

Declaration of Vijay K. Madiseti, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 6,546,397 B1

UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE PATENT TRIAL AND APPEAL BOARD

FACEBOOK, INC.,
Petitioner

v.

EXPRESS MOBILE INC.,
Patent Owner

U.S. Patent No. 6,546,397 B1
Issue Date: April 8, 2003

Title: Browser Based Web Site Generation Tool and Run Time Engine

DECLARATION OF VIJAY K. MADISETTI, PH.D.

Table of Contents

	Page
I. INTRODUCTION AND QUALIFICATIONS	1
A. Qualifications and Experience	1
B. Materials Considered.....	8
II. PERSON OF ORDINARY SKILL IN THE ART	10
III. STATEMENT OF LEGAL PRINCIPLES.....	12
A. Claim Construction	12
B. Obviousness.....	14
IV. THE '397 PATENT	20
A. Overview of the Specification.....	20
B. The Challenged Claim.....	22
V. CLAIM CONSTRUCTION	23
VI. APPLICATION OF THE PRIOR ART TO ASSERTED CLAIMS	25
A. Brief Summary of Prior Art	26
1. Reynolds [Ex. 1003]	26
2. Bernardo [Ex. 1004].....	28
3. Meyer [Ex. 1005]	30
4. Coombs [Ex. 1006]	32
5. Witkowski [Ex. 1007].....	32
B. Ground 1: Claim 1 is obvious over Reynolds, Bernardo, Meyer and Witkowski.....	34
1. Independent Claim 1	34
(a) “A method to allow users to produce Internet websites on and for computers having a browser and a virtual machine capable of generating displays, said method comprising:” (Claim 1[preamble])	35

Table of Contents

(continued)

Page

(b)	“presenting a viewable menu having a user selectable panel of settings describing elements on a website, said panel of settings being presented through a browser on a computer adapted to accept one or more of said selectable settings in said panel as inputs therefrom, and where at least one of said user selectable settings in said panel corresponds to commands to said virtual machine;” (Claim 1[a])	46
(1)	“presenting a viewable menu having a user selectable panel of settings describing elements on a website, said panel of settings being presented through a browser on a computer adapted to accept one or more of said selectable settings in said panel as inputs therefrom...” (Claim 1[a][1])	46
(2)	“. . . and where at least one of said user selectable settings in said panel corresponds to commands to said virtual machine;” (Claim 1[a][2])	62
(c)	“generating a display in accordance with one or more user selected settings substantially contemporaneously with the selection thereof;” (Claim 1[b])	68
(d)	“storing information representative of said one or more user selected settings in a database;” (Claim 1[c])	73
(e)	“generating a website at least in part by retrieving said information representative of said one or more user selected settings stored in said database; and” (Claim 1[d])	78

Table of Contents
(continued)

	Page
(f) “building one or more web pages to generate said website from at least a portion of said database and at least one run time file, where said at least one run time file utilizes information stored in said database to generate virtual machine commands for the display of at least a portion of said one or more web pages.” (Claim 1[e]).	82
C. Ground 2: Claim 1 is obvious over Reynolds, Bernardo, Meyer, and Witkowski, in Further View of Coombs	88
VII. NO SECONDARY CONSIDERATIONS OF NON-OBVIOUSNESS	94
VIII. CONCLUSION	98

Declaration of Vijay K. Madiseti, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 6,546,397 B1

I, Vijay K. Madiseti, Ph.D., declare as follows:

I. INTRODUCTION AND QUALIFICATIONS

A. Qualifications and Experience

1. I received my Bachelor of Technology (Honors) in Electronics and Electrical Communication Engineering at the Indian Institute of Technology (IIT) in Kharagpur, India in 1984. I obtained my Ph.D. in Electrical Engineering and Computer Science at the University of California, Berkeley, in 1989. I received the Demetri Angelakos Outstanding Graduate Student Award from the University of California, Berkeley and the IEEE/ACM Ira M. Kay Memorial Paper Prize in 1989.

2. I am a tenured Professor in Electrical and Computer Engineering at the Georgia Institute of Technology (“Georgia Tech”). I am knowledgeable and familiar with wireless communications, microprocessor architecture, hardware, RF, cellular networks, ASIC design, computer engineering, embedded systems, digital signal processing, and associated software and firmware design for wireless and telecommunications terminals and base stations. I have created and taught undergraduate and graduate courses in hardware and software design for signal processing and wireless communication circuits at Georgia Tech for the past twenty years. Additionally, I have been active in the areas of wireless communications, digital signal processing, integrated circuit design (analog & digital), software engineering, system-level design methodologies and tools, and software systems. I

Declaration of Vijay K. Madiseti, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 6,546,397 B1

have been the principal investigator (“PI”) or co-PI in several active research programs in these areas, including DARPA’s Rapid Prototyping of Application Specific Signal Processors, the State of Georgia’s Yamacraw Initiative, the United States Army’s Federated Sensors Laboratory Program, and the United States Air Force Electronics Parts Obsolescence Initiative. I have received an IBM Faculty Award and NSF’s Research Initiation Award.

3. I have designed several specialized computer and communication systems over the past two decades at Georgia Tech for tasks such as wireless audio and video processing and protocol processing for portable platforms, such as cell phones and PDAs. I have worked on designing systems that are efficient from performance, size, weight, area, and thermal considerations. I have developed courses and classes for the industry on these topics, and many of my lectures in advanced computer system design, developed under the sponsorship of the United States Department of Defense in the late 1990s, are available for educational use at “<http://www.eda.org/rassp>” and have been used by several U.S. and international universities as part of their course work. Some of my recent publications in the area of design of wireless communications systems and associated protocols are listed in Exhibit A. I graduated more than 20 Ph.D. students that now work as professors or in technical positions around the world.

Declaration of Vijay K. Madiseti, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 6,546,397 B1

4. I have been active in research in the area of the World Wide Web and some of my peer-reviewed publications in this area include (i) Arshdeep Bahga and Vijay K. Madiseti, Performance Evaluation Approach for Multi-Tier Cloud Applications, *Journal of Software Engineering and Applications* 74-83 (2013); (ii) Arshdeep Bahga and Vijay K. Madiseti, Synthetic Workload Generation for Cloud Computing Applications, *Journal of Software Engineering and Applications* 396-410 (2011); (iii) Venu Dasigi and Vijay Madiseti, A Web-Based Interface for a Digital Broadband Home, *Yamacraw IAB Workshop* (2000); and (iv) Arshdeep Bahga and Vijay K. Madiseti, Rapid Prototyping of Multitier Cloud-Based Services and Systems, *Computer*, vol. 46, no. 11, pp. 76-83 (Nov. 2013).

5. I have been an active consultant to industry and various research laboratories (including Massachusetts Institute of Technology Lincoln Labs and Johns Hopkins University Applied Physics Laboratory). My consulting work for MIT Lincoln Labs involved high resolution imaging for defense applications, where I worked in the area of prototyping complex and specialized computing systems. My consulting work for the Johns Hopkins Applied Physics Lab (“APL”) mainly involved localization of objects in image fields, where I worked on identifying targets in video and other sensor fields and identifying computer architectures and circuits for power and space-efficient designs.

