

UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE PATENT TRIAL AND APPEAL BOARD

GOOGLE LLC,
Petitioner

v.

SOFTVIEW LLC,
Patent Owner

Patent No. 10,083,154

**DECLARATION OF DR. BENJAMIN GOLDBERG IN SUPPORT OF
PETITION FOR *INTER PARTES* REVIEW
OF U.S. PATENT NO. 10,083,154**

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 '154 PATENT	12
VI.	UNPATENTABLE CLAIMS IN RELATED PATENTS	14
VII.	OVERVIEW OF THE PRIOR ART	14
	A. Zaurus (EX1004)	14
	B. Pad (EX1005 and EX1007)	15
	C. SVG (EX1008)	16
	D. SVF (EX1044)	16
VIII.	CLAIM CONSTRUCTION	17
IX.	DETAILED EXPLANATION OF GROUNDS	19
	A. Ground 1: Zaurus, Pad, and SVG Disclose and/or Suggest Claims 1-35	20
	1. Motivation to Combine	20
	2. Reasonable Expectation of Success	22
	3. Claim 1	24
	4. Claim 2: "The mobile hand-held device of claim 1, wherein the mobile hand-held device is a mobile phone."	77
	5. Claim 3: "The mobile hand-held device of claim 1, wherein the mobile hand-held device is a hand-held computer."	77
	6. Claim 4: "The mobile hand-held device of claim 1, wherein execution of the instructions further enables the user to zoom in on a user-selectable portion of a view of the HTML document in response to a user interface input made via the touch-sensitive display."	77

7.	Claim 5: “The mobile hand-held device of claim 1, wherein the views of the HTML document are rendered in real-time to effect zooming operations.”	78
8.	Claim 6: “The mobile hand-held device of claim 5, wherein the zooming operations are effected in a fraction of a second.”	79
9.	Claim 7	79
10.	Claim 8: “The mobile hand-held device of claim 1, wherein the HTML document is an HTML parent document for a Web page.”	80
11.	Claim 9: “The mobile hand-held device of claim 1, wherein the HTML elements includes an HTML table having a plurality of cells, each cell having at least one of associated text content and associated image content, and the CSS code includes CSS code defining how the table and its associated content is to be rendered.”	81
12.	Claim 10	82
13.	Claim 11	84
14.	Claim 12: “The mobile hand-held device of claim 1, wherein the primary datum is an origin in a two-dimensional XY coordinate system and the vectors from the primary datum to the object datums for the HTML object bounding boxes comprise XY coordinates in the coordinate system.”	86
15.	Claim 13	87
16.	Claim 14: “The method of claim 13, wherein the mobile hand-held device is a mobile phone.”	92
17.	Claim 15: “The method of claim 13, wherein the mobile hand-held device is a hand-held computer.”	92
18.	Claim 16: “The method of claim 13, further comprising enabling the user to zoom in on a user-selectable portion of a view of the HTML document in response to a user interface input made via the touch-sensitive display.”	93

19.	Claim 17: “The method of claim 13, wherein the views of the HTML document are rendered in real-time to effect zooming operations.”	93
20.	Claim 18: “The method of claim 17, wherein the zooming operations are effected in a fraction of a second.”	93
21.	Claim 19	93
22.	Claim 20: “The method of claim 13, wherein the HTML document is an HTML parent document for a Web page.”	94
23.	Claim 21	94
24.	Claim 22: “The method of claim 13, wherein the HTML elements includes an HTML table having a plurality of cells, each cell having at least one of associated text content and associated image content, and the CSS code includes CSS code defining how the table and its associated content is to be rendered.”	95
25.	Claim 23	96
26.	Claim 24	97
27.	Claim 25: “The method of claim 13, wherein the primary datum is an origin in a two-dimensional XY coordinate system and the vectors from the primary datum to the object datums for the HTML object bounding boxes comprise XY coordinates in the coordinate system.”	97
28.	Claim 26	98
29.	Claim 27: “The mobile hand-held device of claim 26, wherein the mobile hand-held device is a mobile phone.”	101
30.	Claim 28: “The mobile hand-held device of claim 26, wherein the mobile hand-held device is a hand-held computer.”	101
31.	Claim 29: “The mobile hand-held device of claim 26, wherein execution of the instructions further enables the user to zoom in on a user-selectable portion of a view of the HTML document in response to a user interface input made via the touch-sensitive display.”	101

32.	Claim 30: “The mobile hand-held device of claim 26, wherein the views of the HTML document are rendered in real-time to effect zooming operations.”	101
33.	Claim 31: “The mobile hand-held device of claim 30, wherein the zooming operations are effected in a fraction of a second.”	102
34.	Claim 32	102
35.	Claim 33: “The mobile hand-held device of claim 26, wherein the HTML document is an HTML parent document for a Web page.”	103
36.	Claim 34: “The mobile hand-held device of claim 26, wherein the HTML-based content includes an HTML table, and the CSS code includes CSS code defining how the table is to be formatted in the interpreted page layout page of the HTML document.”	103
37.	Claim 35	103
B.	Ground 2: Zaurus, Pad, SVG, and SVF Disclose and/or Suggest Claims 1-35	104
1.	Motivation to Combine	104
2.	Reasonable Expectation of Success	105
3.	Claims 1-35	106
X.	CONCLUSION.....	109
	Appendix A - Comparison to Unpatentable Claims	110
	Appendix B - Unpatentable ’353 Patent Claims.....	174
	Appendix C - Unpatentable ’926 Patent Claims in IPR2013-00004.....	182
	Appendix D - Unpatentable ’926 Patent Claims in 90,009,995	187
	Appendix E - Unpatentable ’926 Patent Claims in 95,000,635.....	200

Declaration of Dr. Benjamin Goldberg

U.S. Patent No. 10,083,154

I, Benjamin Goldberg, declare as follows:

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. 10,083,154 (“the ’154 patent”) (EX1001).¹ I have been asked to consider whether prior art references disclose or suggest the features recited in claims 1-35 (“the challenged claims”) of the ’154 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

¹ In this Declaration, I refer to exhibits that I understand are to be attached to the petition for *Inter Partes* Review of the ’154 patent.

Declaration of Dr. Benjamin Goldberg

U.S. Patent No. 10,083,154

Science from September 1995 through August 1998 and from September 2003 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. 10,083,154

EX1001	U.S. Patent No. 10,083,154 to Rohrabau gh et al. (“the ’154 patent”)
EX1003	Prosecution History of U.S. Patent App. No. 15/356,456
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 Rohrabau gh et al. (“the ’926 patent”)
EX1015	U.S. Patent No. 7,461,353 to Rohrabau gh 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)
EX1044	Specification for the Simple Vector Format v.1.1 (Wayback Machine Capture on Oct. 19, 1996) (“SVF”)

EX1045	SoftSource, <i>Changes to SVF</i> , SoftSource (Wayback Machine Capture on July 13, 2002)
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	OpenGL Programming Guide, Second Edition, Addison-Wesley, 1997 (available at https://laplace.physics.ubc.ca/People/ajpenner/GPGPU/redbook.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 '154 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 '154 patent was around 2000 (including and up to June 12, 2000), which

Declaration of Dr. Benjamin Goldberg

U.S. Patent No. 10,083,154

corresponds to the filing date of the earliest application associated with the '154 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 '154 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 '154 patent as of around June 12, 2000. Considering the '154 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 '154 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 '154 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 '154 PATENT

15. The '154 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., 1:65-2:62.) The '154 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), 17:50-61; *see also id.* at 15:50-17:50.)

16. The '154 patent utilizes the SVF (Simple Vector Format) considered by the W3C (World Wide Web Consortium) to translate content into a scalable vector representation. (EX1001, 4:56-5:3, 6:13-28, 6:61-7:5, 9:1-6, 12:23-31.) Relative to the prior art, the '154 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, 17:62-65.) 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, 17:61-19:35, 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, 19:35-20:67, FIG. 4E-4G, 5, 6.)

17. The '154 patent acknowledges that building the display list of vectors and rapid zooming of vector-based objects was well known in the art. (EX1001, 19:42-44.) The '154 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., 21:1-36.) The '154 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 '154 patent is directed to a mobile handheld device with a processor, wireless interface, touch-sensitive display, and memory storing instructions to (i) receive an HTML document (with CSS and image URLs), (ii) render a first, standards-compliant layout, (iii) translate that layout into a scalable vector representation using the datum-and-vector approach, and (iv) render at multiple user-defined zoom levels while preserving the original layout and interactive design. (EX1001, 22:39-23:42.)

19. Prosecution was brief. SoftView filed the patent application with one independent claim and eleven dependent claims but then canceled all the dependent claims in a preliminary amendment filed the same day. (EX1003, 49-51, 75-78.) The first and only Office action was a rejection of the remaining independent claim

under (i) obviousness in view of the prior art and (ii) non-statutory double patenting in view of the '729 patent. (EX1003, 91-104.) In response, SoftView canceled the pending independent claim, presented 35 new claims, and filed a terminal disclaimer over the '729 patent. (EX1003, 161-173, 224.) The Examiner did not raise concerns over the extent of the amendments and instead allowed the claims after an Examiner interview and Examiner's amendments that incorporated subject matter from a dependent claim. (EX1003, 231-258.)

VI. UNPATENTABLE CLAIMS IN RELATED PATENTS

20. I have been informed 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, 6, 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.)

D. SVF (EX1044)

24. The Simple Vector Format (SVF) is designed for describing vector images, structured into three distinct parts: the file intro, the header, and the main body. (EX1044, 1.) The format relies on a coordinate system where the origin (0,0) starts in the lower-left corner, with coordinates specified as positive integers. (EX1044, 1-2.) SVF supports a standard set of drawing objects including points,

lines, circles, arcs, text, and cubic Bezier curves, the latter of which are defined by two control points and a final endpoint. (EX1044, 1, 4.) SVF includes interactive features such as layers to manage object visibility, hyperlinks that allow users to click regions to perform actions, and notifications that trigger specific behaviors when the viewer reaches a certain zoom level. (EX1044, 5-7.) To maintain efficient files, command parameters are stored as variable-length, big-endian integers. (EX1044, 1.)

VIII. CLAIM CONSTRUCTION

25. 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 '154 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 '154 patent and the prior art.

26. I understand that the following terms appearing in the '154 patent were previously construed by a district court and/or submitted as proposed constructions in district court.

Term	Construction
“A mobile hand-held device, comprising”	The preamble is limiting.
“A method performed by a mobile hand-held device including a touch-sensitive display and an HTML rendering engine”	The preamble is limiting.
“format”	“a particular way that information is encoded”
“translating” / “translate”	“converting” / “convert”
“scalable content”	“content capable of being rendered at multiple zoom levels”
“scalable” / “scale” ²	“capable of being rendered at multiple zoom levels” / “render at multiple zoom levels”
“preserved” / “preserving”	The term “preserve” and its variants have their plain and ordinary meaning.
“fit across”	Plain and ordinary meaning

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

“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 representation”	“representation of the [Web page / HTML document] 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”
“primary datum”	“a fixed origin point defined at an X, Y coordinate”
“layout location datum”	“point defined at an X, Y coordinate that corresponds to the location of the object”

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

IX. DETAILED EXPLANATION OF GROUNDS

28. In my opinion, the prior art I discuss below discloses and/or suggests the limitations of claims 1-35.

A. Ground 1: Zaurus, Pad, and SVG Disclose and/or Suggest Claims 1-35

1. Motivation to Combine

29. 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.)

30. 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.)

31. 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.

32. 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

³ Notably, the ’154 patent explains that the invention uses “core functionality provided by the Mozilla rendering engine source code.” (EX1001, 17:50-61.) 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.”).)

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

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

34. 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.)

35. 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. Claim 1

- i) [1.pre] **“A mobile hand-held device, comprising:”**
- ii) [1.a] **“a processor;”**
- iii) [1.b] **“a wireless communications device, to facilitate wireless communication with a network that supports access to the Internet;”**
- iv) [1.c] **“a touch-sensitive display; and”**
- v) [1.d] **“flash memory, operatively coupled to the processor, in which a plurality of instructions are stored comprising a plurality of software components including an HTML rendering engine, wherein the instructions are configured to be executed by the processor to enable the mobile hand-held device to,”**

36. In the Zaurus-Pad-SVG combination, Zaurus is a “Mobile Business Tool” (“mobile hand-held device”) that is “highly portable with compact size and lightweight, browsing can be done by accessing the Internet.” (EX1004, 9-10; *see also id.*, 11-15.) Zaurus includes a CPU (“processor”). (EX1004, 18.) Zaurus includes adapters that support digital cellular for data communication to access the Internet (“a wireless communications device, to facilitate wireless communication with a network that supports access to the Internet”). (EX1004, 5-7, 11-13.) Zaurus further describes a display that receives touch inputs (“touch-sensitive display”). (EX1004, 15 (“Touch the screen with a pen. Let's try various operations by touching the screen with a pen.”).)

37. Zaurus includes flash memory (“flash memory”). (EX1004, 6, 18, 43.) The flash memory is used in connection with the Internet browser by placing the browser's cache in the “main body flash” or “card flash” (“operatively coupled to”).

the processor”). (EX1004, 43, 48, 52.) Zaurus is pre-installed with an Internet browser (“in which a plurality of instructions are stored comprising a plurality of software components...instructions are configured to be executed by the processor to enable the mobile hand-held device to” [perform web functions]”). (EX1004, 39 (“Connect to the Internet, and open Zaurus Plaza (the Zaurus home page).”).) The Internet browser supports HTML 3.2 to display content such as text and images in GIF and JPEG format (“HTML rendering engine”). (EX1004, 40 (“It can display images not only in GIF format, but also in the commonly found JPEG format on a page in 16-gradation grayscale.”).) The Web pages can be displayed frame by frame, which a person of ordinary skill in the art would have understood are implemented using standard HTML <frame> markup (e.g., <frame>/<frameset>). (EX1004, 40 (“Pages composed of frames can also be viewed by displaying them frame by frame.”); *see also id.*, 38, 48-49; EX1041.)

38. In the combined system, Zaurus’s pre-installed browser is an executable software component stored in flash memory and run by the CPU to provide the disclosed Internet browsing functions, including the rendering of HTML content, images, hyperlinks, and multi-region page structures. (EX1004, 18, 39-41, 43.) Zaurus’s disclosure that it “incorporates HTML 3.2 specifications” and supports “pages composed of frames” would have been understood by a person of ordinary skill in the art to require the browser’s “rendering engine” to

interpret and process the underlying HTML structure and corresponding elements to generate the displayed page regions and frame-by-frame presentation. (EX1004, 40, 48-49; EX1041.)

- vi) **[1.e] “receive an HTML document including HTML code and cascading style sheet (CSS) code and content associated with the HTML document, the HTML code comprising a plurality of HTML elements including at least one HTML paragraph element, at least one HTML image element, and at least one HTML hyperlink element,”**

39. In the Zaurus-Pad-SVG combination, Zaurus discloses downloading HTML web pages that contain multiple HTML elements such as text, images, and links (“receive an HTML document including HTML code ... and content associated with the HTML document”). (EX1004, 10 (“download the information you want, and view it later at your own pace”), 40 (“The Internet browser in the MI-100 series incorporates HTML 3.2 specifications in important places, and therefore has rich expressive capabilities.”).) Zaurus displays blocks of text on Web pages (“at least one HTML paragraph element”). (EX1004, 10-11.) A person of ordinary skill in the art would have understood that text is included in a “paragraph” encoded as HTML code <p>. (EX1042, 1 (“The text up to the next <p> element is treated as being part of the current paragraph.”); EX1043, 2, 10.) Zaurus further describes displaying pages frame by frame and images in GIF and JPEG format (“at least one HTML image element”). (EX1004, 38, 40 (“It can display images not only in GIF

format, but also in the commonly found JPEG format on a page in 16-gradation grayscale...Pages composed of frames can also be viewed by displaying them frame by frame.”), 48-49.) Zaurus further describes hyperlinks for selecting links that cause retrieval of a page (“at least one HTML hyperlink element”). (EX1004, 10 (“A separate page which is linked is displayed by touching the underlined characters.”).) A person of ordinary skill in the art would have recognized that incorporating HTML 3.2 to receive text, image, and hyperlink elements would involve the receipt of HTML code and associated content from a server before rendering the content on the device’s display.

40. Pad discloses fetching HTML content, parsing it, handling images, and creating hyperlinks. (EX1005, 5 (“Retrieve a URL over the internet in the background”), 15 (“HTML items know about the internet and will automatically fetch a file from a URL (Universal Resource Locator) as well as in-line images”), 16-19.) Pad further shows text hyperlinks and the user’s selection of those links. (EX1007, 108 (“in the Pad++ HTML browser, some words are shown in blue. They are ‘hotwords’ - positioning the pointer over the word causes it to change color to red. When you click on the word, the appropriate document is placed on the Pad++ surface”), 111-113 (illustrating text acting as a hyperlink).)

41. SVG confirms that HTML-related rendering environments relied on the cascading style sheet (CSS) code for layout and style (“cascading style sheet (CSS)”).

code”). (EX1008, 4 (“Compatible with and/or leverages other relevant standards efforts, including XML namespaces, XML links (XLink), DOM, CSS, XSL and metadata...attributes which make sense to be part of a style should be expressed in CSS”).) The Document Object Model (DOM) provides a standardized programming interface that defines HTML elements as objects. (EX1035.) The authors of SVG, W3C, described “DOM access...the DOM should expose graphic objects down to the individual point.” (EX1008, 15.) SVG reflects that CSS support (including CSS positioning/styling) was a web-standard mechanism the Web community expected to be leveraged in HTML-based Web content. (EX1008, 5, 6, 7 (“All text/font attributes from CSS should be supported. Complete support for CSS is what the Web community will expect.”), 11 (“SVG should use CSS positioning”).) In the relevant timeframe, it was likewise conventional that web page presentation was governed not only by the HTML element structure, but also by associated styling information (including CSS) that specified layout and presentation attributes for structured documents. (EX1037.)

42. SVG confirms that browser-based rendering environments were expected to be “compatible with and/or leverage” standards including “DOM” and “CSS,” and that attributes “which make sense to be part of a style should be expressed in CSS,” reinforcing that CSS was treated as the standard mechanism for expressing style/layout information for web content. (EX1008, 4-7.) SVG further

reflects that “complete support for CSS is what the Web community will expect” and that “SVG should use CSS positioning,” which a person of ordinary skill in the art would have understood as confirming that CSS code (including positioning/layout properties) would be received and interpreted in connection with rendering HTML-related content. (EX1008, 6-7, 11.) In addition, SVG’s repeated requirement for “DOM access” confirms that HTML elements were treated as addressable objects in a standardized object model, consistent with receiving and identifying a plurality of HTML elements (e.g., paragraphs, images, hyperlinks) from the HTML code before rendering them. (EX1008, 15; EX1035.)

