
[113.b] in conjunction with rendering a view of the Web page content, [113.c] employing the bounding boxes and/or data derived therefrom to determine image and text objects to be rendered.	to, render the Web page on the display; and [58.c] logically grouping selected content into objects; [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.	[52.h] render a view of the Web page on the touch-sensitive display; and [72.c] logically grouping selected content into objects; Appendix D (90/009,995) [18.c] generating page layout information including a bounding box for each object, the bounding box defining width and height dimensions for the object; and [1.i.1] render a view of at least a portion of the Web page on the touch-sensitive display using a first scale factor; and Appendix E (95/000,635)	derived from them) to determine which image and text objects to render.

U.S. Patent No. 7,844,889	U.S. Patent No. 7,461,353	U.S. Patent No. 7,831,926	Comparison
		[70.a] 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; [70.b] determining a virtual display limit bounding box for the virtual display associated with the first scale factor and offset; [70.c] identifying object bounding boxes having at least a portion falling within the virtual display limit bounding box; and,	

U.S. Patent No. 7,844,889	U.S. Patent No. 7,461,353	U.S. Patent No. 7,831,926	Comparison
		[70.d] for each of such object bounding boxes, retrieving content associated with that object bounding box; and [70.e] applying an appropriate scale factor to the content to render the display.	

Appendix B - Unpatentable '353 Patent Claims

Unpatentable claims 1, 33, 36, 43, 48, 51, 52, 58, 59, 66, 118, 138, 139, 149, 183, 252, 283, and 317 of U.S. Patent No. 7,461,353	
[1.pre]	A wireless device, comprising:
[1.a]	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,
[1.b]	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;
[1.c]	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
[1.d]	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.
[33]	The wireless device of claim 1, wherein a portion of the scalable content comprises vector-based content.
[36.pre]	A mobile hand-held device, comprising:
[36.a]	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

Unpatentable claims 1, 33, 36, 43, 48, 51, 52, 58, 59, 66, 118, 138, 139, 149, 183, 252, 283, and 317 of U.S. Patent No. 7,461,353

plurality of instructions are stored that when executed by the processor enable the mobile hand-held device to perform operations including,

[36.b] 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;

[36.c] retrieving the Web page via the wireless communications device, and processing HTML-based Web content to produce scalable content; and

[36.d] employing at least one of the scalable content or data derived therefrom to, render the Web page on the display; and

[36.e] 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.

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

[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.

[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

Unpatentable claims 1, 33, 36, 43, 48, 51, 52, 58, 59, 66, 118, 138, 139, 149, 183, 252, 283, and 317 of U.S. Patent No. 7,461,353

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.

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

[58.b] parsing HTML-based code corresponding to the retrieved Web page to determine the original page layout of the content on the Web page;

[58.c] logically grouping selected content into objects;

[58.d] defining a primary datum corresponding to the original page layout; and,

[58.e] for each object,

[58.e.1] defining an object datum corresponding to a layout location datum for the object's associated display content;

[58.e.2] generating a vector from the primary datum to the object datum for the object; and

[58.e.3] 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.

Unpatentable claims 1, 33, 36, 43, 48, 51, 52, 58, 59, 66, 118, 138, 139, 149, 183, 252, 283, and 317 of U.S. Patent No. 7,461,353

[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.

[118.pre] A method, comprising:

[118.a] 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;

[118.b] 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

[118.c] 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.

[138.a] The method of claim 118, further comprising:

[138.b] parsing markup language code corresponding to the retrieved Web page to determine the original page layout of the content on the Web page;

[138.c] logically grouping selected content into objects;

[138.d] defining a primary datum corresponding to the original page layout; and,

Unpatentable claims 1, 33, 36, 43, 48, 51, 52, 58, 59, 66, 118, 138, 139, 149, 183, 252, 283, and 317 of U.S. Patent No. 7,461,353

[138.e] for each object,

[138.e.1] defining an object datum corresponding to a layout location datum for the object's associated display content;

[138.e.2] generating a vector from the primary datum to the object datum for the object; and

[138.e.3] 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.