Declaration of Vijay K. Madiseti, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 6,546,397 B1

6. I have founded three companies in the areas of embedded software, military chipsets involving imaging technology, and wireless communications. I have supervised the Ph.D. dissertations of over twenty engineers in the areas of computer engineering, signal processing, communications, rapid prototyping, and system-level design methodology, of which five have resulted in thesis prizes or paper awards. The first of the companies I founded, VP Technologies, offers products in the area of semiconductor integrated circuits, including building computing systems for imaging systems for avionics electronics for the United States Air Force and the US Navy, since 1995. I remain a director of VP Technologies. The second of these companies, Soft Networks, LLC, offers software for multimedia and wireless computing platforms, including the development of a set-top box for Intel that decodes MPEG-2 video streams, wireless protocol stacks, and imaging codecs for multimedia phones. The technology involved with the design, development, and implementation of the Intel set-top box included parsing the bit streams, decoding communications protocols, extracting image and video data, and then processing for subsequent display or storage. The third of these companies, Elastic Video, uses region of interest based video encoding or decoding for capturing high quality video at very low bit rates, with primary application for wireless video systems.

Declaration of Vijay K. Madiseti, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 6,546,397 B1

7. I have authored more than sixty refereed journal publications and around forty peer reviewed conference publications. I have been active in research in the area of wireless and mobile communications and some of my recent peer-reviewed publications in this area include: (i) Mustafa Turkboylari & Vijay K. Madiseti, Effect of Handoff Delay on the System Performance of TDMA Cellular Systems, Proceedings of the Fourth IEEE Conference on Mobile and Wireless Communications Network 411-15 (Sept. 9-11, 2002); (ii) Loran A. Jatunov & Vijay K. Madiseti, Computationally-Efficient SNR Estimation for Bandlimited Wideband CDMA Systems, 5 IEEE Transactions on Wireless Communications, no. 12 (2006) at 3480-91; and (iii) Nimish Radio, Ying Zhang, Mallik Tatipamula & Vijay K. Madiseti, Next Generation Applications on Cellular Networks: Trends, Challenges, and Solutions, 100 Proceedings of the IEEE, no. 4 (April 2012) at 841-54. I have extensive experience analyzing, designing, and testing systems based on 3GPP Technical Specifications, including specifications describing WCDMA and HSDPA technologies. I have been active in the area of location-based services and wireless localization techniques since the mid-1990s, and have authored several papers on location-based services, including, Vijay K. Madiseti et al., Mobile Fleet Application Using SOAP and System on Devices (SyD) Middleware Technologies, Communications, Internet, and Information Technology (2002) at 426-31. I have

Declaration of Vijay K. Madiseti, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 6,546,397 B1

served as associate editor or on the editorial board for technical journals, including IEEE Transactions on Circuits & Systems II, International Journal in Computer Simulation, and International Journal in VLSI Signal Processing.

8. I have authored or co-authored several books, including VLSI Digital Signal Processors (IEEE Press 1995) and the Digital Signal Processing Handbook (CRC Press, 1998, 2010). I co-authored Quick-Turnaround ASIC Design in VHDL (Kluwer Academic Press 1996) and Platform-Centric Approach to System-on-Chip (SoC) Design (Springer 2004). I am also the editor of several books, including the three-volume DSP Handbook set: Volume 1: Digital Signal Processing Fundamentals, Volume 2: Video, Speech, and Audio Signal Processing and Associated Standards, and Volume 3: Wireless, Networking, Radar, Sensory Array Processing, and Nonlinear Signal Processing, published in 2010 by CRC Press, Boca Raton, Florida. More recently I have authored Cloud Computing (2014, CreateSpace Press), and Internet of Things (2014, CreateSpace), and the book, Cloud Computing, was nominated as a Notable Book of 2014 by the Association of Computing Machinery (ACM) in July 2015.

9. My experience is relevant to this case. I have been working in the area of wireless communications and signal processing, since the early 1980s. In 1983-1984, I designed and prototyped a very low RF frequency (VLF) receiver for

Declaration of Vijay K. Madiseti, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 6,546,397 B1

submarine communications utilizing MSK (Minimum Shift Key) modulation/demodulation techniques in hardware. I have developed wireless baseband and protocol stack software and assembly code for a leading telecommunications handset vendor that focused on efficient realization of speech codecs and echo-cancellation. My work in this regard included creation of software code, and analysis and revision of existing software code. In the early 2000-2001 timeframe, I designed three GSM multiband mobile phones for a leading telecom equipment manufacturer in Asia.

10. I have been elected a Fellow of the IEEE, for contributions to embedded computing systems. The Fellow is the highest grade of membership of the IEEE, a world professional body consisting of over 300,000 electrical and electronics engineers, with only one-tenth of one percent (0.1%) of the IEEE membership being elected to the Fellow grade each year. Election to Fellow is based upon votes cast by existing Fellows in IEEE. I have also been awarded the 2006 Frederick Emmons Terman Medal by the American Society of Engineering Education for contributions to Electrical Engineering, including authoring a widely used textbook in the design of VLSI digital signal processors. I was awarded VHDL International Best PhD Dissertation Advisor Award in 1997 and the NSF RI Award in 1990. I was Technical Program Chair for both the IEEE MASCOTS in 1994 and the IEEE Workshop on

Declaration of Vijay K. Madiseti, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 6,546,397 B1

Parallel and Distributed Simulation in 1990. In 1989 I was recognized with the Ira Kay IEEE/ACM Best Paper Award for Best Paper presented at the IEEE Annual Simulation Symposium.

11. I have submitted approximately thirty invention disclosures and provisional patents over the past ten years. None of them are patents yet.

12. I have testified as an expert witness before. Over the past six years, I've testified as an expert in more than 20 proceedings. About half of the proceedings in which I have testified as an expert were in the area of 2G/3G/4G wireless receiver design, including six to seven in the area of digital physical layer design.

13. I have attached a more detailed list of my qualifications as **Exhibit A**.

14. I am being compensated for my time working on this matter at my standard hourly rate plus expenses. I do not have any personal or financial stake or interest in the outcome of the present proceeding, and my compensation is not dependent on the outcome of this IPR and in no way affects the substance of my statements in this Declaration.

B. Materials Considered

15. The analysis that I provide in this Declaration is based on my education and experience in the field of computer systems, as well as the documents I have considered, including U.S. Patent No. 6,546,397 B1 ("397 patent") [**Ex. 1001**] and

Declaration of Vijay K. Madiseti, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 6,546,397 B1

its prosecution history. The '397 patent states on its face that it issued from an application filed on December 2, 1999. For purposes of this Declaration, I have assumed December 2, 1999 as the effective filing date for the '397 patent. I have cited to the following documents in my analysis below:

Exhibit No.	Description of Document
1001	U.S. Patent No. 6,546,397 B1 to Steven H. Rempell (filed December 2, 1999, issued April 8, 2003)
1003	U.S. Patent No. 6,052,717 to Brian Reynolds et al. (filed October 23, 1996, issued April 18, 2000)
1004	U.S. Patent No. 6,684,369 B1 to Richard S. Bernardo et al. (filed June 19, 1998, issued January 27, 2004)
1005	Excerpts from Jon Meyer et al., <i>Java Virtual Machine</i> (1997)
1006	Excerpts from Jason Coombs et al., <i>Setting up an Internet Site for Dummies</i> (3d ed. 1998)
1007	U.S. Patent No. 6,125,360 to Andrew Witkowski et al. (filed July 2, 1998, issued September 26, 2000)
1008	Excerpts from <i>Microsoft Press Computer Dictionary</i> (3d ed. 1997)
1009	U.S. Patent No. 4,464,652 to William F. Lapson et al. (filed July 19, 1982, issued August 7, 1984)
1010	U.S. Patent No. 5,729,594 to Edwin E. Klingman (filed June 7, 1996, issued March 17, 1998)
1011	Prosecution History of U.S. Patent No. 6,546,397