43. A person of ordinary skill in the art designing a mobile handheld web device would have combined Zaurus’s mobile HTML browser (that supports image display and hyperlinks) with Pad’s receipt and parsing of HTML (including links and images) and SVG’s CSS styling for Web content. The combined device implementing the Zaurus HTML browser would receive and interpret CSS based style information along with the HTML elements. The combination of Zaurus and SVG is a device that receives HTML documents containing multiple standard HTML elements (text, images, and hyperlinks) and uses CSS code to govern their appearance. A person of ordinary skill in the art would have been motivated to have the handheld HTML browser receive HTML documents containing text, images, links, and CSS data, because these were features of HTML pages. (EX1004, 40;

EX1008, 4.) In view of SVG describing that CSS integration is a design goal for Web design, a person of ordinary skill in the art would have understood that HTML documents included CSS information. Combining these teachings would be a predictable and routine operation of an HTML browser that receives and renders web pages. SVG's mention of CSS and DOM further confirms that rendering engines were interoperable with HTML and CSS.

- vii) **[1.f] “process the HTML document with the HTML rendering engine to render a first representation of the HTML document having an interpreted page layout, functionality, and design of the content associated with the HTML document that is in accordance with the HTML code and CSS code, wherein rendering the first representation of the HTML document includes,”**

44. In the Zaurus-Pad-SVG combination, Zaurus describes processing HTML content with a browser that processes and visually renders web pages (“process the HTML document with the HTML rendering engine”). (EX1004, 10, 39-41, 43, 46; Sections IX.A.3.i-vi.) Zaurus's browser can “display...on a page” based on the interpreted HTML (“render a first representation”). (EX1004, 40.) A person of ordinary skill in the art would have understood that the Zaurus HTML engine processes the HTML structure (frames, images, and text) and renders a page layout that is consistent with the HTML specification.

45. Zaurus discloses the ability to switch from a reduced view to a magnified view, as well as a left and right scrolling control and a vertical scrolling

bar, to view material not currently on the screen. (EX1004, 43-47.) The selectable hyperlink is a functional element that responds to user input. (EX1004, 4, 10, 39.) Zaurus further discloses compatibility with client side clickable maps, so that by clicking inside a displayed map, one can jump to the page which corresponds to that portion and maintain the primary features of the page's appearance. (EX1004, 40.) Pad discloses an "HTML browser" and then illustrates how links are rendered in blue or red, how following a link loads the target document with a two-column layout, and how users zoom on the content ("the HTML document having an interpreted page layout, functionality, and design of the content associated with the HTML document that is in accordance with the HTML code"). (EX1007, 86-93.)

46. As I explained above with respect to limitation [1.e], the combined Zaurus-Pad-SVG device receives HTML documents with CSS information that includes the positioning and spacing of the objects. (EX1008, 5, 6, 7, 11.) The authors of SVG, W3C, described CSS as a standard. (EX1037.) A person of ordinary skill in the art would have recognized that processing and interpreting HTML and CSS code would enable the rendering of the intended page layout, functionality, and design. Pad further corroborates that conventional engines were capable of processing structured data into rendered graphical representations. (EX1005, 6, 7, 11.) A person of ordinary skill in the art would have been motivated to incorporate CSS support ("CSS code") because its positioning information would

enable more accurate and visually consistent rendering. Processing standard HTML and CSS encoded graphical information to produce an interactive visual representation was well known and would be the same rendering operation performed by an HTML engine that converts HTML and CSS code into a formatted page layout.

47. A person of ordinary skill in the art would have understood that “render[ing] a first representation ... having an interpreted page layout, functionality, and design” is the ordinary operation of an HTML rendering engine: it interprets the document structure (e.g., text blocks/paragraphs, inline images, anchors/hyperlinks, and frames) and produces a formatted presentation in accordance with the document’s markup and associated style information. (EX1004, 39-41, 43, 48-49; EX1041; EX1042; EX1037.) SVG confirms that, in web rendering environments, CSS and DOM were expected to be leveraged together for layout and styling, such that rendering “in accordance with the HTML code and CSS code” was a predictable and routine implementation choice for an HTML engine seeking accurate and visually consistent rendering. (EX1008, 4-7, 11, 15; EX1035.) Pad further confirms that conventional rendering engines were capable of converting structured page content into rendered graphical representations that preserve page semantics (including links) while supporting interactive browsing operations. (EX1007, 86-93; EX1005, 60-63.)

(1) [1.f.1] “parsing the HTML document to identify the plurality of HTML elements;”

48. In the Zaurus-Pad-SVG combination, Zaurus describes handling HTML content with a browser that processes and visually renders web pages. (EX1004, 10, 39-41, 43, 46; Sections IX.A.3.i-vi.) For example, displaying pages “composed of frames...frame by frame” (EX1004, 40; *see also id.*, 38, 48-49) requires parsing the specific HTML element for frames. A person of ordinary skill in the art would have understood that the Zaurus browser parses the HTML document to identify its defined HTML elements (e.g., paragraphs, images, hyperlinks, frames) for rendering (“parsing the HTML document to identify the plurality of HTML elements”).

49. In my opinion, to the extent a person of ordinary skill in the art required additional implementation details, they would have looked to Pad and SVG. Pad discloses that the HTML data is parsed (“parsing the HTML document”). (EX1005, 60 (“When the html data is fetched, it is parsed and the HTML item is created which contains a method for rendering the page.”).) Pad creates and enumerates HTML anchors as separate items, which demonstrates the identification of HTML elements; Zaurus shows those elements (links, frames, maps) in a handheld browser; and SVG’s DOM framework confirms element-level identification (“to identify the plurality of HTML elements”). (EX1004, 38, 40, 48-49; EX1005, 61-65; EX1008, 15.) Pad describes that HTML documents are represented as graphical objects on a

zoomable surface, and identifies the anchors of HTML items. (EX1005, 62, (“Returns all the anchors that are part of this HTML item”), 63 (“If script is specified, it gets evaluated when the html source has loaded”); EX1007, 17, 20, 108, 111-113.) A person of ordinary skill in the art starting with the Zaurus browser would have used Pad’s parsing and anchor element identification techniques to parse the HTML document for rendering.

50. SVG also confirms the standard methodology for identifying elements in a web context: the Document Object Model (DOM). (EX1035.) SVG mandates “DOM access” and compatibility with the DOM to expose graphic objects. (EX1008, 4, 15 (“DOM access... SVG should provide for complete DOM support for all attributes and elements.”).) A person of ordinary skill in the art would have understood that the DOM is the standard interface for parsing both HTML and XML documents into a tree of addressable objects. Thus, SVG’s disclosure of DOM-based element identification reinforces the motivation to use standard parsing techniques to “identify the plurality of HTML elements” (e.g., anchors, images) within the Zaurus browser’s HTML rendering engine. SVG also describes XML, DOM, and CSS, which involves parsing elements and attributes into objects for rendering. (EX1008, 3-4, 6-8, 15.) The authors of SVG, W3C, described that XML was a standard Extensible Markup Language (XML) in 1998. (EX1046.) In the Zaurus-Pad-SVG combination, a person of ordinary skill in the art would have

incorporated the known parsing techniques of Pad and SVG into the Zaurus browser to parse the HTML document.

(2) [1.f.2] “logically grouping content associated with HTML elements into HTML objects”

51. In the Zaurus-Pad-SVG combination, the parsed HTML elements are represented as discrete items and organized into hierarchical groups (“objects”). Pad maps content to an object type that is then grouped under the page object. (EX1005, 60-64 (hyperlinks), 64-65 (images), 78 (text).) The content is represented as objects and groups in an interactive interface, which groups the content into objects and manages those groups hierarchically (“logically grouping selected content into objects”). (EX1005, 40-41, 59 (“Items of type group are special items that group other items. Group items do not have any visual appearance, but rather are used just for creating structure...Groups are implemented very efficiently, and may be hierarchical (i.e., contain other groups)...Groups automatically resize themselves to contain all their members.”), 62 (“Because an HTML item is a group, it may contain other members in addition to its anchors. This allows setting and retrieving of all members that are part of this HTML item.”).) The HTML browser displays anchored “hotwords” distinctly and, when a link is followed, it loads the sub-document and lays it out next to the parent document, maintaining a grouped spatial relationship among documents and their link objects. (EX1007, 89-96.) A person of ordinary skill in the art would have understood that when an HTML page is accessed in Pad,

rendering the page involves logically grouping the objects corresponding to elements, such as text objects for paragraphs, image objects for images, and anchors for hyperlinks (“logically grouping selected content into objects”).

52. SVG describes DOM representation of elements and an object based vector language that includes grouping and layering objects. (EX1008, 4 (“the elements and attributes of an SVG drawing should be scriptable via the DOM”), 5-6, 10-11, 14, 15 (“SVG should provide for complete DOM support for all attributes and elements”).) A person of ordinary skill in the art would recognize that applying DOM standards to HTML allows for each HTML element to correspond to a DOM object (“object”). SVG also requires grouping objects for scripting and layout into groups. (EX1008, 9 (“There needs to be an ability to define groups (i.e., collections) of objects to allow for convenient access for scripting applications...Explicit layering: objects or collections of objects can be assigned an explicit z-index value.”).)

53. SVG describes designing content in an object based and scriptable environment, and Pad demonstrates the feasibility of grouping in the graphics context (“logically grouping selected content into objects”). A person of ordinary skill in the art would have understood that within an HTML engine pipeline, parsing would be followed by the instantiation of DOM objects for the elements that are then grouped into frames or sections for layout or interaction. Zaurus provides the

browser context with HTML text, images, and links on which to apply grouping.

(EX1004, 38-41, 48-49; EX1005, 59-65; EX1008, 9, 14-15.) Starting with the Zaurus HTML browser, parsing HTML elements into HTML DOM objects and grouping them as taught by Pad and SVG would be a predictable and routine step.

(3) [1.f.3] “generating page layout information including a bounding box for each HTML object; and”

54. In the Zaurus-Pad-SVG combination, the geometric layout information such as bounding boxes can be generated for HTML objects. Zaurus provides the browser; Pad (i) computes and returns per-item bounding boxes (general items, text) and (ii) contains HTML items, i.e., HTML-derived objects, which therefore had corresponding bounding boxes available via the same APIs; and SVG discloses querying an object’s bounds for the web graphics (“including a bounding box for each HTML object”). (EX1004, 40, 48; EX1005, 2, 17, 30, 47, 60; EX1007, 15.)

55. Zaurus explains that “a single page is divided into a certain number of boundaries so that the information displayed within the boundaries can be freely switched.” (EX1004, 48.) Pad’s grid arrangement and multi-column HTML browser layout are page-layout computations (“generating page layout information”). (EX1005, 54-55, 57; EX1007, 91 (“the HTML browser will lay out sub-documents in two columns next to the parent document”).) Pad describes generating bounding boxes in the context of object modeling, layout, navigation,

culling, and animation. (EX1005, 16-17, 22 (“Change the view so as to center the first of the specified items so the largest dimension of its bounding box fills the specified amount of screen (z).”), 34 (“A frame is a bounding box, and all coordinates are specified within the unit square where the unit square is mapped to the frame.”), 41 (“layout prevents any overlapping of nodes by calculating the bounding box of the subtree rooted at a node, and laying out nodes so these bounding boxes do not intersect”), 65-66, 73, 81.) Pad’s layout logic itself relies on bounding boxes by calculating the bounding boxes so that they do not intersect. (EX1005, 38-41.) Pad also describes groups as geometric entities with their own bounding boxes for grouping other items. (EX1005, 59-60 (“Items of type group are special items that group other items...if the event is within the bounding box of the group, but does not hit any members, then it will be ignored by the group.”).)

56. SVG describes an environment for computing object bounds during layout, and describes coordinate systems in relation to CSS positioning. (EX1008, 4 (“the elements and attributes of an SVG drawing should be scriptable via the DOM”), 5-6, 10, 11 (“Coordinate systems, relationship to CSS positioning”), 14, 15 (“SVG should provide for complete DOM support for all attributes and elements...Supplemental utility methods (e.g., query an object’s bounds, its center, its perimeter as expressed as a path, access to ‘connectable’ points for callouts or connection lines”).) A person of ordinary skill in the art implementing a rendering

environment would have understood “object’s bounds” to mean the bounding box maintained by the rendering engine for each object in the layout.

57. SVG includes a requirement to “query an object’s bounds” (EX1008, 15) and Pad discloses a “bounding box” for objects, groups, and layout (EX1005, 22, 34, 59). SVG further contemplates scripting “to perform layout based on the bounds of a given graphics object,” which would involve generating bounding boxes. (EX1008, 15.) A person of ordinary skill in the art would have been motivated to generate and use these bounding boxes in the HTML browser taught by Zaurus. Zaurus establishes the HTML context in which frames are displayed by its browser’s HTML 3.2 rendering engine. (EX1004, 40.) Zaurus further describes displaying content in defined regions on the screen, frame navigation, and zoom; Zaurus’s frame-by-frame viewing and frame selection interface presupposes that the browser identifies spatial regions corresponding to frame portions and rendered objects to support display of a selected region and user navigation among regions. (EX1004, 38, 40, 48-49.)

58. A person of ordinary skill in the art would have understood that such frames would include spatial elements, and that the HTML elements or the DOM objects would have computed bounding boxes for rendering, testing, and interaction as taught by SVG. In the combination, generating per-object bounding boxes while laying out the web page would have been the ordinary course of operation for the

rendering engine on the Zaurus device. A person of ordinary skill in the art would have been motivated to reliably lay out the Web pages by using SVG, which identifies object bounds as routine in the DOM context, Pad, which uses bounding boxes in its resource constrained environment, and Zaurus, which provides the device for displaying web pages frame by frame.

(4) [1.f.4.a] “storing information that links each HTML object with its corresponding page layout information,”

59. In the Zaurus-Pad-SVG combination, each HTML object includes information that links its geometry and attributes to its position on the page. Pad associates each item (including HTML/group items) with its geometry and configuration data (e.g., bounding boxes and grid-position pixel offsets) addressable by tag/id, and SVG’s DOM model assigns unique IDs and exposes per-element attributes/bounds. (EX1005, 2, 16-17, 54-55, 60; EX1008, 9, 14-15.) Given Zaurus’s handheld browser context (frames, caching/offline storage, clickable maps, and reduced/magnified views), the combination would have stored “information that links each HTML object with its corresponding page layout information” for rendering and interaction. (EX1004, 6, 38-43, 46-49.)

60. Zaurus’s disclosed browsing functionality (including cached browsing/offline viewing, hyperlink activation by touching underlined text, and client-side clickable maps) depends on maintaining associations between logical

page constructs (e.g., links/regions) and the rendered geometry presented on the screen so that user inputs (touch/stylus taps) can be mapped to the correct logical target and corresponding action. (EX1004, 6, 10, 40-41, 43.) A person of ordinary skill in the art would have understood that these interactions require the rendering engine to retain, for rendered objects, identifiers and associated geometric/layout metadata (e.g., bounds and position) sufficient to (i) detect whether a user input hits a rendered region and (ii) dispatch the corresponding action (e.g., follow a link or activate a map region). (EX1004, 10, 40.) Zaurus describes that the browser maintains a cache containing frames associated with displayed content such as images. (EX1004, 6, 10, 40-43, 48.) A person of ordinary skill in the art would look to SVG and Pad to establish object and layout associations for the images and frames.

61. Pad provides conventional mechanisms for precisely this association: items have identity via tags/ids and support storing “application-specific information” with an item (e.g., link targets, URLs, or element identifiers), while item geometry and placement are available through bounding box and placement/view APIs. (EX1005, 2, 17, 30, 54-55, 60; EX1007, 107-108.) For example, Pad’s tag/id mechanisms and per-item query commands support linking an item’s logical identity to its geometry and configuration data (including bounds and placement), while event bindings rely on those same associations to invoke item-

specific behaviors when user events occur. (EX1005, 17, 54, 82.) Pad describes associating nodes with layout information, and adjusting the view based on associations between objects and layout. (EX1005, 16, 17 (“returns a list...giving the bounding box”).) Each object stores its position, size, and other metadata information to associate the objects and allow them to be retrieved. (EX1005, 2, 17.) Pad further discloses objects with their own bounding geometries for event routing. (EX1005, 54 (“If the event is within the bounding box of the group, but does not hit any members, then it will be ignored by the group.”).) Each box is associated with a region on the display grid. (EX1005, 54 (“The bounding box (in pixels) is returned for the space occupied by the grid position indicated by column and row.”).) The object’s box having an association with the display grid provides the per item storage.

62. SVG similarly teaches and/or suggests maintaining element-to-geometry associations in a DOM-addressable vector graphics environment: groups/objects have unique IDs and associated attributes (including transformations and visibility) accessible via scripting, and the environment provides utility methods to query object bounds, supporting retrieval of per-object geometry for selection and interaction. (EX1008, 9, 14-15; EX1035.) SVG further indicates that links may be assigned “to any graphic object or group,” reinforcing that rendered vector objects remain linked to logical link targets and can be activated as interactive objects.

(EX1008, 14.) Accordingly, in the combined handheld browsing context, a person of ordinary skill would have been motivated and would have recognized that it would have been routine to store information linking each HTML object (or corresponding DOM element/item) to its page layout information (including bounds and transforms), to enable rendering and interaction. (EX1004, 40-43, 46-49; EX1005, 2, 17, 54-55, 60; EX1008, 9, 14-15.) SVG describes elements and geometric utilities for storing and associating layout attributes and elements. (EX1008, 14-15.) The objects are scriptable via DOM and objects include attributes associated with the layout. (EX1008, 9 (“Groups should have the following attributes (among others), all of which are accessible and/or controllable via scripting: i. Unique ID ii. z-index for explicit layering iii. visibility iv. transformations v. opacity”), 14 (“SVG should provide for complete DOM support for all attributes and elements...Supplemental utility methods (e.g., query an object’s bounds, its center, its perimeter as expressed as a path, access to ‘connectable’ points for callouts or connection lines).”), 15.)

63. Given that (i) Pad discloses per-object stored bounding boxes addressable by tag/id (and applying to HTML items) (EX1005, 17, 54, 55, 60, 82), (ii) SVG discloses DOM access and utilities to query an object’s bounds for selection and zoom/pan (EX1008, 3, 4, 15), and (iii) the Zaurus handheld browser includes offline storage, clickable maps, and reduced/magnified views (EX1004, 40-41, 43, 46-47), a person of ordinary skill would have stored information that links each

HTML object with its corresponding page layout information in a mobile browser (“storing information that links each HTML object with its corresponding page layout information”).