[149.pre] A method, comprising:

[149.a] 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;

[149.b] retrieving the Web page via the hand-held device, and processing HTML-based Web content to produce scalable content; and

[149.c] employing at least one of the scalable content or data derived therefrom to,

[149.d] render the Web page on a display of the hand-held device; and

Unpatentable claims 1, 33, 36, 43, 48, 51, 52, 58, 59, 66, 118, 138, 139, 149, 183, 252, 283, and 317 of U.S. Patent No. 7,461,353

[149.e] 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.

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

[252.pre] 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:

[252.a] 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;

[252.b] retrieving the Web page and processing HTML-based Web content to produce scalable content; and

[252.c] employing at least one of the scalable content or data derived therefrom to, render the Web page on the display; and

[252.d] render the Web page on the display; and

[252.e] 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,

[252.f] 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.

Unpatentable claims 1, 33, 36, 43, 48, 51, 52, 58, 59, 66, 118, 138, 139, 149, 183, 252, 283, and 317 of U.S. Patent No. 7,461,353

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

[317.pre] A hand-held wireless device, comprising:

[317.a] a processor,

[317.b] a wireless communications interface, to facilitate wireless communication with a network that supports access to the Internet;

[317.c] a display; and

[317.d] 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,

[317.e] rendering a browser interface on the display via which a user is enabled to request access to a Web page including at least one image, at least one column, at least one hyperlink to an external reference and having a width and height;

[317.f] retrieving the Web page via the wireless communications interface;

[317.g] 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

[317.h] enabling the user to, zoom and pan a view of the Web page;

Unpatentable claims 1, 33, 36, 43, 48, 51, 52, 58, 59, 66, 118, 138, 139, 149, 183, 252, 283, and 317 of U.S. Patent No. 7,461,353

[317.i] 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;

[317.j] zoom in on an image of the Web page by tapping on the image via the display;

[317.k] zoom in on a column of the Web page by tapping on the column via the display; and

[317.l] zoom out to a previous view of the Web page.

Appendix C - Unpatentable '926 Patent Claims in IPR2013-00004

Unpatentable Claims 30, 31, 40, 41, 43, 52, 55, 59, 72, and 75 of U.S. Patent No. 7,831,926
[30.pre] A mobile phone, comprising:
[30.a] a processor,
[30.b] wireless communications means operatively coupled to the processor, to facilitate communication with a mobile service provider network via which Web content may be accessed;
[30.c] a touch-sensitive display;
[30.d] a memory, operatively coupled to the processor; and
[30.e] 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,
[30.f] 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;
[30.g] retrieving HTML-based content associated with the Web page;
[30.h] 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

Unpatentable Claims 30, 31, 40, 41, 43, 52, 55, 59, 72, and 75 of U.S. Patent No. 7,831,926

[30.i] 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,

[30.j] 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.

[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.

[43.a] The mobile phone of claim 30, wherein execution of the instructions performs further operations comprising:

[43.b] generating a display list associated with the scalable vector-based content; and

[43.c] employing the display list to re-render the display at different scale factors to enable rapid zooming of the Web page.

Unpatentable Claims 30, 31, 40, 41, 43, 52, 55, 59, 72, and 75 of U.S. Patent No. 7,831,926

[52.pre] A mobile device, comprising:

[52.a] a processor,

[52.b] wireless communications means, to facilitate wireless communication with a network via which Web content may be accessed;

[52.c] a touch-sensitive display;

[52.d] 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,

[52.e] 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;

[52.f] retrieving and processing the HTML-based Web content to produce scalable content; and

[52.g] employing the scalable content and/or data derived therefrom to,

[52.h] render a view of the Web page on the touch-sensitive display; and

[52.i] 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,

[52.j] wherein preservation of the functionality defined by the HTML-based Web content includes preservation of hyperlink functionality.