Declaration of Vijay K. Madiseti, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 6,546,397 B1

Exhibit No.	Description of Document
1016	Amended Order Construing Claims (Dkt. 81) in <i>X Commerce, Inc. d/b/a Magento, Inc. v. Express Mobile, Inc.</i> , No. 3:17-cv-02605 (N.D. Cal. Sept. 12, 2018)
1017	Oral Order (Dkt. 35) in <i>Express Mobile, Inc. v. eGrove Systems Corporation</i> , No. 1:17-cv-00703 (D. Del. Aug. 22, 2018)
1018	Claim Construction Memorandum Opinion and Order (Dkt. 97) in <i>Express Mobile, Inc. v. Svanaco, Inc.</i> , No. 2:17-cv-00130 (E.D. Tex. Feb. 7, 2018)
1020	Order on Claim Construction (Dkt. 142) in <i>Shopify Inc. et al v. Express Mobile, Inc.</i> , No. 1:19-cv-00439 (D. Del. June 30, 2020)
1021	Memorandum Opinion (Dkt. 121) in <i>Express Mobile, Inc. v. GoDaddy.Com, LLC</i> , No. 1:19-cv-01937 (D. Del. June 1, 2021)
1023	Joint Claim Construction Brief (Dkt. 33) in <i>Express Mobile, Inc. v. eGrove Systems Corporation</i> , No. 1:17-cv-00703 (D. Del. Aug. 22, 2018)

II. PERSON OF ORDINARY SKILL IN THE ART

16. I understand that an assessment of claims of the '397 patent should be undertaken from the perspective of a person of ordinary skill in the art as of the earliest claimed priority date, which I understand is December 2, 1999. I have also been advised that to determine the appropriate level of a person having ordinary skill in the art, the following factors may be considered: (1) the types of problems encountered by those working in the field and prior art solutions thereto; (2) the sophistication of the technology in question, and the rapidity with which innovations

Declaration of Vijay K. Madiseti, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 6,546,397 B1

occur in the field; (3) the educational level of active workers in the field; and (4) the educational level of the inventor.

17. The '397 patent describes its field of invention as generally relating to the creation of web sites. ('397, 1:5-8 (describing the "Field of the Invention" as "directed to computing systems, and more particularly to methods and apparatus for building a web site using a browser-based build engine").) In my opinion, a person of ordinary skill in the art as of December 1999 would have possessed at least a bachelor's degree in software engineering, computer science, computer engineering, or electrical engineering with at least two years of experience in web-based software application development, including experience in developing software and systems for storing, retrieving, and transmitting information (such as text and images) over a computer network such as the Internet (or equivalent degree or experience). A person could also have qualified as a person of ordinary skill in the art with some combination of (1) more formal education (such as a master's of science degree) and less technical experience or (2) less formal education and more technical or professional experience in the fields listed above. Acquired as part of the person's basic computer education and/or experience, as the '397 patent makes clear, a person of ordinary skill in the art would have had a working knowledge about various ways to implement websites using browser-based input and provide those websites to

browsers upon request.

18. My opinions regarding the level of ordinary skill in the art are based on, among other things, my more than 30 years of experience in computer science, my understanding of the basic qualifications that would be relevant to an engineer or scientist tasked with investigating methods and systems in the relevant area, and my familiarity with the backgrounds of colleagues, co-workers, and employees, both past and present.

19. Although my qualifications and experience exceed those of the hypothetical person having ordinary skill in the art defined above, my analysis and opinions regarding the '397 patent have been based on the perspective of a person of ordinary skill in the art as of December 1999.

III. STATEMENT OF LEGAL PRINCIPLES

A. Claim Construction

20. The following sets forth my understanding of the legal principles of claim construction from counsel. I understand that under the legal principles, claim terms are generally given their ordinary and customary meaning, which is the meaning that the term would have to a person of ordinary skill in the art in question at the time of the invention, i.e., as of the effective filing date of the patent application. I further understand that the person of ordinary skill in the art is deemed to read the claim term not only in the context of the particular claim in which a claim

Declaration of Vijay K. Madiseti, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 6,546,397 B1

term appears, but in the context of the entire patent, including the specification.

21. I am informed by counsel that the patent specification, under the legal principles, has been described as the single best guide to the meaning of a claim term, and is thus highly relevant to the interpretation of claim terms. And I understand for terms that do not have a customary meaning within the art, the specification usually supplies the best context of understanding the meaning of those terms.

22. I am further informed by counsel that other claims of the patent in question, both asserted and unasserted, can be valuable sources of information as to the meaning of a claim term. Because the claim terms are normally used consistently throughout the patent, the usage of a term in one claim can often illuminate the meaning of the same term in other claims. Differences among claims can also be a useful guide in understanding the meaning of particular claim terms.

23. I understand that the prosecution history can further inform the meaning of the claim language by demonstrating how the inventors understood the invention and whether the inventors limited the invention in the course of prosecution, making the claim scope narrower than it otherwise would be. Extrinsic evidence may also be consulted in construing the claim terms, such as my expert testimony.

24. I have been informed by counsel that, in *Inter Partes* Review (IPR)

proceedings, a claim of a patent shall be construed using the same claim construction standard that would be used to construe the claim in a civil action filed in a U.S. district court (which I understand is called the “*Phillips*” claim construction standard), including construing the claim in accordance with the ordinary and customary meaning of such claim as understood by one of ordinary skill in the art and the prosecution history pertaining to the patent.

25. I have been instructed by counsel to apply the “*Phillips*” claim construction standard for purposes of interpreting the claims in this proceeding, to the extent they require an explicit construction. The description of the legal principles set forth above thus provides my understanding of the “*Phillips*” standard as provided to me by counsel.

B. Obviousness

26. The following sets forth my understanding of the legal principles of obviousness from counsel. I understand that a patent claim is obvious if, as of the critical date (i.e., either the earliest claimed priority date (pre-AIA) or the effective filing date (AIA)), it would have been obvious to a person having ordinary skill in the field of the technology (the “art”) to which the claimed subject matter belongs.

27. I understand that the following factors should be considered in analyzing obviousness: (1) the scope and content of the prior art; (2) the differences

Declaration of Vijay K. Madiseti, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 6,546,397 B1

between the prior art and the claims; and (3) the level of ordinary skill in the pertinent art. I also understand that certain other facts known as “secondary considerations” such as commercial success, unexplained results, long felt but unsolved need, industry acclaim, simultaneous invention, copying by others, skepticism by experts in the field, and failure of others may be utilized as indicia of nonobviousness. I understand, however, that secondary considerations should be connected, or have a “nexus,” with the invention claimed in the patent at issue.