64. Starting with the Zaurus HTML 3.2 browser that renders frames and other HTML elements, a person of ordinary skill in the art would have been motivated to incorporate the conventional techniques from SVG and Pad to maintain per-element layout information associated with DOM objects. The Zaurus HTML 3.2 browser supports processing frames, which are multi-region layouts where DOM elements and objects are associated with bounding boxes. (EX1004, 39-40, 43, 48-49.) Zaurus describes that its browser implements HTML 3.2 and renders complex page structures such as frames, which establishes the HTML/DOM context for object and layout linkage. (EX1004, 38-40, 43.) A person of ordinary skill in the art would have understood that the browser would need to maintain per-element layout information (e.g., frames and bounding boxes) that are associated with DOM objects.

65. SVG supports DOM and object bounds to store and retrieve geometry (e.g., layout) information for each object. (EX1008, 14.) Pad stores the object layout attributes and retrieves the bounding boxes. (EX1005, 17.) SVG explains that objects with identifiers and layout attributes (e.g., z-index, transformations) enable the querying of an object's bounds. (EX1008, 9, 14-15.) Each object is linked to its

layout information because the stored per-object identifiers and layout attributes (e.g., z-index, transformations) allow the object to be retrieved. (*Id.*) A person of ordinary skill in the art would have incorporated the Pad and SVG techniques for storing links between elements and layouts into the handheld browser of Zaurus (“links each HTML object with its corresponding page layout information”).

(5) [1.f.4.b] “wherein the page layout information further includes information from which a page layout location of each of the bounding boxes can be determined;”

66. In the Zaurus-Pad-SVG combination, the layout information includes coordinates in a coordinate system for determining each object’s location on the Web page. Zaurus describes rendering frames, images, and links in specific positions, which would require using page layout locations. (EX1004, 39-40, 43.) The Zaurus browser describes layout regions because frames are segmented layout regions and because the user interface enables the selection of which frame region is shown. (EX1004, 40 (“Pages composed of frames can also be viewed by displaying them frame by frame.”), 48 (“On this product, a white frame icon is displayed on the upper right of the screen for pages with a frame, and when changing the frame to be displayed on the screen, switching is done by touching the portion of the frame icon you would like to display. The touched portion turns grey, and information in the selected portion (child page) is displayed.”), 49.) Zaurus further discloses client-side maps and frame navigation that requires knowing the page coordinates of

regions (e.g., rectangles or polygons) relative to the page layout to detect the touch inputs and to render the elements correctly on the small touch display. (EX1004, 40.) A person of ordinary skill in the art would recognize that frame by frame viewing and selections requires stored layout locations to indicate the position of where the elements are to be displayed in each frame, and look to Pad and SVG for bounding boxes with page coordinates.

67. In Pad, the bounding box is returned in coordinates, and the grid returns coordinate offsets with width/height, which together enable the computation of the page/screen location of the bounding box (“information from which a page layout location of each of the bounding boxes can be determined”). (EX1005, 17, 55.) Pad describes that object positions and sizes via bounding boxes are computed to prevent overlap. (EX1005, 17, 29, 41, 54.) The bounding box includes values, axes, grid cell locations as pixel offsets, coordinates, and window view parameters. (EX1005, 29 (“Returns a list with four elements giving the bounding box of string if it is drawn with the drawtext command. The list has the form ‘x1 y1 x2 y2’ such that the text is within the region bounded by x1 on the left, x2 on the right, y1 on the bottom, and y2 on the top.”).) The view can be exposed and enables mapping the window and screen points to page coordinates to computationally link each item’s bounding box to a precise location. (EX1005, 30 (“Returns the current view of the main window in “xview yview zoom” form. Here, (xview, yview) specifies the point at the center of

the window, and *zoom* specifies the magnification.”), 54 (“The bounding box (in pixels) is returned for the space occupied by the grid position indicated by column and row. The return value consists of integers. The first two are the pixel offset from the master window (x then y) of the top-left corner of the grid cell, and the second two are the width and height of the cell.”).) Related calls for the bounding box and its size, translation, and magnification demonstrate the stored usage of numeric data needed to compute where those boxes are located on the rendered page. (EX1007, 108 (“pan and zoom until that text ... appears near the top” and the text is “acting as a hyperlink to a location on the Pad++ surface.”).)

68. SVG’s transforms and viewport positioning disclose mapping each object’s local bounding box into its absolute page location (“information from which a page layout location of each of the bounding boxes can be determined”). (EX1008, 11, 12.) SVG describes using CSS positioning for establishing a viewport, transformation matrices, and coordinates defined with number values so each object would be located on the coordinate system. (EX1008, 9-10, 11 (“Arbitrarily nested 2x3 matrix transformations should be available. This facility is necessary to ensure compatibility with existing artwork and authoring tools...2x3 matrices are sufficient to achieve any types of translation, scaling, rotation and skewing... SVG should use CSS positioning for establishing a viewport for an outermost SVG element, such as an <svg>...</svg> complete drawing or an SVG fragment within an XML Web

page... SVG will support local user coordinates”), 12-15.) The coordinates and transformations enable a rendering engine to map each object’s bounding box to a specific location within the viewport.

69. A person of ordinary skill in the art would have been motivated to implement the Pad and SVG techniques for storing bounding boxes and page layout positions to enable determining the object’s location for the Zaurus browser to display frames and handle zoom, pan, transforms, and layout preservation. Zaurus maintains region locations within the page to handle frame rendering and region selection on the page by providing a browser that displays frames whose selection and display is region specific, which requires maintaining region specific page locations. (EX1004, 40, 48.) Pad provides numeric layout data in the form of bounding boxes and pixel offsets to return object positions (e.g., *x1 y1 x2 y2*) and window view parameters (e.g., *xview yview zoom*) in a coordinate system; these values determine each object’s position on the displayed page. (EX1005, 29, 54.) SVG provides a viewport via positioning, coordinates, and transformations for computing page positions for a rendering engine to map the bounding boxes to page layout positions. (EX1008, 11.) Combining object bounds with the positions, coordinates, and transforms to determine each object’s page layout location would have been routine and straightforward. In the combination, all this layout information (e.g., object bounds, views, coordinates, transform) identifies the

location of the bounding boxes (“information from which a page layout location of each of the bounding boxes can be determined”).

viii) [1.g] **“translate the first representation of the HTML document to generate a scalable vector representation of the HTML document wherein generating the scalable vector representation includes,”**

70. In the Zaurus-Pad-SVG combination, the HTML layouts would be translated into SVG based scalable vector representation. The Zaurus device renders HTML Web pages by interpreting the HTML specifications to generate an initial interpreted layout (“first representation of the HTML document”). (EX1004, 10 (“A separate page which is linked is displayed by touching the underlined characters.”), 40 (“The Internet browser in the MI-100 series incorporates HTML 3.2 specifications in important places, and therefore has rich expressive capabilities...Pages composed of frames can also be viewed by displaying them frame by frame.”).) A person of ordinary skill in the art would have understood that the rendering engine of the combined device would “translate” this first representation into the scalable vector representation described by Pad and SVG (e.g., converting the HTML layout objects into SVG vector objects) to enable the zooming features.

71. Pad suggests converting HTML documents for further processing (“a scalable vector representation”). Pad discloses “a Tcl file” that defines default bindings for HTML items which “lay them out with scale.” (EX1005, 60.) Pad

discloses automatically fetching a file from a URL, parsing the HTML data, and creating an HTML item to render the page, such as with anchor items inside the group. (EX1005, 60-63 (HTML items), 63-64 (HTML anchors).) Pad supports multi-column layouts via its “grid” items, which “arrange one or more items in rows and columns and [treat] them as a group,” with commands to configure columns, rows, spans, and padding. (EX1005, 53-56.) Pad further discloses vector primitives and a rendering pipeline for the HTML document to be translated. (EX1005, 6 (“drawline...drawpolygon...drawtext”).)

72. Pad demonstrated a zoomable UI and a “Pad++ web browser” suggesting “a zooming version of Mosaic and Netscape” (EX1007, 86.) Pad shows Web pages laid out as sub-documents in columns that the user simply zooms in to read. (EX1007, 91-93.) Pad also explains semantic zooming where objects render differently at different scales (EX1007, 85), which a person of ordinary skill in the art would have understood is scalable. A person of ordinary skill in the art would have understood Pad’s rendering is scalable because items are rendered with “zoomaction” commands that execute when an item “grows or shrinks” and commands that scale text and images so that they resize smoothly with the coordinate frame. (EX1005, 34-35, 53-54, 81-82). A person of ordinary skill in the art would have understood that Pad is translating the HTML data into vector items such as text, images, and anchors on a 2D representation because each HTML element is

represented by vector shapes, text objects, and images positioned within the coordinate system, so zooming would involve re-rendering those scalable objects rather than re-rasterizing a bitmap of pixels.

73. SVG is directed to a Web-standard Scalable Vector Graphics format that can convert raster graphics on mobile devices for improved scaling. (EX1008, 1 (“The W3C has chartered a Scalable Vector Graphics working group to produce a specification for an SVG format, written as a modular XML tagset and usable as an XML namespace, which can be widely implemented in browsers and authoring tools and which is suitable for widespread adoption by the content authoring community as a replacement for many current uses of raster graphics. This will mean that the 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. This will be a significant advance in Web functionality. Interoperability, both in multiple browsers across multiple platforms and in multiple authoring tools (both read and write), is a prime focus.”).) Raster graphics are digital images made up of pixels that become blurry when enlarged without vectorization because pixels are challenging to scale without degrading quality. (EX1036.) A person of ordinary skill in the art would have recognized that such object-based representations are naturally compatible with a scalable vector representation because object geometry (e.g., positions, bounds,

and transforms) can be scaled and re-rendered without the pixelation artifacts that occur when scaling raster images. (EX1036.) SVG supports documents of text data, image data, links, CSS support, coordinate systems, and zoom interactions on systems such as mobile devices or printers. (EX1008, 2-5.)

74. SVG discloses converting elements (e.g., shapes, text, images, links, transforms, coordinates, scale, references) from static graphics files into the scalable vector representation. (EX1008, 5 (“Feature set is rich enough such that a reasonable conversion of most existing standard static graphics files into SVG is possible”), 6 (“For example, a picture of an arrow expressed in SVG might appear inline within an XML file. The expected way this would be achieved would be by referencing the SVG namespace within the XML file.”), 7 (“Vector graphics shapes...Text data”), 8 (“Image data...The same image formats available to HTML should be available to SVG.”), 11 (“Transformations...Coordinate systems, relationship to CSS positioning”), 14 (“Zoom and pan...An SVG processor should support zoom and pan on graphics, with true scalability. Thus, all objects and attributes (including such things as text and linewidths) should grow/shrink uniformly with the zoom level...Links.”).)

75. Pad discloses performing a translation by fetching and parsing HTML and creating HTML items made of vector items that include text, images, and links. (EX1005, 60-64.) SVG provides the standardized scalable vector representation for

web content, which covers text, images, links, CSS styling, coordinate systems, and zoom. (EX1008, 3-4.) SVG is a standard blueprint for Web design that defines various components (e.g., paths, text, images), behaviors (e.g., links), and geometry (e.g., transforms, coordinates) to achieve “true scalability” for zoom and pan. The Zaurus browser handles HTML pages (e.g., frames, images, and links) on a small display, where vector representations would enable resolution-independent zooming and panning. (EX1004, 40.)

76. In view of (i) SVG’s disclosure of a browser-implemented, and device-independent scalable vector graphics with zoom/pan, DOM access, coordinate systems, and text/image support (EX1008, 1, 3-5, 7, 15), (ii) Pad’s demonstration of zoomable web browsing (EX1007, 85-86, 91-93), and (iii) Zaurus’s small screen mobile rendering motivating a scalable solution (EX1004, 39-41), a person of ordinary skill in the art would have translated the browser’s first HTML layout representation into a scalable vector representation based on SVG’s suggestion that its “feature set is rich enough such that a reasonable conversion of most existing standard static graphics files into SVG is possible.” (EX1008, 5.) A person of ordinary skill in the art would have understood that this translation (conversion) was necessary to achieve the resolution-independent zooming and panning taught by Pad and SVG on the Zaurus device. Thus, the combination

discloses or suggests “translating the first representation of the HTML document to generate a scalable vector representation of the HTML document.”

77. SVG describes browser integrated scalable vector graphics, so a person of ordinary skill in the art would have been motivated to incorporate SVG into the Zaurus browser to translate HTML layouts into scalable vectors to enable zooming while preserving layout. Zaurus provides the HTML rendering (e.g., Web page with links and frames). (EX1004, 9, 40.) SVG contemplates conversions from other formats and stating that its “feature set is rich enough” for “reasonable conversion” of existing graphics into SVG, which would have motivated a person of ordinary skill in the art to apply SVG to translate from HTML content layouts. (EX1008, 6.) SVG further defines scalable vector representation and describes that generating SVG includes vector paths, text, images, links, transforms, coordinates, and “true scalability” under zoom and pan. (EX1008, 1, 4-7, 10, 13.)

78. A person of ordinary skill in the art would have taken the HTML layout (“first representation”) and created a scalable vector representation based on SVG and Pad. Translating rendered HTML layouts into scalable vector representation would have been a routine application of conventional techniques, which a person of ordinary skill in the art would have been motivated to do to improve readability and interactions on the small Zaurus screen. In the combination, the browser would render the HTML Web page and translate its layout into a scalable output that

Declaration of Dr. Benjamin Goldberg

U.S. Patent No. 10,083,154

includes its elements such as shapes, text, images, transforms, coordinates, and links.

That “scalable vector representation” would also include one or more vectors, e.g., for each HTML object a stored (x, y) translation offset (vector) represented by (X2, Y2) relative to an origin point (X1, Y1), where (X1, Y1) is the page-level origin (“primary datum”) and (X2, Y2) is the object’s anchor/placement point (“object datum”), as part of the object’s placement/transform data.

(1) [1.g.1] “defining a primary datum corresponding to the interpreted page layout;”

79. In the Zaurus-Pad-SVG combination, generating the scalable vector representation includes defining an origin. Pad discloses a global coordinate system for the Pad++ surface with an origin point that provides a base coordinate reference for the page layout so that object locations can be expressed as coordinates in that global system. (EX1005, 8 (“creates a line from the origin to the point”).) Pad implements page and view coordinate frames in that global coordinate system; the page-level coordinate reference (e.g., fixed global origin (0,0) and axes of the Pad coordinate system) provides the “primary datum” corresponding to the interpreted page layout. (EX1005, 8 (“creates a line from the origin to the point”), 30 (“Returns the current view of the main window in ‘xview yview zoom’ form. Here, (xview, yview) specifies the point at the center of the window, and zoom specifies the magnification.”).)

80. The “padxy” routine maps window coordinates to Pad coordinates, demonstrating a fixed underlying coordinate system with an origin and axes. (EX1005, 5 (“padxy[52] Convert a window point (x, y) to pad coordinates”), 33.) The coordinates form frames that are further defined by a bounding box to which the unit square is mapped, so that “all coordinates are interpreted relative to the frame.” (EX1005, 12 (“coordinates on the Pad++ surface”), 13 (“you can zoom into the position”), 23 (“If the flag -objectcoords is specified, then all coordinates are returned in the item’s local coordinate system... If not specified, then all coordinates are returned in the global coordinate system.”), 34.) The grids return a “pixel offset” relative to the corner and width and height, which demonstrates fixing the origin from which the layout positions are measured. (EX1005, 54; EX1007, 107 (“a view onto the infinite Pad++ surface”).)

81. A person of ordinary skill in the art would have understood that the origin and axes of Pad’s fixed global coordinate system define the page layout space (“corresponding to the interpreted page layout”). The view (xview, yview, zoom) operates on that coordinate system for navigation/rendering and does not define the underlying datum itself. Consistent with SVG’s viewport/user-coordinate establishment, Pad’s fixed global coordinate reference (origin/axes) provides the “primary datum” corresponding to the interpreted page layout. (EX1005, 5, 30, 33; EX1008, 11.)

82. SVG describes a coordinate system with origins and transformations for the page. (EX1008, 11 (“Coordinate systems, relationship to CSS positioning”).) The vector representation defines its viewport and user coordinates as a base layout. (*Id.*) SVG’s viewport/user-coordinate system defines a fixed origin point at an X, Y coordinate for the drawing/page coordinate space (“primary datum”). SVG further requires zoom and pan with true scalability. (EX1008, 14.) A person of ordinary skill in the art would have understood that the viewport and coordinates (“primary datum”) correspond to the page layout (“corresponding to the interpreted page layout”).

83. In view of SVG’s outermost viewport and root coordinate system; Pad’s global pad-surface frame with origin (0, 0) and conversions between global and object coordinates; and Zaurus’s magnification of HTML pages, a person of ordinary skill in the art would have been motivated to define a primary datum corresponding to the interpreted page layout, such as page-level global origin/reference point (0,0) in the underlying coordinate system, and measure all object positions relative to that datum (“defining a primary datum corresponding to the interpreted page layout”). Given that Pad and SVG provide the origin in their coordinate systems, a person of ordinary skill in the art would have been motivated to implement such techniques in the Zaurus browser. Pad defines the origin and the global coordinate system used for layout, which is the datum based on layout

coordinates. (EX1005, 29.) SVG describes the viewport and user coordinates, which define a reference for a vector. (EX1008, 11.) Together, the origin, viewport, and user coordinates for the window all provide a person of ordinary skill in the art with a primary datum corresponding to the layout. Zaurus already downloads and renders HTML pages; substituting or augmenting the raster layout with a vector pipeline to support zoom requires defining the page's viewpoint or coordinate origin so that the objects (e.g., text or images) can be positioned and scaled as taught by Pad and SVG.

(2) [1.g.2] “for each HTML object,”

(a) [1.g.2.a] “defining an object datum corresponding to a layout location datum for the HTML object’s bounding box;”

84. In the Zaurus-Pad-SVG combination, each object would have a defined placement reference point (anchor) and an absolute placement coordinate for that anchor in page space. Pad discloses per object bounding boxes in a placement datum (anchor) used to locate the object in the page layout. In Pad's placement model, the “object datum” is the item's anchor reference point, while the “layout location datum” is the placement/positioning coordinate set by “-place” at which that anchor point is placed. (EX1005, 17 (“the coordinates of a sticky item that would appear at the same location on the screen is returned”), 47 (“AnchorPos tells how to position the object relative to the positioning point... e.g., ‘n’ puts the object’s top-center

point at the positioning point.”.) In Pad’s placement model, the “object datum” corresponds to the object’s anchor reference point (as set by “anchor”), and Pad’s placement model uses a positioning point together with an object anchor reference; Pad explains that “AnchorPos tells how to position the object relative to the positioning point,” which describes placing a defined point of the object at a defined layout location (“layout location datum”). (EX1005, 47.)