Unpatentable Claims 30, 31, 40, 41, 43, 52, 55, 59, 72, and 75 of U.S. Patent No. 7,831,926

[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.

[59] The mobile device of claim 52, wherein at least a portion of the scalable content comprises scalable vector-based content.

[72.a] The mobile device of claim 52, wherein execution of the instructions performs further operations comprising:

[72.b] parsing HTML-based code corresponding to the received Web content to identify content on the Web page;

[72.c] logically grouping selected content into objects;

[72.d] defining a primary datum corresponding to the original page layout; and,

[72.e] for each object,

[72.e.1] defining an object datum corresponding to a layout location datum for the object's associated display content;

[72.e.2] generating a vector from the primary datum to the object datum for the object; and

[72.e.3] creating a reference that links the object to its corresponding vector.

[75.a] 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:

Unpatentable Claims 30, 31, 40, 41, 43, 52, 55, 59, 72, and 75 of U.S. Patent No. 7,831,926
[75.b] determining an applicable scale factor to fit the width of the Web page across a display area of the touch-sensitive display; and
[75.c] employing the scale factor to render the display area.

Appendix D - Unpatentable '926 Patent Claims in 90,009,995

Unpatentable Claims 1-28, 32-39, 42, 44-51, and 69 of U.S. Patent No. 7,831,926	
[1.pre]	A mobile device, comprising:
[1.a]	a processor,
[1.b]	a wireless communications device operatively coupled to the processor, to facilitate communication with a network via which Web content may be accessed;
[1.c]	a touch-sensitive display;
[1.d]	a memory, operatively coupled to the processor; and
[1.e]	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,
[1.f]	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;
[1.g]	retrieving HTML-based Web content associated with the Web page;
[1.h]	translating the HTML-based Web content to produce scalable vector-based page layout information;
[1.i]	employing the scalable vector-based page layout information and/or data derived therefrom to,
[1.i.1]	render a view of at least a portion of the Web page on the touch-sensitive display using a first scale factor; and

Unpatentable Claims 1-28, 32-39, 42, 44-51, and 69 of U.S. Patent No. 7,831,926

[1.i.2.a] 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,

[1.i.2.b] 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.a] 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:

[6.b] enabling the user to select the hyperlink via the touch-sensitive display; and, in response thereto,

[6.c] retrieving and translating HTML-based Web content associated with the hyperlink to produce additional scalable vector-based page layout information; and

[6.d] 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.

Unpatentable Claims 1-28, 32-39, 42, 44-51, and 69 of U.S. Patent No. 7,831,926

[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

Unpatentable Claims 1-28, 32-39, 42, 44-51, and 69 of U.S. Patent No. 7,831,926

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.a] The mobile device of claim 1, wherein execution of the instructions performs further operations comprising:

[15.b] generating a display list derived, at least in part, via use of the vector-based page layout information; and

[15.c] employing the display list to re-render the display of the Web page.

[16.a] The mobile device of claim 1, wherein execution of the instructions performs further operations comprising:

[16.b] parsing HTML-based code corresponding to the received Web content to logically group content into objects, the objects including a plurality of display objects;

[16.c] defining a primary datum corresponding to a page layout; and,

[16.d] 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.

Unpatentable Claims 1-28, 32-39, 42, 44-51, and 69 of U.S. Patent No. 7,831,926

[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.a] The mobile device of claim 1, wherein execution of the instructions performs further operations comprising:

[18.b] parsing the HTML-based content to logically group content into objects;

[18.c] generating page layout information including a bounding box for each object, the bounding box defining width and height dimensions for the object; and

[18.d] storing information that links each object with its corresponding page layout information;

[18.e] 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.a] The mobile device of claim 1, wherein translating the HTML-based Web content to produce scalable vector-based page layout information comprises:

[21.b] 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

Unpatentable Claims 1-28, 32-39, 42, 44-51, and 69 of U.S. Patent No. 7,831,926

[21.c] employing the page layout information to produce scalable vector-based page layout information.