28. I understand that a reference qualifies as prior art for obviousness purposes when it is analogous to the claimed invention. The test for determining what art is analogous is: (1) whether the art is from the same field of endeavor, regardless of the problem addressed, and (2) if the reference is not within the field of the inventor’s endeavor, whether the reference still is reasonably pertinent to the particular problem with which the inventor is involved.

29. I understand that a person of ordinary skill in the art is assumed to have knowledge of all prior art. I understand that one skilled in the art can combine various prior art references based on the teachings of those prior art references, the general knowledge present in the art, or common sense. I understand that a motivation to combine references may be implicit in the prior art, and there is no requirement that there be an actual or explicit teaching to combine two references.

Declaration of Vijay K. Madiseti, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 6,546,397 B1

Thus, one may take into account the inferences and creative steps that a person of ordinary skill in the art would employ to combine the known elements in the prior art in the manner claimed by the patent at issue. I understand that one should avoid “hindsight bias” and ex post reasoning in performing an obviousness analysis. But this does not mean that a person of ordinary skill in the art for purposes of the obviousness inquiry does not have recourse to common sense.

30. I understand that when determining whether a patent claim is obvious in light of the prior art, neither the particular motivation for the patent nor the stated purpose of the patentee is controlling. The primary inquiry has to do with the objective reach of the claims, and that if those claims extend to something that is obvious, then the entire patent claim is invalid.

31. I understand one way that a patent can be found obvious is if there existed at the time of the invention a known problem for which there was an obvious solution encompassed by the patent’s claims. I understand that a motivation to combine various prior art references to solve a particular problem may come from a variety of sources, including market demand or scientific literature. I understand that a need or problem known in the field at the time of the invention can also provide a reason to combine prior art references and render a patent claim invalid for obviousness. I understand that familiar items may have obvious uses beyond their

Declaration of Vijay K. Madiseti, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 6,546,397 B1

primary purpose, and that a person of ordinary skill in the art will be able to fit the teachings of multiple prior art references together like the pieces of a puzzle. I understand that a person of ordinary skill is also a person of at least ordinary creativity. I understand when there is a design need or market pressure to solve a problem and there are a finite number of identified, predictable solutions, a person of ordinary skill has good reason to pursue the known options within his or her technical grasp. If this finite number of predictable solutions leads to the anticipated success, I understand that the invention is likely the product of ordinary skill and common sense, and not of any sort of innovation. I understand that the fact that a combination was obvious to try might also show that it was obvious, and hence invalid, under the patent laws. I understand that if a patent claims a combination of familiar elements according to known methods, the combination is likely to be obvious when it does not more than yield predictable results. Thus, if a person of ordinary skill in the art can implement a predictable variation, an invention is likely obvious. I understand that combining embodiments disclosed near each other in a prior art reference would not ordinarily require a leap of inventiveness.

32. I understand that obviousness may be shown by demonstrating that it would have been obvious to modify what is taught in a single piece of prior art to create the patented invention. Obviousness may also be shown by demonstrating

Declaration of Vijay K. Madiseti, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 6,546,397 B1

that it would have been obvious to combine the teachings of more than one item of prior art. I understand that a claimed invention may be obvious if some teaching, suggestion, or motivation exists that would have led a person of ordinary skill in the art to combine the invalidating references. I also understand that this suggestion or motivation may come from the knowledge of a person having ordinary skill in the art, or from sources such as explicit statements in the prior art. I understand that when there is a design need or market pressure, and there are a finite number of predictable solutions, a person of ordinary skill may be motivated to apply common sense and his skill to combine the known options in order to solve the problem.

33. I understand the following are examples of approaches and rationales that may be considered in determining whether a piece of prior art could have been combined with other prior art or with other information within the knowledge of a person having ordinary skill in the art:

- (1) Some teaching, motivation, or suggestion in the prior art that would have led a person of ordinary skill to modify the prior art reference or to combine prior art reference teachings to arrive at the claimed invention;
- (2) Known work in one field of endeavor may prompt variations of it for use in the same field or a different field based on design incentives or other market forces if the variations would have been predictable to a person of ordinary skill in the art;

Declaration of Vijay K. Madiseti, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 6,546,397 B1

- (3) Combining prior art elements according to known methods to yield predictable results;
- (4) Applying a known technique to a known device, method, or product ready for improvement to yield predictable results;
- (5) Applying a technique or approach that would have been “obvious to try” (choosing from a finite number of identified, predictable solutions, with a reasonable expectation of success);
- (6) Simple substitution of one known element for another to obtain predictable results; or
- (7) Use of a known technique to improve similar products, devices, or methods in the same way.

34. I understand that, when determining whether a claimed combination is obvious, the correct analysis is not whether one of ordinary skill in the art, writing on a blank slate, would have chosen the particular combination of elements described in the claim. Instead, I understand the correct analysis considers whether one of ordinary skill, facing the wide range of needs created by developments in the field of endeavor, would have seen a benefit to selecting the combination claimed.

35. I understand that the test for obviousness is not whether the features of a secondary reference may be bodily incorporated into the structure of the primary reference. The test for obviousness, in other words, is not whether the references could be physically combined but whether the claimed inventions are rendered obvious by the teachings of the prior art as a whole.

IV. THE '397 PATENT

A. Overview of the Specification

36. The '397 patent, entitled "Browser Based Web Site Generation Tool and Run Time Engine," is generally directed to a system for "designing and building a web page." ('397 patent, Abstract.) The '397 patent issued on April 8, 2003 from a patent application filed on December 2, 1999.

37. The '397 patent suggests that "conventional implementations" of web page editors were deficient because they "utilize[d] full-featured programming language[s], such as C++ or Visual Basic" that were not supported by "current popular browsers." ('397, 1:23-29.) Because "current popular browsers do not support these languages," the patent claims, existing web page editors could "offer only a crude preview capability of what a real web page will look like." ('397, 1:23-31.) The '397 patent purports to improve on existing technologies through a "Browser Based build engine that is written entirely in a web based full featured programming language (e.g. JAVA)." ('397, 1:52-54.) The '397 patent does not claim to have invented Java, and as explained below, Java was well-known.

38. The web page editor includes a browser-based "user interface" including a "panel" with "one or more menus for controlling the form of content to be placed on the web page." ('397, Abstract; *see also id.*, 2:33-37, 10:16-52.) Changes to the web page are presented in a "build frame" as a what-you-see-is-what-

you-get (WYSIWYG) representation of the web page. ('397, Abstract.) The completed web page is stored into a database, which is uploaded to a user's web site along with an "HTML Shell File" and a "run time engine." ('397, 2:37-42, 5:53-57.) The '397 patent describes the database, HTML shell, and runtime engine as "run time files." (*Id.*, 5:53-57.) When a web surfer views the website, the runtime engine is called by the browser to read the database and "draw" a web page. (*Id.*, 5:57-62.) The process of accepting user input, creating a database and other "run time files," and uploading those files to a website for other users to view is illustrated in Fig. 2, reproduced below:

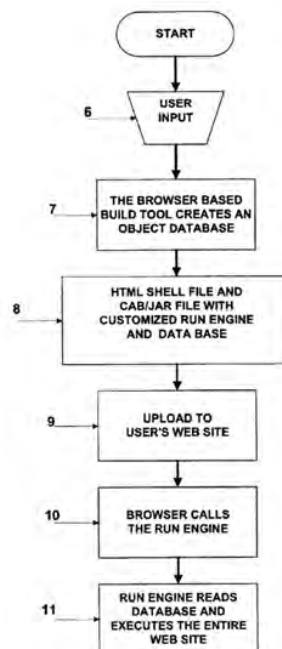


Fig. 2

('397, Fig. 2.)