85. Pad discloses a bounding box for each item and a placement transform with coordinate translations and a designated anchor such that every object has both a bounding box and a specific internal reference point (datum) that the system uses when positioning the object on the layout. (EX1005, 2-3, 22 (recenter the view so the object or its bounding box fills the screen and place the item’s center at a specified screen position), 38 (dx/dy applied to the item transformation, i.e., the “-place itemconfigure option,” and reporting position at the item’s anchor point), 47-50.) Pad further discloses coordinate frames where a “frame is a bounding box, and all coordinates are specified within the unit square where the unit square is mapped to the frame,” which reinforces the association between a coordinate origin and a particular object’s bounding box. (EX1005, 34-35 (“resetcoordframe”), 53 (push/pop).) Pad further describes that the bounding boxes provide layout data for items (EX1005, 29 (“Returns a list with four elements giving the bounding box of string if it is drawn with the drawtext command. The list has the form ‘x1 y1 x2 y2’

such that the text is within the region bounded by x1 on the left, x2 on the right, y1 on the bottom, and y2 on the top.”).) A person of ordinary skill in the art would have understood that the anchor points are per-object frames that are anchored to bounding boxes.

86. SVG discloses per-object coordinates and transformations, including an implementation to query an object’s bounds. SVG discloses that each element is in the user coordinate system and that transforms can be applied to the elements to establish the object level datum. (EX1008, 9 (“Groups should have the following attributes (among others), all of which are accessible and/or controllable via scripting: i. Unique ID ii. z-index for explicit layering iii. visibility iv. transformations v. opacity”), 11 (“Arbitrarily nested 2x3 matrix transformations should be available...SVG will support local user coordinates”), 14 (“Supplemental utility methods (e.g., query an object’s bounds, its center, its perimeter as expressed as a path, access to ‘connectable’ points for callouts or connection lines”).) A person of ordinary skill in the art would have understood that each object has its own coordinates and its bounds can be queried, the per-object datum is defined with respect to the layout.

87. In the Zaurus-Pad-SVG combination, after the HTML/CSS page is parsed and elements are grouped into rendered objects, each such object can be materialized as a vector item at the per object level. (EX1005, 3, 17, 47; EX1008,

3, 11, 16.) A person of ordinary skill in the art would have been motivated to incorporate vector transformations of each object as taught by SVG and Pad into the Zaurus browser. Pad associates anchors with bounding boxes (“for each object”) and SVG requires per-object coordinates and object bounds queries (“for each object”). (EX1005, 2-3, 34-35; EX1008, 9, 11, 14.) SVG describes user coordinates, object transforms, and object bounds, (EX1008, 9, 11, 14), which a person of ordinary skill in the art would have understood is a per-object datum model for a Web rendering engine (“defining an object datum”). Thus, for each rendered object, Pad/SVG provide (i) an internal object reference point (“object datum”, e.g., anchor within the bounding box) and (ii) a placement coordinate (“layout location datum”) in the page coordinate system; the object datum is positioned at that placement coordinate for rendering and subsequent transformations.

88. A person of ordinary skill in the art would have understood that after a layout engine computed a bounding box for each object, defining the object’s reference relative to the bounding box would enable the mathematical baseline for further zooming and panning. (EX1005, 6, 12-13, 24.) Pad discloses that an anchor is defined for every item, which is a stable reference point inside the object used by the transform to position that object. (EX1005, 38, 47-50.) Pad provides a coordinate frame and uses the item’s bounding box as that frame, so each coordinate frame is a bounding box such that the object’s datum corresponds to the box

(“defining an object datum”). (EX1005, 47.) SVG requires user coordinates, per-object transforms, and querying an object’s bounds; associating the object’s datum to its box location would enable placement and scaling (“defining an object datum”). (EX1008, 15.)

89. Pad discloses that each item’s bounding box can be mapped to the coordinate frame of that bounding box, so that coordinates are defined with respect to the box (“corresponding to a layout location datum for the HTML object’s bounding box”). (EX1005, 34, 35, 38, 47-50.) The routine data was also available because Pad provides the coordinates of the bounding boxes (EX1005, 29), so it would have been straightforward to set the object’s datum at that box to a frame mapped to that box. (EX1005, 18, 22-23, 47.) SVG further recognizes per object coordinate systems and transforms so that a datum can be defined and transformed with the object, and discloses object bounds as values that can be defined or computed for a datum to be mapped to a bounding box for layout (“corresponding to a layout location datum for the HTML object’s bounding box”). (EX1008, 11.)

(b) [1.g.2.b] “generating a vector from the primary datum to the object datum for the HTML object’s bounding box; and”

90. As I explained above, the Zaurus-Pad-SVG combination discloses and/or suggests the primary datum and the object datum corresponding to the object’s bounding box. (See Sections IX.A.3.viii.1-viii.2(a).)

91. Pad discloses bounding boxes and vectors for positions. Pad discloses a bounding box command that returns a bounding box as four coordinates, which can be two corner points for locating an object's box in a plane. (EX1005, 18-22.) The view translation and object movement are provided in vector or coordinate form. (EX1005, 32 ("Change the view so that the point 'xview yview' is at the center of the screen with a magnification of zoom."); EX1007, 97 ("The Pad++ surface as an infinite zoomable space, with objects residing at specific x, y, zoom coordinates within that space.")) The API uses the term "vector" to move points. (EX1005, 82 ("lineto vector [] Specifies a straight line in the current path from the current point to (x, y) specified by vector. Makes (x, y) the current point. [] moveto vector [] Moves the current point within the current path to (x, y) specified by vector.")) Pad further applies a translation vector to locate each object relative to the origin by manipulating the positions of the translation on per-item transformation. (EX1005, 38 ("This command returns a string with the (x, y) position at the item's anchor point.")) When computing bounding boxes, Pad explains that the bounding box script should return two-element vectors that specify their coordinates, which demonstrates that Pad applies vectors to represent object-location data. (EX1005, 65 ("A Kpl script that will be evaluated to compute the bounding box of this item. It should return two-element vectors that specify (x1, y1), (x2, y2) which are the lower left and upper right corners of this item's bounding box.")) Coordinates that

are local are distinguished from those that are global coordinates, which enables subtraction. (EX1005, 23.)

92. SVG discloses establishing a viewport for SVG elements based on CSS positioning. (EX1008, 11.) The transformation matrices and scaling attributes enable vector associations from the origin to the object. (EX1008, 11 (“Arbitrarily nested 2x3 matrix transformations should be available. This facility is necessary to ensure compatibility with existing artwork and authoring tools...2x3 matrices are sufficient to achieve any types of translation, scaling, rotation and skewing”).)

93. In the Zaurus-Pad-SVG combination, Pad discloses the origin point and SVG discloses establishing a viewport (“primary datum”). (EX1005, 30-32; EX1008, 11.) Pad discloses bounding boxes for objects and SVG discloses retrieving an object’s bounds (i.e., bounding-box/bounds information from which the HTML object’s bounding box and its layout-location datum can be determined) (“object datum for the HTML object’s bounding box”). (EX1005, 17; EX1008, 15.) Pad defined the frames as specified by bounding boxes. (EX1005, 18-22, 66). SVG also disclosed that it “will support local user coordinates,” which enables each group or object to have its coordinate frame relative to a parent frame to establish vectors between primary and object datums. (EX1008, 11.) Consistent with that disclosure, the “object datum” is a predictable reference point for the object (e.g., the object’s anchor point) used for placement relative to its bounding box and at its layout

location. A person of ordinary skill in the art would have understood that the vector is the translation offset (x, y) represented in a transform (e.g., SVG's 2x3 matrix) to move from the page-level reference ("primary datum") to the object's anchor reference point ("object datum"). Pad places each item by positioning its anchor point at "-place" coordinates in the global Pad coordinate system, so the item's position is represented as a translation offset (x, y) from the page-level origin/reference ("primary datum") to the item's anchor/reference point ("object datum"). (EX1005, 30-32, 17, 18-22, 66, 38; EX1008, 11, 15.) Pad's slide command confirms this translation-vector model: it applies dx/dy to the item transformation (the "-place itemconfigure option") and reports the resulting position at the item's anchor point, which is the updated (x, y) translation from the page reference ("primary datum") to the object's anchor datum ("object datum"). (EX1005, 38.)

94. In both Pad and SVG, the position of an object relative to the page origin is represented as a coordinate translation, which is a vector difference between the page datum and the object datum. (EX1005, 38; EX1008, 11.) The combination of Pad and SVG would use a stored vector transform from origin to object to manage layout, which a person of ordinary skill in the art would have incorporated into Zaurus for a vector representation.

95. After establishing the primary datum (e.g., origin) and the object datum (e.g., frame defined by the object's bounding box), computing the vector between them would be an ordinary translation or mathematical operation that would enable further rendering and transformation ("generating a vector from the primary datum to the object datum"). (EX1005, 23, 30.) A person of ordinary skill in the art would have understood how to generate the vector because SVG describes the necessary vector mathematics (EX1008, 11) and Pad already used numeric views and commands that demonstrate routine translation operations (EX1005, 32). Specifically, SVG describes 2x3 transformation matrices that "are sufficient to achieve any types of translation," where the translation component of the matrix (typically the last column) mathematically represents the vector offset from the coordinate system's origin to the object's position. (EX1008, 11.) This "vector" can be expressed as the translation/offset (x, y) from the page/viewport reference point ("primary datum") to the object's reference point at its layout location ("object datum"), consistent with Pad's coordinate operations and SVG's translation component of a 2x3 transform.

(c) [1.g.2.c] "creating a reference that links the HTML object to the vector that is generated; and"

96. The Zaurus-Pad-SVG combination describes using an object identifier to associate each HTML object with the vector.

97. Zaurus discloses a handheld HTML browser with images and client side maps. Zaurus implements HTML 3.2, renders images and animated GIFs, supports client side clickable maps, and supports caching and fetching pages by URL. (EX1004, 6, 10, 40.) A person of ordinary skill in the art would have looked to Pad and SVG to implement and improve the client side image maps because they would require the device to maintain a mapping from the HTML anchor areas to the clickable regions drawn on the screen, which demonstrate a reference from an HTML element to the rendered areas that detect touch inputs (“creating a reference that links the HTML object to the vector that is generated”).

98. Pad discloses vector items that include identifiers, tags, and arbitrary application data. Pad discloses identifies and tags for each item including a “place to store application-specific information with an item.” (EX1005, 2.) The same options table lists links, anchors, and other options that demonstrate that the items are associated with identifiers. (EX1005, 2.) The bounding box and other per-item queries used in rendering further demonstrate each item’s unique identity (“creating a reference that links the HTML object to the vector that is generated”). (EX1005, 17 (“Returns a list with four elements giving the bounding box for all the items named by *tagOrId* argument(s). The list has the form ‘x1 y1 x2 y2’ such that the drawn areas of all the named elements are within the region bounded by x1 on the left, x2 on the right, y1 on the bottom, and y2 on the top....This command associates

command with all the items given by *tagOrld* such that whenever the event sequence given by sequence occurs for one of the items the command will be invoked”). Pad further discloses object configurations that include transforms and an association between the object and its bounding box or transform. (EX1005, 30 (“Returns the current view of the main window in ‘xview yview zoom’ form.”), 60 (“HTML items are an extension of group items, and thus have several of the same options as groups.”), 80 (“drawline vector [] Draws a line specified by vector. As Kpl vectors may be up to 16-dimensional, this vector can specify up to 8 (x, y) points. This routine will draw a line connecting as many points as are specified within *vector*.”), 82 (“lineto vector [] Specifies a straight line in the current path from the current point to (x, y) specified by vector. Makes (x, y) the current point. [] moveto vector [] Moves the current point within the current path to (x, y) specified by vector.”); EX1007, 107 (“links to specific items are established and broken continually as the portal’s view changes”), 108 (“Any object can be turned into a hyperlink”).)

99. SVG discloses DOM linkage and linking on any graphic content. SVG requires “complete DOM support for all attributes and elements” and other techniques such as querying an object’s bounds. (EX1008, 15.) SVG contemplates that it “should be possible to assign a link to any graphic object or group.” (EX1008, 14.) SVG further describes 2x3 transformation matrices and scaling attributes to enable vector associations from the origin to the object. (EX1008, 9 (“Groups

should have the following attributes (among others), all of which are accessible and/or controllable via scripting: i. Unique ID”), 10, 11 (“Arbitrarily nested 2x3 matrix transformations should be available. This facility is necessary to ensure compatibility with existing artwork and authoring tools...2x3 matrices are sufficient to achieve any types of translation, scaling, rotation and skewing”), 12 (“These transformations should apply to all object types (shapes, images and text) in a uniform and consistent manner”).) A person of ordinary skill in the art would have understood that these requirements enable an association between DOM elements and the corresponding vector object to support events, linking, and testing (“creating a reference that links the HTML object to the vector that is generated”).

100. A person of ordinary skill in the art would have been motivated to maintain a reference between the object and its transform in the Zaurus browser to vectorize Web pages. As I explained above, Zaurus teaches HTML content with hyperlinks and clickable regions; Pad teaches vector items having persistent identity, tags, and other information for application specific associations; and SVG teaches that vector graphics in the web stack should be DOM-addressable with links on objects. (EX1004, 40; EX1005, 2, 17; EX1008, 15.) A person of ordinary skill in the art would have created and stored a reference from each HTML element to the corresponding vector items in the scalable vector representation to support link activation, testing, zoomed redrawing, and DOM scripting.

101. Zaurus's client side clickable maps mean that the rendering engine must detect taps in a rendered region and associates the corresponding area, which effectively establishes a reference from the HTML to the rendered geometry. A person of ordinary skill in the art would have understood that this function requires the browser to maintain a reference that links the logical HTML element (e.g., the map area definition) to the rendered geometry on the display. Reimplementing the same behavior in a vectorized renderer as taught by Pad and SVG would keep the same mapping, except from DOM elements to vector items. Pad provides the mechanism to store the reference in the vectorized layer. Pad discloses that the link can be for item identifiers and other information designed for application specific information such as a pointer or URL string. A person of ordinary skill in the art would have utilized these attributes to store the DOM node identifier or URL on the vector item, creating a shared tag or reference across all vector items belonging to the HTML element. SVG provides the standard to associate the DOM and graphics. SVG's DOM support and links to any graphic object suggests the association between DOM and graphics. Implementing SVG in a browser would include creating and maintaining references from the DOM to the corresponding vector items so that events, links, inputs, and zoom requests can function.

- ix) **[1.h.1] “render the scalable vector representation of the HTML document on the touch-sensitive display using a first scale factor to display the HTML document at a first zoom level under which the HTML document is displayed to fit across a width of the touch-sensitive display,”**

102. Zaurus discloses a touch-sensitive mobile device that renders Web pages (“touch-sensitive display”). (EX1004, 15 (“Touch the screen with a pen. Let’s try various operations by touching the screen with a pen”).) Zaurus describes zooming controls to switch from reduced to magnified display (“first scale factor...first zoom level”). (EX1004, 46-47 (switching between reduced and magnified displays).)

103. Pad discloses scaling and zooming a vector representation using a selectable scale factor (“render the scalable vector representation...first scale factor...first zoom level”). Pad discloses numeric view and zoom settings that controls the rendering of vector items. (EX1005, 29 (“Returns a list with four elements giving the bounding box of string if it is drawn with the drawtext command. The list has the form ‘x1 y1 x2 y2’ such that the text is within the region bounded by x1 on the left, x2 on the right, y1 on the bottom, and y2 on the top.”), 30 (“Returns the current view of the main window in ‘xview yview zoom’ form. Here, (xview, yview) specifies the point at the center of the window, and zoom specifies the magnification.”), 34 (“A frame is a bounding box, and all coordinates are specified within the unit square where the unit square is mapped to the frame.”).) Pad

discloses zoom capabilities that demonstrate that rendering is vector based and controlled by a zoom factor. (EX1005, 13 (“maxZoom...gives 16 orders of magnitude of zooming”).)

104. Pad discloses fitting the rendered content across the display (“under which the HTML document is displayed to fit across a width of the touch-sensitive display”). 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, 54 (“The bounding box (in pixels) is returned for the space occupied by the grid position indicated by column and row.”).) In Pad’s browser example, selecting a text element “zooms in so that that text fills the screen horizontally and appears near the top of the screen vertically.” (EX1007, 54.)

105. SVG describes vector graphics that are intended to be scaled for rendering Web pages (“render the scalable vector representation”). SVG emphasizes uniform transformation and scaling of shapes, images, and text on a view port established via CSS positioning. (EX1008, 12.) SVG further describes zoom and pan with scalability of text and images. (EX1008, 1 (“The W3C has chartered a Scalable Vector Graphics working group to produce a specification for an SVG format, written as a modular XML tagset.”), 11 (“SVG should use CSS positioning

for establishing a viewport for an outermost SVG element...SVG will support local user coordinates”), 13, 14 (“An SVG processor should support zoom and pan on graphics, with true scalability. Thus, all objects and attributes (including such things as text and linewidths) should grow/shrink uniformly with the zoom level.”).)

106. A person of ordinary skill in the art would have been motivated to combine the Zaurus touch web device with zoomed views, Pad’s zooming and rendering of vector content, and SVG’s scalable and transformable web graphics. On the Zaurus small screen that already supported reduce and magnify operations, usability would depend on presenting an entire page or region legibly without excessive panning. Pad teaches zoom to fit operations driven by a bounding box in a numeric zoom factor, and shows an HTML browsing scenario where content is zoomed so that it fills the width of the screen horizontally. SVG confirmed that vector graphics could be scaled and transformed uniformly to establish zoom-based vector rendering for web content. A person of ordinary skill in the art would have been motivated to incorporate Pad’s zoom to fit vector rendering into the Zaurus browser to improve readability and navigation on a handheld display.

- x) **[1.h.2] “wherein the scalable vector representation of the HTML document is configured to enable a user to view the HTML document at one or more user-defined zoom levels by rendering the scalable vector representation on the touch-sensitive display using one or more respective scale factors in response to associated user inputs made via the touch-sensitive display, and”**
- xi) **[1.h.3] “wherein the interpreted page layout, functionality, and design of the content associated with the HTML document is preserved at each of the first zoom level and the one or more user-defined zoom levels.”**

107. Zaurus discloses user selectable reduce and magnify to control zooming while browsing (“wherein the scalable vector representation of the HTML document is configured to enable a user to view the HTML document at one or more user-defined zoom levels...in response to associated user inputs made via the touch-sensitive display”). (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.”).)

108. Pad discloses iterative zoom that can be set by a parameter to render HTML content (“by rendering the scalable vector representation on the touch-sensitive display using one or more respective scale factors”). (EX1005, 13, 24-28, 29 (“Returns the current magnification of tagOrld for this specific render (it could

be rendered multiple times if visible through different portals).”), 30 (“Returns the current view of the main window in “*xview yview zoom*” form. Here, (*xview*, *yview*) specifies the point at the center of the window, and *zoom* specifies the magnification.”); EX1007, 4 (“In Pad++, clicking and holding on the middle mouse button performs a smooth animated zoom in.”).)