[22.a] 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:

[22.b] defining a primary datum corresponding to a page layout; and

[22.c] for each object,

[22.c.1] defining an object datum corresponding to the layout location for the object on the page layout;

[22.c.2] generating a vector from the primary datum to the object datum for the object; and

[22.c.3] creating a reference that links the object to its corresponding vector.

[23.a] 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,

[23.b] 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,

[23.b.1] 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;

[23.b.2] 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;

Unpatentable Claims 1-28, 32-39, 42, 44-51, and 69 of U.S. Patent No. 7,831,926

[23.b.3] scaling content associated with the display object using the scale factor; and

[23.b.4] 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.a] The mobile device of claim 23, wherein rendering scaled content associated with a text object comprises

[24.b] retrieving presentation attributes for the text object, the presentation attributes including a font typeface, size and color;

[24.c] 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.a] 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

[25.b] determining an applicable scale factor to fit the width of the Web page across a display area of the touch-sensitive display; and

[25.c] employing the scale factor that is determined as the first scale factor.

[26.a] 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

Unpatentable Claims 1-28, 32-39, 42, 44-51, and 69 of U.S. Patent No. 7,831,926

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

[26.b] $X' = XSF$;

[26.c] $Y' = YSF$;

[26.d] 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.

[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.a] 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:

[33.b] enabling the user to select the hyperlink via the touch-sensitive display; and, in response thereto,

Unpatentable Claims 1-28, 32-39, 42, 44-51, and 69 of U.S. Patent No. 7,831,926

[33.c] retrieving and translating the Web content associated with the hyperlink to produce additional scalable vector-based content; and

[33.d] employing the additional scalable vector-based content to render the Web content associated with the hyperlink on the touch-sensitive display.

[34.a] The mobile phone of claim 30, wherein execution of the instructions performs further operations comprising:

[34.b] 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;

[34.c] defining a primary datum corresponding to the original page layout; and,

[34.d] for each object,

[34.d.1] defining an object datum corresponding to the layout location for the object;

[34.d.2] generating a vector from the primary datum to the object datum for the object; and

[34.d.3] 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.

Unpatentable Claims 1-28, 32-39, 42, 44-51, and 69 of U.S. Patent No. 7,831,926

[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.

[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.

[44.a] The mobile phone of claim 30, wherein execution of the instructions performs further operations comprising:

[44.b] 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;

[44.c] logically grouping selected content into objects;

Unpatentable Claims 1-28, 32-39, 42, 44-51, and 69 of U.S. Patent No. 7,831,926	
[44.d]	defining a primary datum corresponding to the page layout; and,
[44.e]	for each object,
[44.e.1]	defining an object datum corresponding to a layout location datum for the object's associated display content;
[44.e.2]	generating a vector from the primary datum to the object datum for the object; and
[44.e.3]	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.a]	The mobile phone of claim 45, wherein execution of the instructions performs further operations comprising:
[46.b]	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;
[46.c]	determining a virtual display limit bounding box for the virtual display area associated with the first scale factor and offset;
[46.d]	identifying objects having at least a portion of their content falling within the virtual display limit bounding box; and,
[46.e]	for each of such objects,

Unpatentable Claims 1-28, 32-39, 42, 44-51, and 69 of U.S. Patent No. 7,831,926

[46.e.1] retrieving content associated with that object; and

[46.e.2] applying an appropriate scale factor and offset to the content to render the view of the Web content.

[47.a] The mobile phone of claim 30, wherein execution of the instructions performs further operations comprising:

[47.b] parsing markup language code corresponding to the received Web content to logically group selected content into objects;

[47.c] generating page layout information including a bounding box for each object, the bounding box defining width and height dimensions for the object; and

[47.d] storing information that links each object with its corresponding page layout information;

[47.e] wherein the page layout information further includes information from which a page layout location of each of the bounding boxes can be determined.