B. The Challenged Claim

39. This Declaration addresses claim 1, which recites in its entirety:

1. A method to allow users to produce Internet websites on and for computers having a browser and a virtual machine capable of generating displays, said method comprising:

[a][1] (a) presenting a viewable menu having a user selectable panel of settings describing elements on a website, said panel of settings being presented through a browser on a computer adapted to accept one or more of said selectable settings in said panel as inputs therefrom,

[a][2] and where at least one of said user selectable settings in said panel corresponds to commands to said virtual machine;

[b] (b) generating a display in accordance with one or more user selected settings substantially contemporaneously with the selection thereof;

[c] (c) storing information representative of said one or more user selected settings in a database;

[d] (d) generating a website at least in part by retrieving said information representative of said one or more user selected settings stored in said database; and

[e] (e) building one or more web pages to generate said website from at least a portion of said database and at least one run time file, where said at least one run time file utilizes information stored in said database to

generate virtual machine commands for the display of at least a portion
of said one or more web pages.

(’397, 65:44-66:2 (Claim 1).)

V. CLAIM CONSTRUCTION

40. In addition to the “*Phillips*” claim construction standard, I have been asked to apply each of the claim constructions of the terms recited in claim 1, listed below. I understand that the constructions listed below were proposed and/or adopted in prior litigations involving the ’397 patent. My mapping of the prior art to claim 1, as demonstrated in **Part VI** below, does not depend on the Board adopting any of these particular definitions. To the contrary, in my opinion, the differences between the various constructions are immaterial as the prior art renders claim 1 obvious under each of these definitions:

Term	Construction(s)
“virtual machine”	• abstract machine emulated in software (EX1017) • abstract machine that is emulated in software and that executes intermediate code (EX1018, p.10) • an abstract machine that is not built in hardware but is emulated in software (EX1016, p.6) • software that emulates a physical machine (EX1020, p.1)
“at least one run time file”	• one or more files, including a run time engine, that are downloaded or created when a browser is pointed to a web page or website (EX1017; EX1020, p.2; EX1021, p. 11)

Declaration of Vijay K. Madiseti, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 6,546,397 B1

Term	Construction(s)
	• one or more files, including a run time engine, that are downloaded or created, and executed by a browser to display a webpage when the browser is pointed to a web page (EX1016, p.8) • one or more files that are downloaded or created when a browser is pointed to a web page or website (EX1023, p. 25 (proposed by Patent Owner))
“run time engine” ¹	• file that is executed at runtime that utilizes information from the database and generates commands to display a web page or website (EX1021, p.16 (proposed by Patent Owner)) • file that is executed at runtime that facilitates the retrieval of information from the database and generates commands to display a web page or website (EX1017) • file that is executed at runtime that reads information from the database and generates commands to display a web page or website (EX1016, pp.7-8; EX1020, p.1; EX2021, p.17)
“substantially contemporaneously”	• happening at the same period of time from a human perspective (EX1018, p.16) • occurring immediately after (EX1016, p.11) • substantially in the same time period, from a human perspective (EX1018, p.13 (proposed by Patent Owner))

¹ The sole challenged claim (*i.e.* claim 1) does not recite the term “**run time engine**,” but I have included it in the table because two of the definitions of “**run time file**” set forth in the table use this term as part of their definitions.

Term	Construction(s)
“database”	- an electronic information storage system offering data storage and retrieval (EX1016, p.9) - an electronic information storage system offering data storage and retrieval and that stores information on a record-by-record basis, each record divided into one or more fields (EX1018, p.19; EX1020, p. 2)
“storing information representative of said one or more user selected settings in a database”	storing data in a database, which data pertains to one or more attributes of an object available for selection by a user (EX1020, p.2)
“settings”	attributes of an object available for selection (EX1020, p.1)

VI. APPLICATION OF THE PRIOR ART TO ASSERTED CLAIMS

41. I have reviewed and analyzed the prior art references and materials listed in **Part I.B** above. In my opinion, the claims of the '397 patent are rendered obvious based on the following prior art:

Ground	Basis for Challenge Under §103(a)	Claim
1	Reynolds (Ex. 1003), Bernardo (Ex. 1004), Meyer (Ex. 1005), and Witkowski (Ex. 1007)	1
2	Reynolds, Bernardo, Meyer, Witkowski, and Coombs (Ex. 1006)	1

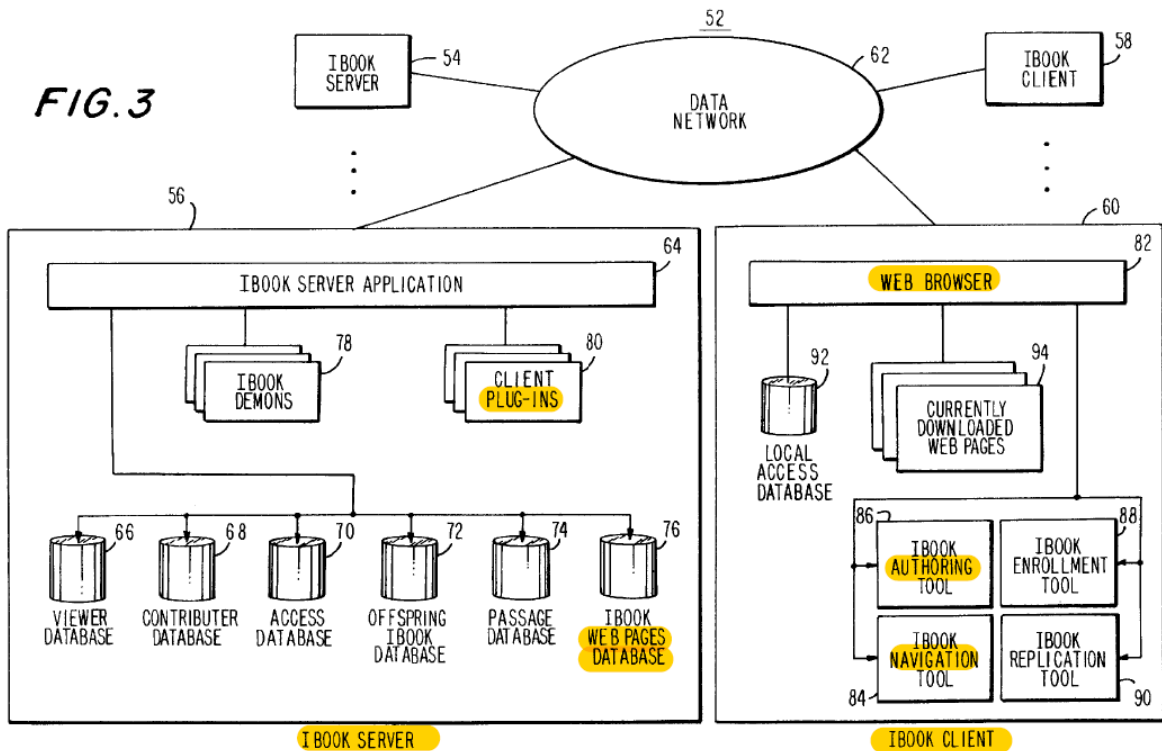
42. I am informed by counsel that each reference listed above qualifies as prior art to the challenged claims at least because each reference was filed and/or published before the earliest claimed priority date for the '397 patent.