109. SVG discloses zooming with scalability of text and images to preserve functionality and design across zoom levels (“wherein the interpreted page layout, functionality, and design of the content associated with the HTML document is preserved at each of the first zoom level and the one or more user-defined zoom levels”). (EX1008, 10, 11 (“Transformed objects should have the option of exhibiting true scalability (where all attributes should scale uniformly, and linewidths, images and text should scale along with the sizes of the shapes)”), 11 (“These transformations should apply to all object types (shapes, images and text) in a uniform and consistent manner...SVG should use CSS positioning for establishing a viewport for an outermost SVG element...SVG will support local user coordinates”), 13, 14 (“It should be possible to assign a link to any graphic object or group.”).)

110. A person of ordinary skill in the art would have implemented the combination to enable smooth zooming of Web content on handheld screens without compromising legibility or functionality. Zaurus discloses the handheld browser

Declaration of Dr. Benjamin Goldberg

U.S. Patent No. 10,083,154

that already supported touch inputs to perform user-selected magnify and reduce operations (“in response to associated user inputs made via the touch-sensitive display”). Pad illustrates scalability at a variety of magnifications that preserves the layout, functionality, and design of the content displayed in the HTML browser. (EX1007, 86-96.) In the combination, SVG provides the scalable vector representation designed to zoom by numeric scale factors and Pad provides zoom parameters supporting the preservation of the content at a variety of zoom levels (“wherein the scalable vector representation of the HTML document is configured to enable a user to view the HTML document at one or more user-defined zoom levels by rendering the scalable vector representation on the touch-sensitive display using one or more respective scale factors”). SVG’s uniform scaling would preserve layout, design, and interactivity while Pad’s techniques would parse HTML content into interactive objects with anchors that retained their link functionality after zooming (“wherein the interpreted page layout, functionality, and design of the content associated with the HTML document is preserved at each of the first zoom level and the one or more user-defined zoom levels”). A person of ordinary skill in the art would have been motivated to combine these teachings to achieve the benefit of resolution-independent zooming, preserved page layout and design (i.e., uniform scaling), and continued functionality of links at various zoom levels. In the resulting combination, there would be scalable vector representation rendered at user-defined

zoom levels via touch-driven scale factors while preserving layout, functionality (i.e., links), and design.

4. Claim 2: “The mobile hand-held device of claim 1, wherein the mobile hand-held device is a mobile phone.”

111. Zaurus discloses a mobile handheld with cellular connectivity to access the Internet. (EX1004, 7-8 (adapters), 10 (portable), 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, 141.) SVG supports for “small mobile devices.” (EX1008, 1, 4.) In view of the aforementioned teachings, a person of ordinary skill in the art would have implemented the combination on “a mobile phone.”

5. Claim 3: “The mobile hand-held device of claim 1, wherein the mobile hand-held device is a hand-held computer.”

112. Zaurus discloses a mobile handheld computer with cellular connectivity to access the Internet, and a person of ordinary skill in the art would have implemented the combined HTML to vector translation on the handheld computer as I explained above. (EX1004, 10 (portable), 12 (device), 15 (touch screen), 18 (specifications); EX1007, 141; EX1008, 1, 4.)

6. Claim 4: “The mobile hand-held device of claim 1, wherein execution of the instructions further enables the user to zoom in on a user-selectable portion of a view of the HTML

document in response to a user interface input made via the touch-sensitive display.”

113. Pad discloses zooming by enabling the user to click on text so that it fills the screen (“enables the user to zoom in on a user-selectable portion of a view of the HTML document”). (EX1005, 22, 30; EX1007, 4, 54, 61, 108.) SVG discloses supporting zooming and that event handling should be “at least as extensive as what is available for HTML (e.g., mouseovers, mouseclicks).” (EX1008, 14.) Zaurus describes enabling the user to touch the screen to select between reducing and magnifying displayed content (“in response to a user interface input made via the touch-sensitive display”). (EX1004, 10, 14-16, 39-41, 46-47.) In the Zaurus-Pad-SVG combination, the zooming techniques of Pad and SVG would be incorporated into the magnification option of Zaurus so that a user would tap a portion of the HTML page to zoom in on the HTML document.

7. Claim 5: “The mobile hand-held device of claim 1, wherein the views of the HTML document are rendered in real-time to effect zooming operations.”

114. Pad discloses “animated zoom” in response to user input, which demonstrates continuous re-rendering as the user zooms. (EX1007, 4, 13, 22, 31.) Pad also discloses continuous zoom navigation, which demonstrates that zooming is a real-time operation. (EX1007, 66-70, 87-96.) Pad specifically discloses performing “animation” in “milliseconds.” (EX1005, 22-23, 32-33.) SVG discloses “[r]easonable display performance” and “fast performance.” (EX1008, 7.) Zaurus

discloses touch driven HTML browsing. (EX1004, 10-11, 41.) In the Zaurus-Pad-SVG combination, a person of ordinary skill in the art would have understood that high performing zoom that animates in milliseconds would enable the HTML document to be rendered in real-time.

8. Claim 6: “The mobile hand-held device of claim 5, wherein the zooming operations are effected in a fraction of a second.”

115. As I explained above with respect to claim 5, a person of ordinary skill in the art would have understood that high performing zoom that animates in milliseconds enables zooming operations within a fraction of a second.

9. Claim 7

- i) [7.a] “The mobile hand-held device of claim 1, wherein execution of the instructions performs further enables the device to:”**
- ii) [7.b] “generate a vector-based display list associated with the scalable vector representation; and”**

116. I have been asked to interpret “performs further enables” to mean “further enables.” 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, 19:42-44.) Pad discloses a “display list” associated with text, lines, polygons, and images for computing bounding boxes, which is a vector display list representation of objects and their layout. (EX1005, 65.) SVG discloses the scalable

vector representation context for Web content. (EX1008, 3.) In the Zaurus-Pad-SVG combination, the browser would generate the display list to be associated with scalable vector representation.

iii) [7.c] “employ the vector-based display list to render views of the HTML document on the touch-sensitive display at different scale factors to effect zooming operations.”

117. Pad discloses rendering at various zoom levels so that content fills the screen, which demonstrates using the “display list” of vectorized objects to implement zooming. (EX1007, 31-36, 61, 93.) SVG discloses zooming and transformations with uniform scaling. (EX1008, 3, 6, 7.) Zaurus discloses touch driven HTML browsing. (EX1004, 10-11, 40-41.) In the Zaurus-Pad-SVG combination, the list of vectorized objects would be used to render the content at various scales to implement the zooming features.

10. Claim 8: “The mobile hand-held device of claim 1, wherein the HTML document is an HTML parent document for a Web page.”

118. In the Zaurus-Pad-SVG combination, Zaurus describes downloading and displaying HTML 3.2 web pages, which a person of ordinary skill in the art would have understood to be part of a top level “parent” document. (EX1004, 40.)

- 11. Claim 9: “The mobile hand-held device of claim 1, wherein the HTML elements includes an HTML table having a plurality of cells, each cell having at least one of associated text content and associated image content, and the CSS code includes CSS code defining how the table and its associated content is to be rendered.”**

119. Zaurus discloses downloading and displaying HTML web pages by implementing HTML 3.2 to define and display HTML tables, images, and text. (EX1004, 40.) HTML 3.2 is the W3C HTML standard that includes the <TABLE> element and associated attributes for defining table layout. (EX1040.) A browser that “incorporates HTML 3.2” would handle HTML-based Web content that includes HTML code defining the layout of an “HTML table” such as <TABLE> elements. (EX1040.) Pad discloses a grid to arrange objects in rows and columns (“an HTML table having a plurality of cells”) as well as loading text and images (“each cell having at least one of associated text content and associated image content”). (EX1005, 31, 60-61.) SVG discloses handling CSS for presenting Web content to apply properties to text and graphic objects (“the CSS code includes CSS code defining how the table and its associated content is to be rendered”). (EX1008, 3, 8, 11-14.) 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.) In the Zaurus-Pad-SVG combination, given that the Zaurus browser already browsed HTML pages containing tables and SVG disclosed supporting CSS, a person of ordinary skill in the art would have relied on CSS attributes for the tables (e.g., cell alignment, borders, widths, and font styling) to define how the table is to be rendered. The combination would apply CSS styling to the tables (HTML structures) on the resulting device to enable consistent page rendering.

12. Claim 10

- i) [10.a] “The mobile hand-held device of claim 1, wherein the content associated with the HTML document includes a plurality of images that are referenced using respective Universal Resource Locators (URLs) in the HTML document, and wherein execution of the instructions further enables the device to:”**
- ii) [10.b] “for each of the plurality of images;”**
 - (1) [10.b.1] “retrieve the image via the URL for the image; and”**

120. Zaurus discloses retrieving HTML Web pages with images referenced with URLs, and selectable options such as for loading home pages (“the content associated with the HTML document includes a plurality of images that are referenced using respective Universal Resource Locators (URLs) in the HTML document”). (EX1004, 10, 39, 45-47.) Pad discloses HTML items that fetch content and images from URLs. (EX1005, 17, 29-30, 54-61.) SVG discloses linking to

external media via URL and embedded images in Web content. (EX1008, 11, 13, 15-16.) In the Zaurus-Pad-SVG combination, Zaurus's browser would incorporate Pad's URL based HTML and image fetching and SVG's linked and embedded graphics to retrieve the images referenced by URLs ("for each of the plurality of images; retrieve the image via the URL for the image"). A person of ordinary skill in the art would have understood that for each image reference in an HTML page, the resulting device would retrieve the image via its URL to load (or cache) the page.

(2) [10.b.2] "generate a respective bounding box for the image; and"

121. Zaurus provides the mobile HTML browser that renders images. (EX1004, 39, 45-47.) Pad discloses generating bounding boxes for items and coordinates for views; and coordinate frames as bounding boxes for text and images to be scaled so that the height of the image is scaled to the height of the frame. (EX1005, 5, 6, 17-18, 34-35.) SVG discloses treating images as objects to which transformations could be applied, as well as links to external media. (EX1008, 9, 11, 15.) In the Zaurus-Pad-SVG combination, the image objects would have their geometry and coordinates defined by a bounding box ("generate a respective bounding box for the image").

- iii) **[10.c] “employ the respective bounding boxes to scale images in the rendered views of the HTML document.”**

122. Zaurus discloses reduced and magnified page views. (EX1004, 46-47.)

Pad discloses scaling text and images so that the height of the image is scaled to the height of the frame for optimizing image rendering to enable real time zooming of images. (EX1005, 34-35.) Pad further discloses zooming and scaling by using each item’s bounding box to draw the object. (EX1005, 5, 6, 16, 17.) SVG discloses transformations that can be applied to images for scalability for zoom and pan. (EX1008, 11-13, 15.) In the Zaurus-Pad-SVG combination, starting with the reduced and magnified views of Zaurus, a person of ordinary skill in the art would have incorporated the rendering techniques of Pad and SVG where each image is an object associated with a bounding box (“employ the respective bounding boxes to scale images in the rendered views of the HTML document”). In the resulting combination, each image would be retrieved via its URL, a bounding box would be computed for each image, and the bounding box would be used to scale the image during zoom.

13. Claim 11

- i) **[11.a] “The mobile hand-held device of claim 1, wherein the HTML document includes a reference to an XML document containing XML content, and wherein execution of the instructions further enables the mobile hand-held device to:”**

- ii) **[11.b] “retrieve the XML document and processing the XML content in connection with rendering the first representation of the HTML document, wherein the XML content is used to define one or more of the interpreted page layout, functionality, and design of the content associated with the HTML document.”**

123. Zaurus in view of Pad and SVG discloses and/or suggests this limitation. Zaurus discloses an HTML browser. (EX1004, 10, 40.) SVG discloses that XML-based vector graphics are designed to be implemented in browsers and coordinated with CSS and links. (EX1008, 1, 4-7, 9, 14-15.) The authors of SVG, W3C, described that XML was a standard Extensible Markup Language (XML) in 1998. (EX1046.) A person of ordinary skill in the art would have understood that supporting XML naming and tags would involve referencing external XML documents in the HTML document (“wherein the HTML document includes a reference to an XML document containing XML content”).

124. Pad discloses routine resource references in HTML and integrating fetched results into the rendered representation. (EX1005, 5 (“Retrieve a URL over the internet in the background”), 62 (“creating an html typically starts a network communication process for fetching the URL”).) SVG discloses the SVG supports CSS, DOM, and compatibility with various browsers. (EX1008, 3, 4.) Given that both Zaurus and Pad disclosed fetching resources via URL, a person of ordinary skill in the art would have understood that SVG supports retrieving XML with style rules and the DOM while rendering. In the resulting combination, rendering HTML

would have involved referencing and retrieving XML content for handling the layout, appearance, or interactions of the page (“retrieve the XML document and processing the XML content in connection with rendering the first representation of the HTML document”).

125. SVG further discloses that SVG is “an application of XML,” which means that SVG files are XML documents intended to be retrieved and rendered by a browser (“wherein the XML content is used to define one or more of the interpreted page layout, functionality, and design of the content associated with the HTML document”). (EX1008, 4.)

14. Claim 12: “The mobile hand-held device of claim 1, wherein the primary datum is an origin in a two-dimensional XY coordinate system and the vectors from the primary datum to the object datums for the HTML object bounding boxes comprise XY coordinates in the coordinate system.”

126. Zaurus describes rendering HTML on a mobile device. (EX1004, 45-47.) Pad describes APIs and view control that uses x and y coordinates; centering the view at x and y; and returning the x and y coordinates for the views. (EX1005, 17-19, 29-32.) SVG discloses a coordinate system with coordinates, and transformation matrices. (EX1008, 10, 11) In the Zaurus-Pad-SVG combination, the coordinates and the matrices mathematically presuppose an origin in a two dimension XY space (“the primary datum is an origin in a two-dimensional XY coordinate system”).

127. Pad discloses a frame as a bounding box and maps coordinates into the box for scaling images and text. (EX1005, 34-35.) SVG describes positioning objects with coordinates and using matrices to scale images and text. (EX1008, 10-12.) A person of ordinary skill in the art would have understood that Pad's bounding boxes would be represented by x and y vectors from an origin in the coordinate system as taught by SVG's XY systems with transform vectors. Given Pad's bounding boxes and SVG's transformations, a person of ordinary skill in the art would have incorporated into the Zaurus browser the known concept of representing bounding boxes as XY vectors from an origin in a 2D coordinate system. In the resulting combination, a person of ordinary skill in the art would have understood that representing each object's layout position as a vector from an origin to the object's bounding box is the conventional approach to store a coordinate for an object and apply the matrix to scale and compute the object's position on the page. ("the vectors from the primary datum to the object datums for the HTML object bounding boxes comprise XY coordinates in the coordinate system").

15. Claim 13

- i) **[13.pre] "A method performed by a mobile hand-held device including a touch-sensitive display and an HTML rendering engine, comprising:"**

128. The Zaurus-Pad-SVG combination discloses and/or suggests this limitation for the same reasons as I discussed for limitations [1.pre]-[1.d]. (Section IX.A.3.i-v.)

- ii) **[13.a] “receiving an HTML document including HTML code and cascading style sheet (CSS) code and content associated with the HTML document, the HTML code comprising a plurality of HTML elements including at least one HTML paragraph element, at least one HTML image element, and at least one HTML hyperlink element;”**

129. The Zaurus-Pad-SVG combination discloses and/or suggests this limitation for the same reasons as I discussed for limitation [1.e]. (Section IX.A.3.vi.)

- iii) **[13.b] “processing the HTML document with the HTML rendering engine to render a first representation of the HTML document having an interpreted page layout, functionality, and design of the content associated with the HTML document that is in accordance with the HTML code and CSS code, wherein rendering the first representation of the HTML document includes,”**

130. The Zaurus-Pad-SVG combination discloses and/or suggests this limitation for the same reasons as I discussed for limitation [1.f]. (Section IX.A.3.vii.)

(1) [13.b.1] “parsing the HTML document to identify the plurality of HTML elements;”

131. The Zaurus-Pad-SVG combination discloses and/or suggests this limitation for the same reasons as I discussed for limitation [1.f.1]. (Section IX.A.3.vii.1.)

(2) [13.b.2] “logically grouping content associated with HTML elements into HTML objects;”

132. The Zaurus-Pad-SVG combination discloses and/or suggests this limitation for the same reasons as I discussed for limitation [1.f.2]. (Section IX.A.3.vii.2.)

(3) [13.b.3] “generating page layout information including a bounding box for each HTML object; and”

133. The Zaurus-Pad-SVG combination discloses and/or suggests this limitation for the same reasons as I discussed for limitation [1.f.3]. (Section IX.A.3.vii.3.)

(4) [13.b.4.a] “storing information that links each HTML object with its corresponding page layout information,”

134. The Zaurus-Pad-SVG combination discloses and/or suggests this limitation for the same reasons as I discussed for limitation [1.f.4.a]. (Section IX.A.3.vii.4.)

- (5) **[13.b.4.b] “wherein the page layout information further includes information from which a page layout location of each of the bounding boxes can be determined;”**

135. The Zaurus-Pad-SVG combination discloses and/or suggests this limitation for the same reasons as I discussed for limitation [1.f.4.b]. (Section IX.A.3.vii.5.)

- iv) **[13.c] “translating the first representation of the HTML document to generate a scalable vector representation of the HTML document wherein generating the scalable vector representation includes,”**

136. The Zaurus-Pad-SVG combination discloses and/or suggests this limitation for the same reasons as I discussed for limitation [1.g]. (Section IX.A.3.viii.)

- (1) **[13.c.1] “defining a primary datum corresponding to the interpreted page layout;”**

137. The Zaurus-Pad-SVG combination discloses and/or suggests this limitation for the same reasons as I discussed for limitation [1.g.1]. (Section IX.A.3.viii.1.)

(2) [13.c.2] “for each HTML object,”

(a) [13.c.2.a] “defining an object datum corresponding to a layout location datum for the HTML object’s bounding box;”

138. The Zaurus-Pad-SVG combination discloses and/or suggests these limitations for the same reasons as I discussed for limitations [1.g.2]-[1.g.2.a].

(Section IX.A.3.viii.1-viii.1.a.)

(b) [13.c.2.b] “generating a vector from the primary datum to the object datum for the HTML object's bounding box; and”

139. The Zaurus-Pad-SVG combination discloses and/or suggests this limitation for the same reasons as I discussed for limitation [1.g.2.b]. (Section

IX.A.3.viii.1.b.)