[48.a] The mobile phone of claim 30, wherein the scalable vector-based content includes scalable text content, and

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

[49.a] 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:

Unpatentable Claims 1-28, 32-39, 42, 44-51, and 69 of U.S. Patent No. 7,831,926

[49.b] determining an applicable scale factor to fit the width of the Web page across a display area of the touch-sensitive display; and

[49.c] 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.a] 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:

[51.b] 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;

[51.c] employing the page layout information to generate the scalable vector-based page layout information.

[69.a] 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:

[69.b] mapping the object vectors and associated bounding boxes to a virtual display area in the dynamic memory.

Appendix E - Unpatentable '926 Patent Claims in 95,000,635

Unpatentable Claims 29, 53, 54, 56-58, 60-68, 70, 71, 73, 74, and 76-88 of U.S. Patent No. 7,831,926
[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.
[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.
[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.a] 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:
[58.b] enabling the user to select the hyperlink via the touch-sensitive display; and, in response thereto,
[58.c] retrieving and processing HTML-based Web content associated with the hyperlink to produce additional scalable content; and
[58.d] employing the additional scalable content and/or data derived therefrom to render the Web content associated with the hyperlink on the touch-sensitive display.

Unpatentable Claims 29, 53, 54, 56-58, 60-68, 70, 71, 73, 74, and 76-88 of U.S. Patent No. 7,831,926

[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

Unpatentable Claims 29, 53, 54, 56-58, 60-68, 70, 71, 73, 74, and 76-88 of U.S. Patent No. 7,831,926

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.a] 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:

[68.b] receiving content corresponding to the text and layout attributes via a first connection; and

[68.c] receiving content corresponding to at least one image via a second connection.

[70.a] 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;

[70.b] determining a virtual display limit bounding box for the virtual display associated with the first scale factor and offset;

[70.c] identifying object bounding boxes having at least a portion falling within the virtual display limit bounding box; and,

[70.d] for each of such object bounding boxes, retrieving content associated with that object bounding box; and

[70.e] applying an appropriate scale factor to the content to render the display.

Unpatentable Claims 29, 53, 54, 56-58, 60-68, 70, 71, 73, 74, and 76-88 of U.S. Patent No. 7,831,926

[71.a] The mobile device of claim 52, further comprising dynamic memory having at least a portion employed for rendering purposes,

[71.b] 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

[71.c] scaling the display list to re-render the display of the Web page.

[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.

[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.pre] A mobile device, comprising:

[79.a] processing means;

Unpatentable Claims 29, 53, 54, 56-58, 60-68, 70, 71, 73, 74, and 76-88 of U.S. Patent No. 7,831,926

[79.b] wireless communications means, to facilitate wireless communication with a network via which Web content may be accessed;

[79.c] touch-sensitive display means, to facilitate user input and display rendered content;

[79.d] programmed circuit means; and

[79.e] storage means, in which a plurality of instructions are stored,

[79.f] 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,

[79.g] 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;

[79.h] retrieving and processing the HTML-based Web content to produce scalable content; and

[79.i] employing the scalable content and/or data derived therefrom to,

[79.i.1] render a view of the Web page on the touch-sensitive display; and

[79.i.2] 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,

[79.i.2.a] wherein preservation of the functionality defined by the HTML-based Web content includes preservation of hyperlink functionality.

Unpatentable Claims 29, 53, 54, 56-58, 60-68, 70, 71, 73, 74, and 76-88 of U.S. Patent No. 7,831,926

[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.

Unpatentable Claims 29, 53, 54, 56-58, 60-68, 70, 71, 73, 74, and 76-88 of U.S. Patent No. 7,831,926

[88.a] 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

[88.b] 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

[88.c] scaling the scalable content in the display list to re-render the display of the Web page.