A. Brief Summary of Prior Art

1. Reynolds [Ex. 1003]

43. U.S. Patent No. 6,052,717 to Brian Reynolds et al. is entitled “Interactive Web Book System.” I understand from counsel that Reynolds qualifies as prior art to the '397 patent because it was filed before the effective filing date of the '397 patent.

44. Reynolds teaches a system for creating web sites in the form of interactive Web books (also referred to as “ibooks”) that are organized as collections of “Web pages.” (Reynolds, 1:24-30; *see also id.*, Abstract.) Reynolds’ system for creating ibooks is depicted in Fig. 3, reproduced and annotated below.



(Reynolds, Fig. 3 (annotation added).) Reynolds explains that the material for ibook web pages are stored in an “ibook web pages database 76” at a server, as shown in Figure 3 above. (Reynolds, 7:35-39.) To edit or view these pages, a user downloads plug-ins 80 from the ibook server that extend the capabilities of the user’s web browser 82. (Reynolds, 5:27-31.) When downloaded, client plug-ins form browser tools including “authoring tool 86” used to create web page content such as text, multimedia, and executable programs stored at the server’s database (Reynolds, 5:27-37, 7:37-41, 8:4-11), and “navigation tool 84” used to view ibook web pages by downloading and displaying the content from the database. (Reynolds, 11:24-

39.)

2. Bernardo [Ex. 1004]

45. U.S. Patent No. 6,684,369 to Richard S. Bernardo et al. is entitled “Web Site Creator Using Templates.” I understand from counsel that Bernardo qualifies as prior art to the ’397 patent because it was filed before the effective filing date of the ’397 patent.

46. Bernardo describes “[a] tool for creating and modifying a Web site that eliminates the need for a user to write or edit Hypertext Markup Language (HTML) code.” (Bernardo, 1:25-27.) Bernardo’s website creation tool is a “software program for use on a computer network,” and “a network browser may be used to access the tool.” (Bernardo, 6:56-59.) Bernardo’s tool presents, through the user’s browser, “one or more views” which “preferably presents a screen containing a menu of features and/or options for creating and customizing the Web site.” (Bernardo, 7:7-10.) Figure 8, reproduced and highlighted in yellow below, illustrates one such “view” that “provides a screen from which the user is prompted to decide on the options for the design of the web page, for example, by selecting colors, background images, frame styles, and logos.” (Bernardo, 8:56-60.)

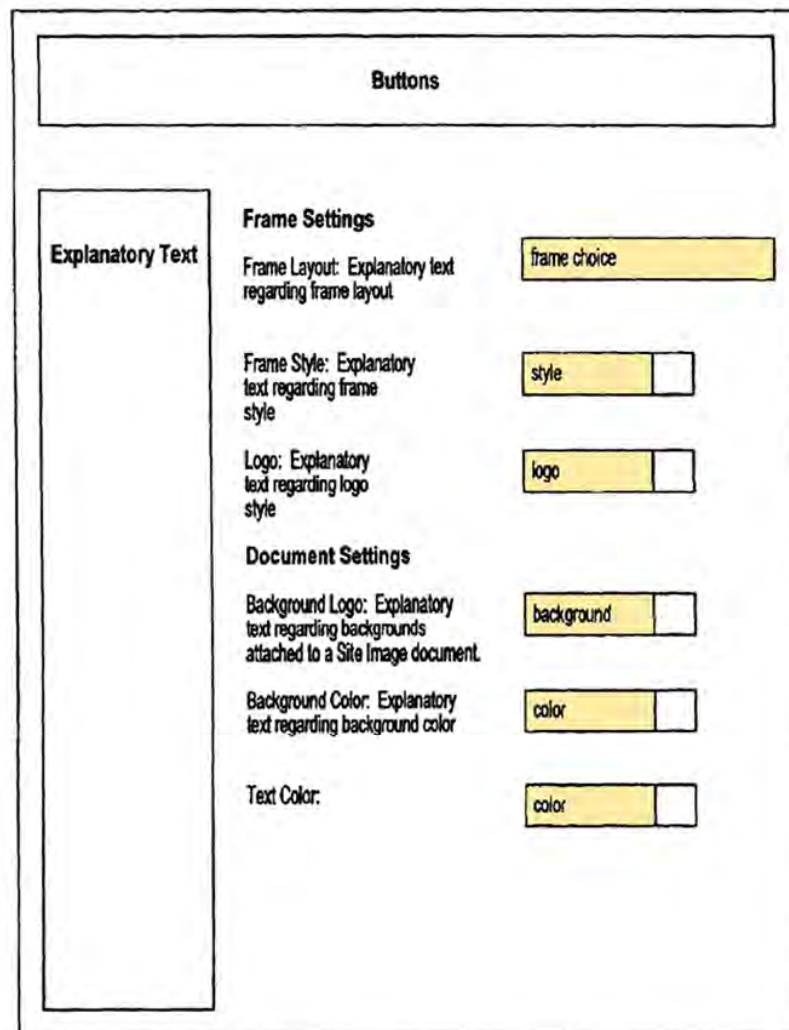


FIG. 8

(Bernardo, Fig. 8 (highlighting added).)

47. Bernardo also discloses a “preview function” that renders pages “on-the-fly” to display previews of the options selected as they are made by the user:

Some embodiments of the invention may include a preview function. A preview function may enable the user to view the Web page as it is created. For example as various options are selected, a preview of the

Declaration of Vijay K. Madiseti, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 6,546,397 B1

option(s) selected may be displayed for a user to observe. This ability is provided by the tool based on the templates without requiring the user to close the file and reopen it with a browser as would be the case if the user were writing HTML code. Thus, the tool enables pages/objects to be rendered on-the-fly.

(Bernardo, 10:5-13.)

3. Meyer [Ex. 1005]

48. **Meyer**, entitled *Java Virtual Machine* (1997), is a reference book describing aspects of the Java programming framework. I am informed that Mayer qualifies as prior art to the '397 patent because it was published at least one year before the earliest effective filing date for the '397 patent.

49. "Java" refers to a well-known computer programming framework developed originally by Sun Microsystems. (Meyer, p.2.) My Declaration cites Meyer with respect to certain claim limitations reciting a "**virtual machine.**" Meyer discloses such a virtual machine—the Java Virtual Machine—which is used to execute web-based applications known as applets:

The Java Virtual Machine is at the core of the Java programming language. In fact, you can't run a Java program without also running an implementation of the Java Virtual Machine. The Java Virtual Machine is the name of the engine that actually executes a Java program and is the key to many of Java's features, including its portability, efficiency, and security.

Declaration of Vijay K. Madiseti, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 6,546,397 B1

(Meyer, p.5.) Meyer further explains that Java functionality can also be embedded in Web browsers and used to generate displays for the user:

With a Java-enabled Web browser, Web content providers are suddenly able to embed *applets* as part of their online offerings. Applets are mini-applications written in the Java language that are downloaded over the Internet on the fly, and executed on the user's local computer. Applets can be used to create interactive data displays, animations, multimedia games, or database interfaces.