(c) [13.c.2.c] “creating a reference that links the HTML object to the vector that is generated; and”

140. The Zaurus-Pad-SVG combination discloses and/or suggests this limitation for the same reasons as I discussed for limitation [1.g.2.c]. (Section

IX.A.3.viii.1.c.)

v) [13.d.1] “render the scalable vector representation of the HTML document on the touch-sensitive display using a first scale factor to display the HTML document at a first zoom level under which the HTML document is displayed to fit across a width of the touch-sensitive display,”

141. The Zaurus-Pad-SVG combination discloses and/or suggests this limitation for the same reasons as I discussed for limitation [1.h.1]. (Section IX.A.3.ix.)

- vi) **[13.d.2] “wherein the scalable vector representation of the HTML document is configured to enable a user to view the HTML document at one or more user-defined zoom levels by rendering the scalable vector representation on the touch-sensitive display using one or more respective scale factors in response to associated user inputs made via the touch-sensitive display, and”**
- vii) **[13.d.3] “wherein the interpreted page layout, functionality, and design of the content associated with the HTML document is preserved at each of the first zoom level and the one or more user-defined zoom levels.”**

142. The Zaurus-Pad-SVG combination discloses and/or suggests this limitation for the same reasons as I discussed for limitations [1.h.2]-[1.h.3]. (Section IX.A.3.x-xi.)

16. Claim 14: “The method of claim 13, wherein the mobile hand-held device is a mobile phone.”

143. The Zaurus-Pad-SVG combination discloses and/or suggests this claim for the same reasons as I discussed for claim 2. (Section IX.A.4.)

17. Claim 15: “The method of claim 13, wherein the mobile hand-held device is a hand-held computer.”

144. The Zaurus-Pad-SVG combination discloses and/or suggests this claim for the same reasons as I discussed for claim 3. (Section IX.A.5.)

- 18. Claim 16: “The method of claim 13, further comprising enabling the user to zoom in on a user-selectable portion of a view of the HTML document in response to a user interface input made via the touch-sensitive display.”**

145. The Zaurus-Pad-SVG combination discloses and/or suggests this claim for the same reasons as I discussed for claim 4. (Section IX.A.6.)

- 19. Claim 17: “The method of claim 13, wherein the views of the HTML document are rendered in real-time to effect zooming operations.”**

146. The Zaurus-Pad-SVG combination discloses and/or suggests this claim for the same reasons as I discussed for claim 5. (Section IX.A.7.)

- 20. Claim 18: “The method of claim 17, wherein the zooming operations are effected in a fraction of a second.”**

147. The Zaurus-Pad-SVG combination discloses and/or suggests this claim for the same reasons as I discussed for claim 6. (Section IX.A.8.)

21. Claim 19

- i) [19.a] “The method of claim 13, further comprising:”**
- ii) [19.b] “generating a vector-based display list associated with the scalable vector representation; and”**

148. The Zaurus-Pad-SVG combination discloses and/or suggests this limitation for the same reasons as I discussed for limitations [7.a]-[7.b]. (Section IX.A.9.i-ii.)

- iii) **[19.c] “employing the vector-based display list to render views of the HTML document on the touch-sensitive display at different scale factors to effect zooming operations.”**

149. The Zaurus-Pad-SVG combination discloses and/or suggests this limitation for the same reasons as I discussed for limitation [7.c]. (Section IX.A.9.iii.)

22. Claim 20: “The method of claim 13, wherein the HTML document is an HTML parent document for a Web page.”

150. The Zaurus-Pad-SVG combination discloses and/or suggests this claim for the same reasons as I discussed for claim 8. (Section IX.A.10.)

23. Claim 21

- i) **[21.a] “The mobile hand-held device of claim 20, wherein the HTML document includes references to a plurality of images, each reference comprising a respective Universal Resource Locators (URLs), and wherein execution of the instructions performs further operations comprising:”**
- ii) **[21.b] “for each of the plurality of images;**
 - (1) [21.b.1] “retrieve the image via the URL for the image; and”**

151. I have been instructed to assume that Claim 21 is a method claim that depends on method Claim 20. The Zaurus-Pad-SVG combination discloses and/or suggests this limitation for the same reasons as I discussed for limitations [10.a], [10.b], and [10.b.1]. (Section IX.A.12.i-ii.1.)

(2) [21.b.2] “generating a respective bounding box for the image; and”

152. The Zaurus-Pad-SVG combination discloses and/or suggests this limitation for the same reasons as I discussed for limitation [10.b.2]. (Section IX.A.12.ii.2.)

iii) [21.c] “employ the respective bounding boxes to scale images in the rendered views of the HTML document.”

153. The Zaurus-Pad-SVG combination discloses and/or suggests this limitation for the same reasons as I discussed for limitation [10.c]. (Section IX.A.12.iii.)

24. Claim 22: “The method of claim 13, wherein the HTML elements includes an HTML table having a plurality of cells, each cell having at least one of associated text content and associated image content, and the CSS code includes CSS code defining how the table and its associated content is to be rendered.”

154. The Zaurus-Pad-SVG combination discloses and/or suggests this claim for the same reasons as I discussed for claim 9. (Section IX.A.11.)

25. Claim 23

- i) **[23.a] “The method of claim 13, wherein the content associated with the HTML document includes a plurality of images that are referenced using respective Universal Resource Locators (URLs) in the HTML document, further comprising:”**
- ii) **[23.b] “for each of the plurality of images;”**

- (1) **[23.b.1] “retrieving the image via the URL for the image; and”**

155. The Zaurus-Pad-SVG combination discloses and/or suggests these limitations for the same reasons as I discussed for limitations [10.a], [10.b], and [10.b.1]. (Section IX.A.12.i-ii.1.)

- (2) **[23.b.2] “generating a respective bounding box for the image; and”**

156. The Zaurus-Pad-SVG combination discloses and/or suggests this limitation for the same reasons as I discussed for limitation [10.b.2]. (Section IX.A.12.ii.2.)

- iii) **[23.c] “employing the respective bounding boxes to scale images in the rendered views of the HTML document.”**

157. The Zaurus-Pad-SVG combination discloses and/or suggests this limitation for the same reasons as I discussed for limitation [10.c]. (Section IX.A.12.iii.)

26. Claim 24

- i) **[24.a] “The method of claim 13, wherein the HTML document includes a reference to an XML document containing XML content, further comprising:”**
- ii) **[24.b] “retrieving the XML document and processing the XML content in connection with rendering the first representation of the HTML document, wherein the XML content is used to define one or more of the interpreted page layout, functionality, and design of the content associated with the HTML document.”**

158. The Zaurus-Pad-SVG combination discloses and/or suggests these limitations for the same reasons as I discussed for limitations [11.a]-[11.b]. (Section IX.A.13.i-ii.)

- 27. Claim 25: “The method of claim 13, wherein the primary datum is an origin in a two-dimensional XY coordinate system and the vectors from the primary datum to the object datums for the HTML object bounding boxes comprise XY coordinates in the coordinate system.”**

159. The Zaurus-Pad-SVG combination discloses and/or suggests this limitation for the same reasons as I discussed for claim 12. (Section IX.A.14.)

28. Claim 26

- i) [26.pre] “A mobile hand-held device, comprising:**
- ii) [26.a] “a processor;”**
- iii) [26.b] “a wireless communications device, to facilitate wireless communication with a network that supports access to the Internet;”**
- iv) [26.c] “a touch-sensitive display; and”**
- v) [26.d] “flash memory, operatively coupled to the processor, in which a plurality of instructions are stored comprising a plurality of software components including an HTML rendering engine, wherein the instructions, when executed by the mobile hand-held device, enable the mobile hand-held device to,”**

160. The Zaurus-Pad-SVG combination discloses and/or suggests this limitation for the same reasons as I discussed for limitations [1.pre]-[1.d]. (Section IX.A.3.i-v.)

- vi) [26.e] “receive an HTML document comprising HTML-based content including HTML code and cascading style sheet (CSS) code and content associated with the HTML document;”**

161. The Zaurus-Pad-SVG combination discloses and/or suggests this limitation for the same reasons as I discussed for limitation [1.e]. (Section IX.A.3.vi.)

vii) [26.f] “produce scalable content by,”

- (1) [26.f.1] “processing the HTML-based content with the rendering engine to render an interpreted page layout, functionality, and design of the content associated with the HTML document in accordance with the HTML code and CSS code;”**

162. The Zaurus-Pad-SVG combination discloses and/or suggests these limitations for the same reasons as I discussed for limitations [1.f] and [1.g].
(Section IX.A.3.vii, viii.)

- (2) [26.f.2] “logically grouping selected content into HTML objects, each HTML object including associated display content;”**

163. The Zaurus-Pad-SVG combination discloses and/or suggests this limitation for the same reasons as I discussed for limitation [1.f.2]. (Section IX.A.3.vii.2.)

- (3) [26.f.3] “defining a primary datum corresponding to the interpreted page layout; and,”**

164. The Zaurus-Pad-SVG combination discloses and/or suggests this limitation for the same reasons as I discussed for limitation [1.g.1]. (Section IX.A.3.viii.1.)

- (4) [26.f.4] “for each HTML object,”**

- (a) [26.f.4.a] “defining an object datum corresponding to a layout location datum for the HTML object’s associated display content;”**

- (b) **[26.f.4.b] “generating a vector from the primary datum to the object datum for the HTML object; and”**
- (c) **[26.f.4.c] “creating a reference that links the HTML object to the vector that is generated; and”**

165. The Zaurus-Pad-SVG combination discloses and/or suggests this limitation for the same reasons as I discussed for limitations [1.g.2-g.2.c]. (Section IX.A.3.viii.2-c.)

- viii) **[26.g] “employ at least one of the scalable content or data derived therefrom to,”**
 - (1) **[26.g.1] “render a view of the HTML document on the touch-sensitive display at a first zoom level under which the HTML document is displayed to fit across a width of the touch-sensitive display; and”**

166. The Zaurus-Pad-SVG combination discloses and/or suggests this limitation for the same reasons as I discussed for limitation [1.h]. (Section IX.A.3.ix.)

- (2) **[26.g.2] “render views of the HTML document on the touch-sensitive display in response to associated user inputs to enable the HTML document to be viewed at various zoom levels and panned views”**

167. The Zaurus-Pad-SVG combination discloses and/or suggests this limitation for the same reasons as I discussed for limitations [1.h.2]. (Section IX.A.3.x.)

- (3) [26.g.3] “while preserving the interpreted page layout, functionality, and design of the content associated with the HTML document at each zoom level and panned view including the first zoom level.”

168. The Zaurus-Pad-SVG combination discloses and/or suggests this limitation for the same reasons as I discussed for limitations [1.h.3]. (Section IX.A.3.xi.)

29. **Claim 27: “The mobile hand-held device of claim 26, wherein the mobile hand-held device is a mobile phone.”**

169. The Zaurus-Pad-SVG combination discloses and/or suggests this claim for the same reasons as I discussed for claim 2. (Section IX.A.4.)

30. **Claim 28: “The mobile hand-held device of claim 26, wherein the mobile hand-held device is a hand-held computer.”**

170. The Zaurus-Pad-SVG combination discloses and/or suggests this claim for the same reasons as I discussed for claim 3. (Section IX.A.5.)

31. **Claim 29: “The mobile hand-held device of claim 26, wherein execution of the instructions further enables the user to zoom in on a user-selectable portion of a view of the HTML document in response to a user interface input made via the touch-sensitive display.”**

171. The Zaurus-Pad-SVG combination discloses and/or suggests this claim for the same reasons as I discussed for claim 4. (Section IX.A.6.)

32. **Claim 30: “The mobile hand-held device of claim 26, wherein the views of the HTML document are rendered in real-time to effect zooming operations.”**

172. The Zaurus-Pad-SVG combination discloses and/or suggests this claim for the same reasons as I discussed for claim 5. (Section IX.A.7.)

33. Claim 31: “The mobile hand-held device of claim 30, wherein the zooming operations are effected in a fraction of a second.”

173. The Zaurus-Pad-SVG combination discloses and/or suggests this claim for the same reasons as I discussed for claim 6. (Section IX.A.8.)

34. Claim 32

- i) [32.a] “The mobile hand-held device of claim 26, wherein execution of the instructions further enables the device to:”**
- ii) [32.b] “generate a vector-based display list associated with the scalable content; and”**

174. The Zaurus-Pad-SVG combination discloses and/or suggests this limitation for the same reasons as I discussed for limitations [7.a]-[7.b]. (Section IX.A.9.i-ii.)

- iii) [32.c] “employ the vector-based display list to render views of the HTML document on the touch-sensitive display at different scale factors to effect zooming operations.”**

175. The Zaurus-Pad-SVG combination discloses and/or suggests this limitation for the same reasons as I discussed for limitation [7.c]. (Section IX.A.9.iii.)

- 35. Claim 33: “The mobile hand-held device of claim 26, wherein the HTML document is an HTML parent document for a Web page.”**

176. The Zaurus-Pad-SVG combination discloses and/or suggests this claim for the same reasons as I discussed for claim 8. (Section IX.A.10.)

- 36. Claim 34: “The mobile hand-held device of claim 26, wherein the HTML-based content includes an HTML table, and the CSS code includes CSS code defining how the table is to be formatted in the interpreted page layout page of the HTML document.”**

177. The Zaurus-Pad-SVG combination discloses and/or suggests this claim for the same reasons as I discussed for claim 9. (Section IX.A.11.)

- 37. Claim 35**

- i) [35.a] “The mobile hand-held device of claim 26, wherein execution of the instructions further enables the device to:”**
- ii) [35.b] “for each HTML object, generate a bounding box for the HTML object; and”**

178. The Zaurus-Pad-SVG combination discloses and/or suggests this limitation for the same reasons as I discussed for limitation [1.f.3]. (Section IX.A.3.vii.3.)

- iii) [35.c] “employ the bounding boxes to interpret the interpreted page layout of the HTML document.”**

179. The Zaurus-Pad-SVG combination discloses and/or suggests this limitation for the same reasons as I discussed for limitation [1.f.4.b]. (Section IX.A.3.vii.5.)

B. Ground 2: Zaurus, Pad, SVG, and SVF Disclose and/or Suggest Claims 1-35

180. In my opinion, the Zaurus-Pad-SVG combination discloses and/or suggests claims 1-35. However, to the extent Patent Owner argues, or the Board finds, that a person of ordinary skill in the art would have needed more information about certain features directed to the “datum” that are recited in limitations [1.g] and [12] (and similar limitations in limitations [13.c], [25], and [26.f]), SVF discloses and/or suggests those features.

1. Motivation to Combine

181. A person of ordinary skill in the art would have been motivated to incorporate SVF into the Zaurus-Pad-SVG combination to optimize the zooming capabilities of the combination. SVF was also based on basic W3C software standards that a person of ordinary skill in the art would have been motivated to combine with Zaurus-Pad-SVG. (*See e.g.*, EX1001, 4:67-5:3; EX1008; EX1035; EX1037; EX1044; EX1046.) SVF offers an efficient architecture utilizing variable-length integer command parameters (1 to 8 bytes) to create compact files (EX1044, 1), which would enable efficient usage of the device memory in the combination. SVF includes a “Notifications” feature designed to trigger actions, such as turning layers on or off, when a specific “magnification” or zoom level is reached. (EX1044, 6-7.) In the combination, SVF would make the zooming features more efficient and responsive for the user and improve the user experience.

182. While the Zaurus-Pad-SVG combination defines reference points and an origin for layout and zooming, a person of ordinary skill in the art would have been motivated to incorporate SVF's "coordinate system" that defines that the "origin (0,0) is in the lower left corner with increasing x coordinates moving to the right and increasing y coordinates moving up." (EX1044, 1-2.) A person of ordinary skill in the art would have been motivated to incorporate this integer-based coordinate system to conduct the vector calculations for rendering the objects. Furthermore, while the Zaurus-Pad-SVG combination mentions utilizing grid arrangements, a person of ordinary skill in the art would have been motivated to utilize SVF's support for "Scale and base point" data so that when "original coordinate system needed to be recreated" from the stored values, SVF would enable mapping the layout of an HTML Web page onto the integer-based vector grid.

2. Reasonable Expectation of Success

183. A person of ordinary skill in the art would have had a reasonable expectation of success in implementing SVF in the Zaurus-Pad-SVG combination. As an initial matter, the '154 patent acknowledges that "SVF has been under consideration by the W3C (World Wide Web Consortium) for adoption as a standard for vector content on the World Wide Web." (EX1001, 4:67-5:3.) Zaurus, Pad, and SVG were also based on W3C standards that a person of ordinary skill in the art would have understood to be interoperable. (*See e.g.*, EX1008; EX1035; EX1037;

EX1044; EX1046.) SVF includes standard C++ code, such as “svfout.c” to handle the output of commands, which a person of ordinary skill in the art would have integrated into SVF’s rendering logic and Pad’s “C++ core” of Pad. (EX1007, 140-141; EX1044, 2.) The technical architecture of SVF is compatible with the Zaurus’s “32-bit RISC” processor because SVF utilizes “integer” command parameters and coordinates. (EX1004, 18; EX1044, 1.)

3. Claims 1-35

i) [1.g.1] / [13.c.1] / [26.f.3]⁴

184. SVF defines a global coordinate system where the “origin (0,0) is in the lower left corner” and objects in the SVF file are drawn using integer coordinates relative to this defined coordinate system, which corresponds to the “primary datum” for the “page layout.” (EX1044, 1.) SVF defines the “image’s size (or extents) so the program viewing it will know how to display it initially” (“interpreted page layout”). (EX1044, 2.) SVF’s coordinate system is used to create a scalable representation because the “size of the image can change,” and the format supports “Scale and base point” parameters. (EX1044, 2, 7.) SVF’s “Notifications” trigger

⁴ The claim language is provided above in Ground 1 and is not repeated in Ground 2.

actions based on “magnification” (EX1044, 6), further demonstrating that SVF supports operations for scalable viewing.

ii) [1.g.2.a] / [13.c.2.a] / [26.f.4.a]

185. SVF commands use specific coordinates to position each object. SVF supports HTML-equivalent objects: “text,” “basic drawing objects” like “circles” and “arcs,” and “hyperlinks.” (EX1044, 1.) SVF also discloses “embedding SVF images in HTML documents” (EX1044, 7.) SVF objects can be “Filled regions” or specific shapes like “rectangles.” (EX1044, 1, 4.) These shapes define the boundaries (“bounding box”) of an image or graphic. The “MoveTo” command “moves the current point to a new location.” (EX1044, 2.) This “current point” is the reference point (“object datum”) for subsequent drawing commands. For example, “Polylines start drawing from the current point.” (EX1044, 2.) The coordinates specified in these commands correspond to the “layout location datum” derived from the original layout. (EX1044, 1.)

iii) [1.g.2.b] / [13.c.2.b] / [26.f.4.b]

186. SVF is defined as a “Simple Vector Format” where “objects are drawn in the order they occur in the file,” which would involve handling a list of vectors. (EX1044, 1.) Specifying a coordinate pair (X, Y) for a “MoveTo” command defines a vector from the origin (0,0) (“primary datum”) to that point (X, Y) (“object datum”). (EX1044, 1, 2.) Using integers for coordinates allows the commands to

be mathematically scaled vectors. SVF also discloses “Relative lines and polylines,” where “each point describes an offset from the current point,” which is a generation of vectors between datums. (EX1044, 3.)

iv) [1.g.2.c] / [13.c.2.c] / [26.f.4.c]

187. SVF discloses a hyperlink that defines a region (“vector”) where the user can click to “perform an action” (“HTML object”), which is a “reference that links the HTML object to the vector.” (EX1044, 1, 5.) SVF allows defining a “2 dimensional” link region, which is “chosen by picking inside the region.” (EX1044, 5.) Furthermore, SVF discloses grouping “objects” into “Layers” for controlling visibility, which creates a link between the “objects” and their “vectors.” (EX1044, 1, 6.)

v) [12] / [25]

188. SVF defines its coordinate system as a two-dimensional XY system where “the origin (0,0) is in the lower left corner with increasing x coordinates moving to the right and increasing y coordinates moving up” and coordinates are specified as “positive integers.” (EX1044, 1.) In the combination, the coordinates are part of the “vector” and the origin (“primary datum”) as required by the claim. The coordinates would enable the calculation of scale factors for zooming operations, such as those configured by the “Notifications” logic. (EX1044, 6.)

Declaration of Dr. Benjamin Goldberg

U.S. Patent No. 10,083,154

X. CONCLUSION

189. 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. 10,083,154	U.S. Patent No. 7,461,353	U.S. Patent No. 7,831,926	Comparison
[1.pre] A mobile hand-held device, comprising: [1.a] a processor; [1.b] a wireless communications device, to facilitate wireless communication with a network that supports access to the Internet; [1.c] a touch-sensitive display; and [1.d] flash memory, operatively coupled to the processor, in which a plurality of instructions are stored comprising a plurality of software components including an HTML rendering engine, wherein the instructions are configured to be executed by the	[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 plurality of instructions are stored that when executed by the processor enable the mobile hand-held device to perform operations including, [52] The mobile hand-held device of claim 51, wherein the corresponding user input comprises tapping on the image via the display. [317.pre] A hand-held wireless device, comprising: [317.a] a processor,	Appendix C (IPR2013-00004) [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	These limitations correspond to a mobile hand-held device having a processor, wireless communications, a touch-sensitive display, and non-volatile storage (including flash memory) storing executable browser/HTML-rendering instructions. The HTML rendering engine corresponds to the disclosed browser software and associated rendering engine functionality, and the touch-sensitive display corresponds to tapping on the display via the cited tap input limitations.

U.S. Patent No. 10,083,154	U.S. Patent No. 7,461,353	U.S. Patent No. 7,831,926	Comparison
processor to enable the mobile hand-held device to, [13.pre] A method performed by a mobile hand-held device including a touch-sensitive display and an HTML rendering engine, comprising: [26.pre] A mobile hand-held device, comprising: [26.a] a processor; [26.b] a wireless communications device, to facilitate wireless communication with a network that supports access to the Internet; [26.c] a touch-sensitive display; and [26.d] flash memory, operatively coupled to	[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,	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;	

Declaration of Dr. Benjamin Goldberg
U.S. Patent No. 10,083,154

U.S. Patent No. 10,083,154	U.S. Patent No. 7,461,353	U.S. Patent No. 7,831,926	Comparison
the processor, in which a plurality of instructions are stored comprising a plurality of software components including an HTML rendering engine, wherein the instructions, when executed by the mobile hand-held device, enable the mobile hand-held device to,		[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, Appendix E (95/000,635) [79.pre] A mobile device, comprising: [79.a] processing means; [79.b] wireless communications means,	

Declaration of Dr. Benjamin Goldberg
U.S. Patent No. 10,083,154

U.S. Patent No. 10,083,154	U.S. Patent No. 7,461,353	U.S. Patent No. 7,831,926	Comparison
		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	

U.S. Patent No. 10,083,154	U.S. Patent No. 7,461,353	U.S. Patent No. 7,831,926	Comparison
		enabled to perform operations, including,	
[1.e] receive an HTML document including HTML code and cascading style sheet (CSS) code and content associated with the HTML document, the HTML code comprising a plurality of HTML elements including at least one HTML paragraph element, at least one HTML image element, and at least one HTML hyperlink element; [13.a] receiving an HTML document including HTML code and cascading style sheet (CSS) code and content associated with the HTML document, the	[36.c] retrieving the Web page via the wireless communications device, and processing HTML-based Web content to produce scalable content; and [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. [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	Appendix C (IPR2013-00004) [30.g] retrieving HTML-based content associated with the Web page; [30.j] wherein preservation of the functionality defined by the HTML-based content includes preservation of hyperlink functionality. [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	Receiving the HTML document with HTML and CSS elements corresponds to retrieving the Web page and associated style information (including CSS). The cited unpatentable limitations address an HTML image element, an HTML paragraph element, an HTML hyperlink element via preserved hyperlink functionality, and CSS content/data defining layout and presentation attributes.

U.S. Patent No. 10,083,154	U.S. Patent No. 7,461,353	U.S. Patent No. 7,831,926	Comparison
HTML code comprising a plurality of HTML elements including at least one HTML paragraph element, at least one HTML image element, and at least one HTML hyperlink element; [26.e] receive an HTML document comprising HTML-based content including HTML code and cascading style sheet (CSS) code and content associated with the HTML document;	reference and having a width and height;	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) [1.g] retrieving HTML-based Web content associated with the Web page; [1.i.2.b] wherein preservation of the functionality defined by the HTML-based	

Declaration of Dr. Benjamin Goldberg
U.S. Patent No. 10,083,154

U.S. Patent No. 10,083,154	U.S. Patent No. 7,461,353	U.S. Patent No. 7,831,926	Comparison
		content includes preservation of hyperlink functionality. [13] The mobile device of claim 1, wherein the Web content includes at least one image, and wherein execution of the instructions performs further operations comprising enabling a user to view an image at a higher resolution than a current resolution by tapping on the image via the touch-sensitive display, wherein in response thereto, the display is re-rendered such that the image is displayed to fit across a width of a display area of the touch-sensitive display. [14] The mobile device	

Declaration of Dr. Benjamin Goldberg
U.S. Patent No. 10,083,154

U.S. Patent No. 10,083,154	U.S. Patent No. 7,461,353	U.S. Patent No. 7,831,926	Comparison
		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. Appendix E (95/000,635)	

Declaration of Dr. Benjamin Goldberg
U.S. Patent No. 10,083,154

U.S. Patent No. 10,083,154	U.S. Patent No. 7,461,353	U.S. Patent No. 7,831,926	Comparison
		[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. [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	

Declaration of Dr. Benjamin Goldberg
U.S. Patent No. 10,083,154

U.S. Patent No. 10,083,154	U.S. Patent No. 7,461,353	U.S. Patent No. 7,831,926	Comparison
		re-rendered such that the image is displayed to fit a width of a display area of the touch-sensitive display. [66] The mobile device of claim 52, wherein execution of the instructions performs further operations comprising enabling a user to view a paragraph of the Web content at a higher resolution than a current resolution by tapping on the paragraph via the touch-sensitive display, wherein in response thereto, the display is re-rendered such that content corresponding to the selected paragraph is displayed across at least one of a width and height of a	

U.S. Patent No. 10,083,154	U.S. Patent No. 7,461,353	U.S. Patent No. 7,831,926	Comparison
		display area of the touch-sensitive display. [79.i.2.a] wherein preservation of the functionality defined by the HTML-based Web content includes preservation of hyperlink functionality.	
[1.f] process the HTML document with the HTML rendering engine to render a first representation of the HTML document having an interpreted page layout, functionality, and design of the content associated with the HTML document that is in accordance with the HTML code and CSS code, wherein rendering the first representation of	[36.c] retrieving the Web page via the wireless communications device, and processing HTML-based Web content to produce scalable content; and [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) [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 while preserving	Both this limitation and the unpatentable limitations recite processing HTML into a representation. Processing the HTML document with a rendering engine to produce an interpreted page layout and design corresponds to the cited “processing” and “rendering engine interpretation” limitations, which

U.S. Patent No. 10,083,154	U.S. Patent No. 7,461,353	U.S. Patent No. 7,831,926	Comparison
the HTML document includes, [13.b] processing the HTML document with the HTML rendering engine to render a first representation of the HTML document having an interpreted page layout, functionality, and design of the content associated with the HTML document that is in accordance with the HTML code and CSS code, wherein rendering the first representation of the HTML document includes, [26.f] produce scalable content by, [26.f.1] processing the HTML-based content with the rendering engine to render an		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) [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.h] retrieving and processing the HTML-based Web content to	generate and preserve the interpreted layout, functionality, and design used for subsequent scalable rendering.

Declaration of Dr. Benjamin Goldberg
U.S. Patent No. 10,083,154

U.S. Patent No. 10,083,154	U.S. Patent No. 7,461,353	U.S. Patent No. 7,831,926	Comparison
interpreted page layout, functionality, and design of the content associated with the HTML document in accordance with the HTML code and CSS code;		produce scalable content; and [78] The mobile device of claim 52, wherein a portion of the HTML-based Web content comprises cascading style sheet data. [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. 10,083,154	U.S. Patent No. 7,461,353	U.S. Patent No. 7,831,926	Comparison
[1.f.1] parsing the HTML document to identify the plurality of HTML elements; [13.b.1] parsing the HTML document to identify the plurality of HTML elements;	[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; [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;	Appendix C (IPR2013-00004) [72.b] parsing HTML-based code corresponding to the received Web content to identify content on the Web page; 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; Appendix E (95/000,635) [79.h] retrieving and processing the HTML-based Web content to	Both this limitation and the unpatentable limitations recite parsing HTML for its elements and layout. Parsing the HTML document to identify HTML elements corresponds to the parsing HTML code to determine the layout.

U.S. Patent No. 10,083,154	U.S. Patent No. 7,461,353	U.S. Patent No. 7,831,926	Comparison
		produce scalable content; and	
[1.f.2] logically grouping content associated with HTML elements into HTML objects; [13.b.2] logically grouping content associated with HTML elements into HTML objects; [26.f.2] logically grouping selected content into HTML objects, each HTML object including associated display content;	[58.c] logically grouping selected content into objects; [138.c] logically grouping selected content into objects;	Appendix C (IPR2013-00004) [72.c] logically grouping selected content into objects; Appendix D (90/009,995) [18.b] parsing the HTML-based content to logically group content into objects; Appendix E (95/000,635) [79.h] retrieving and processing the HTML-based Web content to produce scalable content; and	Grouping the HTML elements into objects corresponds to grouping content into objects.
[1.f.3] generating page layout information	[59] The mobile hand-held device of claim 58, wherein	Appendix C (IPR2013-00004)	Generating page layout information and

U.S. Patent No. 10,083,154	U.S. Patent No. 7,461,353	U.S. Patent No. 7,831,926	Comparison
including a bounding box for each HTML object; and [13.b.3] generating page layout information including a bounding box for each HTML object; and [35.a] The mobile hand-held device of claim 26, wherein execution of the instructions further enables the device to: [35.b] for each HTML object, generate a bounding box for the HTML object; and	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. [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.	[72.e] for each object, 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 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	bounding boxes for HTML objects and objects is equivalent.

U.S. Patent No. 10,083,154	U.S. Patent No. 7,461,353	U.S. Patent No. 7,831,926	Comparison
		by the object's associated group of content.	
[1.f.4.a] storing information that links each HTML object with its corresponding page layout information, [13.b.4.a] storing information that links each HTML object with its corresponding page layout information,	[58.e.3] creating a reference that links the object to the vector that is generated. [138.e.3] creating a reference that links the object to the vector that is generated.	Appendix C (IPR2013-00004) [72.e.3] creating a reference that links the object to its corresponding vector. Appendix D (90/009,995) [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	The cited object/vector references store per-object associations to positional/layout data (i.e., the vector/location information forming part of the page layout information), consistent with linking each object to corresponding page layout information used for scalable rendering. The cited object to vector references further show stored associations between objects and corresponding position data used for scalable rendering, consistent with maintaining per object linkage to layout information.

Declaration of Dr. Benjamin Goldberg
U.S. Patent No. 10,083,154

U.S. Patent No. 10,083,154	U.S. Patent No. 7,461,353	U.S. Patent No. 7,831,926	Comparison
		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,	

U.S. Patent No. 10,083,154	U.S. Patent No. 7,461,353	U.S. Patent No. 7,831,926	Comparison
		retrieving content associated with that object bounding box; and [70.e] applying an appropriate scale factor to the content to render the display.	
[1.f.4.b] wherein the page layout information further includes information from which a page layout location of each of the bounding boxes can be determined; [13.b.4.b] wherein the page layout information further includes information from which a page layout location of each of the bounding boxes can be determined; [35.c] employ the bounding boxes to	[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	Appendix C (IPR2013- 00004) [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	Both this limitation and the unpatentable limitations describe assigning layout location information to each object and the bounding box location corresponds to the object datum. The layout information including bounding box locations corresponds to the layout location including the defined object datum.

Declaration of Dr. Benjamin Goldberg
U.S. Patent No. 10,083,154

U.S. Patent No. 10,083,154	U.S. Patent No. 7,461,353	U.S. Patent No. 7,831,926	Comparison
interpret the interpreted page layout of the HTML document.		Appendix D (90/009,995) [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. 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	

Declaration of Dr. Benjamin Goldberg
U.S. Patent No. 10,083,154

U.S. Patent No. 10,083,154	U.S. Patent No. 7,461,353	U.S. Patent No. 7,831,926	Comparison
		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	

U.S. Patent No. 10,083,154	U.S. Patent No. 7,461,353	U.S. Patent No. 7,831,926	Comparison
		to the content to render the display.	
[1.g] translate the first representation of the HTML document to generate a scalable vector representation of the HTML document wherein generating the scalable vector representation includes, [13.c] translating the first representation of the HTML document to generate a scalable vector representation of the HTML document wherein generating the scalable vector representation includes,	[43] The mobile hand-held device of claim 36, wherein at least a portion of the scalable content comprises scalable vector-based content. [33] The wireless device of claim 1, wherein a portion of the scalable content comprises 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	Both this limitation and the unpatentable limitations translate HTML into scalable content that can be vector-based. Translating the HTML document into a scalable vector representation corresponds to translating HTML content into scalable content, including scalable vector-based content.

Declaration of Dr. Benjamin Goldberg
U.S. Patent No. 10,083,154

U.S. Patent No. 10,083,154	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) [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	

Declaration of Dr. Benjamin Goldberg
U.S. Patent No. 10,083,154

U.S. Patent No. 10,083,154	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. [79.h] retrieving and processing the HTML-based Web content to produce scalable content; and	
[1.g.1] defining a primary datum corresponding to the interpreted page layout; [13.c.1] defining a primary datum corresponding to the interpreted page layout; [26.f.3] defining a primary datum	[58.d] defining a primary datum corresponding to the original page layout; and, [138.d] defining a primary datum corresponding to the original page layout; and,	Appendix C (IPR2013-00004) [72.d] defining a primary datum corresponding to the original page layout; and, Appendix D (90/009,995)	The interpreted page layout corresponds to original page layout.

Declaration of Dr. Benjamin Goldberg
U.S. Patent No. 10,083,154

U.S. Patent No. 10,083,154	U.S. Patent No. 7,461,353	U.S. Patent No. 7,831,926	Comparison
corresponding to the interpreted page layout; and,		[16.c] defining a primary datum corresponding to a page layout; and, 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	

Declaration of Dr. Benjamin Goldberg
U.S. Patent No. 10,083,154

U.S. Patent No. 10,083,154	U.S. Patent No. 7,461,353	U.S. Patent No. 7,831,926	Comparison
		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.	
[1.g.2] for each HTML object, [13.c.2] for each HTML object,	[58.e] for each object, [138.e] for each object,	Appendix C (IPR2013-00004) [72.e] for each object,	The HTML object corresponds to the object.

Declaration of Dr. Benjamin Goldberg
U.S. Patent No. 10,083,154

U.S. Patent No. 10,083,154	U.S. Patent No. 7,461,353	U.S. Patent No. 7,831,926	Comparison
[26.f.4] for each HTML object,		Appendix D (90/009,995) [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. 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	

Declaration of Dr. Benjamin Goldberg
U.S. Patent No. 10,083,154

U.S. Patent No. 10,083,154	U.S. Patent No. 7,461,353	U.S. Patent No. 7,831,926	Comparison
		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	

U.S. Patent No. 10,083,154	U.S. Patent No. 7,461,353	U.S. Patent No. 7,831,926	Comparison
		object bounding box; and [70.e] applying an appropriate scale factor to the content to render the display.	
[1.g.2.a] defining an object datum corresponding to a layout location datum for the HTML object's bounding box; [13.c.2.a] defining an object datum corresponding to a layout location datum for the HTML object's bounding box; [26.f.4.a] defining an object datum corresponding to a layout location datum for the HTML object's	[58.e.1] defining an object datum corresponding to a layout location datum for the object's associated display content; [138.e.1] defining an object datum corresponding to a layout location datum for the object's associated display content;	Appendix C (IPR2013-00004) [72.e.1] defining an object datum corresponding to a layout location datum for the object's associated display content; Appendix D (90/009,995) [16.d] for each display object, defining an object datum corresponding to a layout location datum for the object's	The HTML object's bounding box corresponds to the object's associated display content because the bounding box represents content.

Declaration of Dr. Benjamin Goldberg
U.S. Patent No. 10,083,154

U.S. Patent No. 10,083,154	U.S. Patent No. 7,461,353	U.S. Patent No. 7,831,926	Comparison
associated display content;		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. 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	

Declaration of Dr. Benjamin Goldberg
U.S. Patent No. 10,083,154

U.S. Patent No. 10,083,154	U.S. Patent No. 7,461,353	U.S. Patent No. 7,831,926	Comparison
		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.	

U.S. Patent No. 10,083,154	U.S. Patent No. 7,461,353	U.S. Patent No. 7,831,926	Comparison
[1.g.2.b] generating a vector from the primary datum to the object datum for the HTML object's bounding box; and [13.c.2.b] generating a vector from the primary datum to the object datum for the HTML object's bounding box; and [26.f.4.b] generating a vector from the primary datum to the object datum for the HTML object; and	[58.e.2] generating a vector from the primary datum to the object datum for the object; and [138.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) [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.	Generating the vector for the HTML object's bounding box is patentably indistinct from generating the vector for the object.