(Meyer, p.4 (italics in original); *see also id.*, p.5 (“And underlying each running Java applet, there is a Java virtual machine that is doing all the running.”).)² Meyer further explains that “applets are Java programs which can be embedded in an HTML page and then viewed within a Web browser” (Meyer, p.36.)

50. Meyer confirms that the claimed recitation of a “**virtual machine**” in claim 1 is nothing more than a reference to existing Java Virtual Machine (JVM) technologies. In fact, the **only** disclosure of “virtual machine” in the entire '397 patent specification is a single reference to the “JAVA Virtual Machine” ('397, 35:34-37). Meyer thus renders obvious the claimed virtual machine.

² Unless otherwise noted, all underlining and emphasis in quotations from the prior art was added for emphasis.

4. Coombs [Ex. 1006]

51. **Coombs**, entitled *Setting up an Internet Site for Dummies* (3d ed. 1998), is a reference book that describes how to set up Internet sites such as web sites. I am informed that Coombs qualifies as prior art to the '397 patent because it was published at least one year before the earliest effective filing date for the '397 patent and also published before the invention described in the '397 patent.

52. As Coombs explains, the “first steps that everyone needs to take in setting up an Internet site,” such as a web site, are “getting a domain name and establishing a presence on the Internet.” (Coombs, p.3.) A domain name enables Internet users to access the server hosting the Internet site without “worry[ing] about translating computer names into IP addresses.” (Coombs, p.25.) Coombs further explains that obtaining a domain name for a web site involves confirming that the domain name was not taken, submitting an application with the appropriate registrar, and paying the appropriate fee. (Coombs, pp. 29-39.)

5. Witkowski [Ex. 1007]

53. U.S. Patent No. 6,125,360 to Andrew Witkowski et al., relates to database technologies. My Declaration cites Witkowski solely for its brief description of relational databases that I cite in my discussion of claim 1[c] to account for the longer definition of “database” identified in **Part V**, which I understand was adopted in a prior litigation involving the '397 patent. I am informed

Declaration of Vijay K. Madiseti, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 6,546,397 B1

that Witkowski qualifies as prior art to the '397 patent because it issued from an application filed before the effective priority date of the patent.

54. Witkowski confirms that the longer definition of “database” identified in **Part V** above (*i.e.* “an electronic information storage system offering data storage and retrieval and that stores information on a record-by-record basis, each record divided into one or more fields”) is simply a description of database (and relational database) technologies that were well-known to persons of ordinary skill in the art. The “Background of the Invention” in Witkowski provides the following brief summary of prior art database management systems:

In a database management system (DBMS), data is stored in one or more data containers, each container contains records, and the data within each record is organized into one or more fields. In relational database systems, the data containers are referred to as tables, the records are referred to as rows, and the fields are referred to as columns.

(Witkowski, 1:20-25.) Witkowski thus discloses and renders obvious a “database” under even the longer and more detailed definition shown in **Part V**. As explained below, it would have been obvious to implement the database in Reynolds as a relational database that would meet any of the definitions set forth above.

B. Ground 1: Claim 1 is obvious over Reynolds, Bernardo, Meyer and Witkowski

1. Independent Claim 1

55. I have reproduced independent claim 1 below, and divided up the limitations using bracketed notations (e.g., “[a],” “[b],” etc.) to facilitate identification of the limitations in my analysis below:

1. A method to allow users to produce Internet websites on and for computers having a browser and a virtual machine capable of generating displays, said method comprising:

[a][1] presenting a viewable menu having a user selectable panel of settings describing elements on a website, said panel of settings being presented through a browser on a computer adapted to accept one or more of said selectable settings in said panel as inputs therefrom,

[a][2] and where at least one of said user selectable settings in said panel corresponds to commands to said virtual machine;

[b] generating a display in accordance with one or more user selected settings substantially contemporaneously with the selection thereof;

[c] storing information representative of said one or more user selected settings in a database;

[d] generating a website at least in part by retrieving said information

representative of said one or more user selected settings stored in said database; and

[e] building one or more web pages to generate said website from at least a portion of said database and at least one run time file, where said at least one run time file utilizes information stored in said database to generate virtual machine commands for the display of at least a portion of said one or more web pages.

(’397, 65:44-66:2 (Claim 1).) As explained in detail below, the prior art discloses and renders obvious each limitation of claim 1.

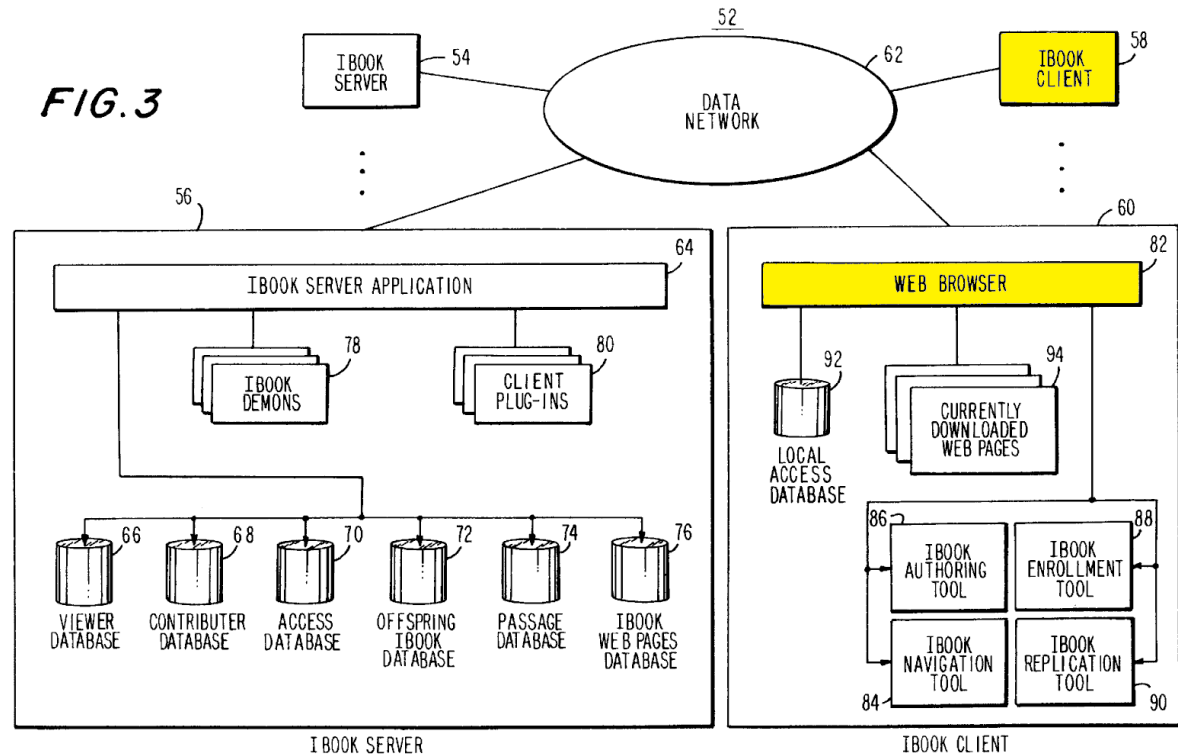
(a) “A method to allow users to produce Internet websites on and for computers having a browser and a virtual machine capable of generating displays, said method comprising:” (Claim 1[preamble])

56. Reynolds discloses “[a] **method to allow users to produce Internet websites on and for computers having a browser,**” as claimed. Reynolds refers to user-produced **websites** as interactive Web books or “ibooks” accessible via the World Wide Web. (Reynolds, Abstract; *see also id.*, 1:5-8 (“This invention relates to the Internet, and more particularly, to techniques for creating and viewing material on the World Wide Web in the form of an interactive Web book.”).) Reynolds explains that each “Web book,” or “ibook,” is composed of a collection of web pages. (Reynolds, 3:46-47 (“Contributed material is preferably stored in the form

of ibook web pages 42”), 7:35-36 (“The material contributed to an ibook is contained in one or more Web pages”).).