Declaration of Dr. Benjamin Goldberg
U.S. Patent No. 10,083,154

U.S. Patent No. 10,083,154	U.S. Patent No. 7,461,353	U.S. Patent No. 7,831,926	Comparison
		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;	

Declaration of Dr. Benjamin Goldberg
U.S. Patent No. 10,083,154

U.S. Patent No. 10,083,154	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.	
[1.g.2.c] creating a reference that links the HTML object to the vector that is generated; and [13.c.2.c] creating a reference that links the HTML object to the	[58.e.3] creating a reference that links the object to the vector that is generated. [138.e.3] creating a reference that links the object to the vector that is generated.	Appendix C (IPR2013-00004) [72.e.3] creating a reference that links the object to its corresponding vector.	The HTML object corresponds to the object.

Declaration of Dr. Benjamin Goldberg
U.S. Patent No. 10,083,154

U.S. Patent No. 10,083,154	U.S. Patent No. 7,461,353	U.S. Patent No. 7,831,926	Comparison
vector that is generated; and [26.f.4.c] creating a reference that links the HTML object to the vector that is generated; and		Appendix D (90/009,995) [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. 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	

Declaration of Dr. Benjamin Goldberg
U.S. Patent No. 10,083,154

U.S. Patent No. 10,083,154	U.S. Patent No. 7,461,353	U.S. Patent No. 7,831,926	Comparison
		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	

U.S. Patent No. 10,083,154	U.S. Patent No. 7,461,353	U.S. Patent No. 7,831,926	Comparison
		object bounding box; and [70.e] applying an appropriate scale factor to the content to render the display.	
[1.h.1] render the scalable vector representation of the HTML document on the touch-sensitive display using a first scale factor to display the HTML document at a first zoom level under which the HTML document is displayed to fit across a width of the touch-sensitive display, [13.d.1] render the scalable vector representation of the HTML document on the touch-sensitive display	[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.	Appendix C (IPR2013-00004) [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)	Both this limitation and the unpatentable limitations require scaling content so that the page width fits the display. Rendering scalable content with a scale factor to fit across the display corresponds to scaling the scalable content such that a width is rendered to fit across the display.

Declaration of Dr. Benjamin Goldberg
U.S. Patent No. 10,083,154

U.S. Patent No. 10,083,154	U.S. Patent No. 7,461,353	U.S. Patent No. 7,831,926	Comparison
using a first scale factor to display the HTML document at a first zoom level under which the HTML document is displayed to fit across a width of the touch-sensitive display, [26.g] employ at least one of the scalable content or data derived therefrom to, [26.g.1] render a view of the HTML document on the touch-sensitive display at a first zoom level under which the HTML document is displayed to fit across a width of the touch-sensitive display; 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.	

Declaration of Dr. Benjamin Goldberg
U.S. Patent No. 10,083,154

U.S. Patent No. 10,083,154	U.S. Patent No. 7,461,353	U.S. Patent No. 7,831,926	Comparison
		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 display area of the touch-sensitive display.	

U.S. Patent No. 10,083,154	U.S. Patent No. 7,461,353	U.S. Patent No. 7,831,926	Comparison
[1.h.2] wherein the scalable vector representation of the HTML document is configured to enable a user to view the HTML document at one or more user-defined zoom levels by rendering the scalable vector representation on the touch-sensitive display using one or more respective scale factors in response to associated user inputs made via the touch-sensitive display, and [13.d.2] wherein the scalable vector representation of the HTML document is configured to enable a user to view the HTML document at one or more user-defined zoom levels	[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) [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	Iterative zooming constitutes viewing at “user-defined zoom levels,” and re-rendering at “different scale factors” corresponds to the “one or more respective scale factors.”

Declaration of Dr. Benjamin Goldberg
U.S. Patent No. 10,083,154

U.S. Patent No. 10,083,154	U.S. Patent No. 7,461,353	U.S. Patent No. 7,831,926	Comparison
by rendering the scalable vector representation on the touch-sensitive display using one or more respective scale factors in response to associated user inputs made via the touch-sensitive display, and [26.g.2] render views of the HTML document on the touch-sensitive display in response to associated user inputs to enable the HTML document to be viewed at various zoom levels and panned views		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 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. 10,083,154	U.S. Patent No. 7,461,353	U.S. Patent No. 7,831,926	Comparison
[1.h.3] wherein the interpreted page layout, functionality, and design of the content associated with the HTML document is preserved at each of the first zoom level and the one or more user-defined zoom levels. [13.d.3] wherein the interpreted page layout, functionality, and design of the content associated with the HTML document is preserved at each of the first zoom level and the one or more user-defined zoom levels. [26.g.3] while preserving the interpreted page layout, functionality, and design of the content associated with 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. [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) [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.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	Near verbatim.

Declaration of Dr. Benjamin Goldberg
U.S. Patent No. 10,083,154

U.S. Patent No. 10,083,154	U.S. Patent No. 7,461,353	U.S. Patent No. 7,831,926	Comparison
HTML document at each zoom level and panned view including the first zoom level.		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 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,	

Declaration of Dr. Benjamin Goldberg
U.S. Patent No. 10,083,154

U.S. Patent No. 10,083,154	U.S. Patent No. 7,461,353	U.S. Patent No. 7,831,926	Comparison
[2] The mobile hand-held device of claim 1, wherein the mobile hand-held device is a mobile phone. [14] The method of claim 13, wherein the mobile hand-held device is a mobile phone. [27] The mobile hand-held device of claim 26, wherein the mobile hand-held device is a mobile phone.	[36.pre] A mobile hand-held device, comprising:	Appendix C (IPR2013-00004) [30.pre] A mobile phone, comprising: Appendix D (90/009,995) [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.	This limitation corresponds to the mobile phone embodiments recited in the unpatentable claims.
[3] The mobile hand-held device of claim 1, wherein the mobile hand-held device is a hand-held computer.	[317.pre] A hand-held wireless device, comprising:	Appendix C (IPR2013-00004) [52.pre] A mobile device, comprising: Appendix D (90/009,995)	Hand-held computer corresponds to mobile hand-held device, hand-held device, and hand-held wireless device.

U.S. Patent No. 10,083,154	U.S. Patent No. 7,461,353	U.S. Patent No. 7,831,926	Comparison
[15] The method of claim 13, wherein the mobile hand-held device is a hand-held computer. [28] The mobile hand-held device of claim 26, wherein the mobile hand-held device is a hand-held computer.		[3] The mobile device of claim 1, wherein the device comprises one of a hand-held device or a palm-held device. Appendix E (95/000,635) [54] The mobile device of claim 52, wherein the device comprises one of a Personal Digital Assistant (PDA) or pocket PC.	
[4] The mobile hand-held device of claim 1, wherein execution of the instructions further enables the user to zoom in on a user-selectable portion of a view of the HTML document in response to a user interface input made via	[317.h] enabling the user to, zoom and pan a view of the Web page;	Appendix C (IPR2013-00004) [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	The unpatentable claims' zoom-on-image/zoom-on-column behavior is an example of zooming on a user-selectable portion of the displayed Web page content (i.e., a selected region/object). Zooming on a user-selectable portion

U.S. Patent No. 10,083,154	U.S. Patent No. 7,461,353	U.S. Patent No. 7,831,926	Comparison
the touch-sensitive display. [16] The method of claim 13, further comprising enabling the user to zoom in on a user-selectable portion of a view of the HTML document in response to a user interface input made via the touch-sensitive display. [29] The mobile hand-held device of claim 26, wherein execution of the instructions further enables the user to zoom in on a user-selectable portion of a view of the HTML document in response to a user interface input made via the touch-sensitive display.		page in response to a user interface input made via the touch-sensitive display. 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. Appendix E (95/000,635) [82] The mobile device of claim 79, wherein execution of the	corresponds to zooming in on an image or column by tapping as well as rendering at various zoom levels and panned views.

U.S. Patent No. 10,083,154	U.S. Patent No. 7,461,353	U.S. Patent No. 7,831,926	Comparison
		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 hand-held device of claim 1, wherein the views of the HTML document are rendered in real-time to effect zooming operations. [17] The method of claim 13, wherein the views of the HTML document are rendered in real-time to effect zooming 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) [5] The mobile device of claim 1, wherein the display of the Web page is re-rendered in real-	This limitation corresponds to the unpatentable claims reciting that the display is “re-rendered in real-time to effect zooming operations,” and to “rapid zooming” re-rendering via different scale factors.

U.S. Patent No. 10,083,154	U.S. Patent No. 7,461,353	U.S. Patent No. 7,831,926	Comparison
[30] The mobile hand-held device of claim 26, wherein the views of the HTML document are rendered in real-time to effect zooming operations.		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.	
[6] The mobile hand-held device of claim 5, wherein the zooming operations are effected in a fraction of a second. [18] The method of claim 17, wherein the zooming operations are effected in a fraction of a second. [31] The mobile hand-held device of claim 30, wherein the zooming	[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	Re-rendering the display at various zoom levels in response to user inputs is consistent with rapid/real-time zooming described in the unpatentable claims, and the “fraction of a second” description is supported by the patent’s disclosure of efficient rendering without perceptible wait. (See EX1001, 5:20-25 “rendering engine

U.S. Patent No. 10,083,154	U.S. Patent No. 7,461,353	U.S. Patent No. 7,831,926	Comparison
operations are effected in a fraction of a second.		is re-rendered in real-time to effect zooming operations. Appendix E (95/000,635) [84] The mobile device of claim 79, wherein the display of the Web page is re-rendered in real-time to effect zooming operations.	within the client is so efficient that file manipulation happens in a fraction of a second. There is no perceptible wait for the user as the file is resized, or the window is repositioned. Content created for one display resolution now can be sized, real time, for any other display without degradation.”.)
[7.a] The mobile hand-held device of claim 1, wherein execution of the instructions performs further enables the device to: [7.b] generate a vector-based display list associated with the scalable vector representation; and	[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) [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	Both this limitation and the unpatentable limitations generate vector-based structures for objects, and the display list is a structured form of vector-based scalable content. The display list is an implementation form of the vector-object references. Using

U.S. Patent No. 10,083,154	U.S. Patent No. 7,461,353	U.S. Patent No. 7,831,926	Comparison
[7.c] employ the vector-based display list to render views of the HTML document on the touch-sensitive display at different scale factors to effect zooming operations. [19.a] The method of claim 13, further comprising: [19.b] generating a vector-based display list associated with the scalable vector representation; and [19.c] employing the vector-based display list to render views of the HTML document on the touch-sensitive display at different scale factors to effect zooming operations.		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 portion employed for rendering purposes,	the vector list to re-render at different zoom levels is equivalent creating and storing object-vector references for rendering at various zoom levels.

U.S. Patent No. 10,083,154	U.S. Patent No. 7,461,353	U.S. Patent No. 7,831,926	Comparison
[32.a] The mobile handheld device of claim 26, wherein execution of the instructions further enables the device to: [32.b] generate a vector-based display list associated with the scalable content; and [32.c] employ the vector-based display list to render views of the HTML document on the touch-sensitive display at different scale factors to effect zooming operations.		[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.	
[8] The mobile handheld device of claim 1, wherein the HTML document is an HTML parent document for a Web page.	[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,	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	Both this limitation and the unpatentable limitations recite that the Web page includes HTML. The HTML parent document is equivalent to the

U.S. Patent No. 10,083,154	U.S. Patent No. 7,461,353	U.S. Patent No. 7,831,926	Comparison
[20] The method of claim 13, wherein the HTML document is an HTML parent document for a Web page. [33] The mobile hand-held device of claim 26, wherein the HTML document is an HTML parent document for a Web page.	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	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; 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	original Web page that includes content because HTML parent documents refer to or include other objects. (See EX1001, 8:31-34 (“the HTML content of the parent HTML document is parsed to search for references to external objects such as HTML frames and graphics.”).)

Declaration of Dr. Benjamin Goldberg
U.S. Patent No. 10,083,154

U.S. Patent No. 10,083,154	U.S. Patent No. 7,461,353	U.S. Patent No. 7,831,926	Comparison
		associated with the Web page; 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 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	
[9] The mobile hand-held device of claim 1, wherein the HTML elements includes an	[66] The mobile hand-held device of claim 36, wherein a portion of the HTML-based Web content comprises	Appendix C (IPR2013-00004)	The unpatentable claims require preserving page layout/design via interpreted HTML/CSS

U.S. Patent No. 10,083,154	U.S. Patent No. 7,461,353	U.S. Patent No. 7,831,926	Comparison
HTML table having a plurality of cells, each cell having at least one of associated text content and associated image content, and the CSS code includes CSS code defining how the table and its associated content is to be rendered. [22] The method of claim 13, wherein the HTML elements includes an HTML table having a plurality of cells, each cell having at least one of associated text content and associated image content, and the CSS code includes CSS code defining how the table and its associated content is to be rendered. [34] The mobile hand-held device of claim 26,	cascaded style sheet data defining aspects of the Web page design that are preserved at each zoom level and panned view.	[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;	and cascaded style sheet data. To the extent the HTML-based content uses an HTML table for layout (a common structure for multi-column layouts), CSS defining table formatting and rendering is consistent with the unpatentable claims' CSS/layout-preservation teachings.

U.S. Patent No. 10,083,154	U.S. Patent No. 7,461,353	U.S. Patent No. 7,831,926	Comparison
wherein the HTML-based content includes an HTML table, and the CSS code includes CSS code defining how the table is to be formatted in the interpreted page layout page of the HTML document.		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.	
[10.a] The mobile hand-held device of claim 1, wherein the content associated with the HTML document includes a plurality of images that are referenced using respective Universal Resource Locators (URLs) in the HTML document, and wherein execution of the	[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.	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 higher resolution than a current	A plurality of images is patentably indistinct from at least one image. Because HTML images are commonly referenced by URL attributes, retrieving HTML-based Web content that includes an image entails retrieving (or enabling retrieval of) the corresponding image resource at the referenced URL.

U.S. Patent No. 10,083,154	U.S. Patent No. 7,461,353	U.S. Patent No. 7,831,926	Comparison
instructions further enables the device to: [10.b] for each of the plurality of images; [10.b.1] retrieve the image via the URL for the image; and [10.b.2] generate a respective bounding box for the image; and [10.c] employ the respective bounding boxes to scale images in the rendered views of the HTML document. [21.a] The mobile hand-held device of claim 20, wherein the HTML document includes references to a plurality of images, each reference comprising a respective Universal Resource Locators	[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.	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	Separately, preserved hyperlink functionality addresses retrieval/navigation via hyperlink elements. Because images are a type of object, it is equivalent to generate bounding boxes for the image or the object. Equivalent because using bounding boxes to scale images is a mechanism for scaling content across the display.

U.S. Patent No. 10,083,154	U.S. Patent No. 7,461,353	U.S. Patent No. 7,831,926	Comparison
(URLs), and wherein execution of the instructions performs further operations comprising: [21.b] for each of the plurality of images; [21.b.1] retrieving the image via the URL for the image; and [21.b.2] generating a respective bounding box for the image; and [21.c] employing the respective bounding boxes to scale images in the rendered views of the HTML document.		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. [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 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	

Declaration of Dr. Benjamin Goldberg
U.S. Patent No. 10,083,154

U.S. Patent No. 10,083,154	U.S. Patent No. 7,461,353	U.S. Patent No. 7,831,926	Comparison
		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. [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	

U.S. Patent No. 10,083,154	U.S. Patent No. 7,461,353	U.S. Patent No. 7,831,926	Comparison
		display page occupied by the object's associated group of content.	
[11.a] The mobile handheld device of claim 1, wherein the HTML document includes a reference to an XML document containing XML content, and wherein execution of the instructions further enables the mobile handheld device to: [11.b] retrieve the XML document and processing the XML content in connection with rendering the first representation of the HTML document, wherein the XML content is used to define one or more of the	[36.c] retrieving the Web page via the wireless communications device, and processing HTML-based Web content to produce scalable content; and	Appendix C (IPR2013-00004) [52.f] retrieving and processing the HTML-based Web content to produce scalable content; and 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 as interpreted by the rendering engine; and	The unpatentable claims describe a mobile device executing instructions to access a Web page with “HTML-based content.” It would have been standard for HTML to include references to external documents (like XML, CSS, or scripts) to define content and layout. The unpatentable limitations recite parsing “markup language code” and “HTML-based code,” which encompasses XML (Extensible Markup Language) because it is a type of markup language. The

U.S. Patent No. 10,083,154	U.S. Patent No. 7,461,353	U.S. Patent No. 7,831,926	Comparison
interpreted page layout, functionality, and design of the content associated with the HTML document. [24.a] The method of claim 13, wherein the HTML document includes a reference to an XML document containing XML content, further comprising: [24.b] retrieving the XML document and processing the XML content in connection with rendering the first representation of the HTML document, wherein the XML content is used to define one or more of the interpreted page layout, functionality, and design of the content associated		Appendix E (95/000,635) [77] The mobile device of claim 52, wherein a portion of the HTML-based Web content comprises XML-based content.	unpatentable limitations recite that a portion of the HTML-based Web content comprises XML-based content ([77]), and the unpatentable claims' parsing/processing of markup language code and HTML-based content is consistent with processing XML-based content referenced by or included with HTML during rendering.

U.S. Patent No. 10,083,154	U.S. Patent No. 7,461,353	U.S. Patent No. 7,831,926	Comparison
with the HTML document.			
[12] The method of claim 13, wherein the primary datum is an origin in a two-dimensional XY coordinate system and the vectors from the primary datum to the object datums for the HTML object bounding boxes comprise XY coordinates in the coordinate system. [25] The method of claim 13, wherein the primary datum is an origin in a two-dimensional XY coordinate system and the vectors from the primary datum to the object datums for the HTML object bounding	[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 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	Both this limitation and the unpatentable limitations recite primary datums and vectors to object datums. Because the XY coordinate system can represent the page layout, the primary datum can be an origin and the vectors can include XY coordinates. Because the bounding boxes define the dimensions of content, the HTML object bounding boxes correspond to the layout location datum for the object's associated display content.

Declaration of Dr. Benjamin Goldberg
U.S. Patent No. 10,083,154

U.S. Patent No. 10,083,154	U.S. Patent No. 7,461,353	U.S. Patent No. 7,831,926	Comparison
boxes comprise XY coordinates in the coordinate system.		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 location of the point after transformation, and SF is the scale factor.	

Declaration of Dr. Benjamin Goldberg
U.S. Patent No. 10,083,154

U.S. Patent No. 10,083,154	U.S. Patent No. 7,461,353	U.S. Patent No. 7,831,926	Comparison
		[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.	

Declaration of Dr. Benjamin Goldberg
U.S. Patent No. 10,083,154

U.S. Patent No. 10,083,154	U.S. Patent No. 7,461,353	U.S. Patent No. 7,831,926	Comparison
		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.	

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.