57. Reynolds discloses a number of tools for allowing users to produce ibooks, including ibook authoring tool **86**. (Reynolds, 7:2-3 (“To create or modify material in an ibook, a contributor uses ibook authoring tool **86** (FIG. 3).”). Reynolds thus discloses “[a] **method to allow users to produce Internet websites,**” as recited in the preamble. I will provide more detail about the tools in Reynolds for producing ibooks in my discussion of the claim limitations below.

58. Reynolds further discloses that ibooks qualify as websites “**on and for computers having a browser.**” The “**computers**” take the form of ibook client computers, which run software including a “**browser,**” as shown in Figure 3:

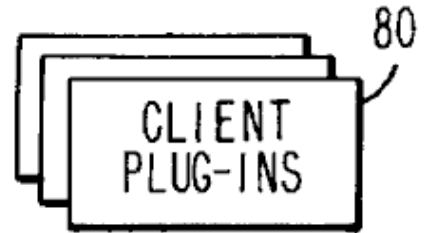


(Reynolds, Fig. 3 (highlighting added).) Figure 3 shows two ibook clients **58** and **60**, but shows a more expanded view of ibook client **60**. “Ibook clients **58** and **60** may be personal computers.” (Reynolds, 5:13-14.) The ibook clients also include Web browser **82**, which “is preferably a standard Web browser, such as Microsoft Internet Explorer or Netscape Navigator.” (Reynolds, 5:31-33.) The ibook clients also store “[c]urrently downloaded Web pages **94**,” which were “downloaded from ibook Web pages database **76**” at the ibook server. (Reynolds, 5:41-42.)

59. Reynolds, alone and in combination with Meyer, discloses and renders obvious “a browser and a virtual machine capable of generating displays,” as claimed. With respect to the claimed “**browser**,” Web browser **82** in Reynolds is

capable of **generating displays**, such as an ibook displayed to a viewing user. (Reynolds, *e.g.*, 1:10-11 (“One can search the Web and view a large amount of material using a Web browser.”), 3:40-43 (“‘Viewing’ (by a contributor or by a viewer) includes reading, listening, and observing any type of contributed material that is contained in an ibook and its behavior.”).) It would have been apparent and obvious to an ordinarily skilled artisan that Web browser **82** would have been capable of generating displays in connection with viewing webpages on the web.

60. The displays provided through Web browser **82** are facilitated in part through client plug-ins **80**, which are software modules downloaded by the ibook client from an ibook server. “Client plug-ins **80** are downloadable executable program modules that can be run on client **60** in conjunction with Web browser **82** to add additional capabilities to Web browser **82**.” (Reynolds, 5:28-31.) The ability to download and execute “plug-ins” was a standard capability of Microsoft and Netscape web browsers widely used by the late 1990s. (Reynolds, 5:31-33.) Reynolds explains that “[w]hen client plug-ins **80** are downloaded onto client **60**, they form tools **84**, **86**, **88**, and **90**.” (Reynolds, 5:35-37.) One such tool—navigation tool **84**—allows a viewer to request access to content of an ibook, which Reynolds refers to as a “passage”:



Navigation tool 84 preferably generates a request to view a desired passage and transmits this request to server 56 at step 200 [Figure 9]. Server application 64 at server 56 receives the request and locates the requested passage in ibook Web pages database 76 at step 202 . . . Server 56 transmits the requested passage to client 60 at step 206. Client 60 displays the passage and may make a time and date entry in local access database 92 at step 208.

(Reynolds, 11:31-40; *see also id.*, 1:30-38 (“Material in an ibook is preferably organized in the form of Web pages . . . Within each Web page, material may be organized in passages. Passages can be of any suitable size, such as a paragraph for textual material, or a video or audio clip of a certain length for multimedia material.”).) Another client plug-in, authoring tool 86, allows a user to create or modify material in an ibook. (Reynolds, *e.g.*, 7:2-3.) Because client plug-ins in Reynolds execute under Web browser 82 to facilitate generation of displays, Reynolds discloses “**a browser . . . capable of generating displays.**”

61. With respect to the claimed “**virtual machine**,” although Reynolds does not specifically expressly disclose a “**virtual machine**,” Reynolds does describe using a “Java program” at the client in connection with presenting content. (Reynolds, 9:54-65 (“Style buttons 180 allow a user to select a desired format for the presentation of the information on client 60. Each style is controlled by an ibook passage (e.g., a Java program) which implements that style.”).) A person of ordinary

skill in the art would have understood that a virtual machine—in particular, the Java Virtual Machine (JVM)—would be required at the client in order to run the Java program. “Java” refers to a well-known computer programming framework developed originally by Sun Microsystems, (Meyer, p.2), and the JVM is software that runs on the client used to run Java programs. These and other details of Java and the JVM are disclosed by Meyer, which in combination with Reynolds renders the claimed “**virtual machine**” obvious. As Meyer explains:

The Java Virtual Machine is at the core of the Java programming language. In fact, you can’t run a Java program without also running an implementation of the Java Virtual Machine. The Java Virtual Machine is the name of the engine that actually executes a Java program and is the key to many of Java’s features, including its portability, efficiency, and security.

(Meyer, p.5.) Meyer further explains that Java functionality can also be embedded in Web browsers and used to generate displays for the user:

With a Java-enabled Web browser, Web content providers are suddenly able to embed *applets* as part of their online offerings. Applets are mini-applications written in the Java language that are downloaded over the Internet on the fly, and executed on the user’s local computer. Applets can be used to create interactive data displays, animations, multimedia games, or database interfaces.

(Meyer, p.4 (italics in original); *see also id.*, p.5 (“And underlying each running Java

applet, there is a Java virtual machine that is doing all the running.”).) Meyer further explains that “applets are Java programs which can be embedded in an HTML page and then viewed within a Web browser” (Meyer, p.36.) The combination of Reynolds and Meyer thus discloses “**a browser and a virtual machine capable of generating displays.**” As I explain in detail below in connection with my analysis for claims 1[a][2] and 1[d]/1[e], it would have been obvious to an ordinarily skilled artisan to implement the ibook authoring tool **86** and navigation tool **84** of Reynolds as Java applets, which would have in turn required the Java Virtual Machine in order to generate their associated displays at the client.

62. I also note that mapping the “virtual machine” to the Java Virtual Machine (JVM) is consistent with the ’397 patent specification—in fact, the **only** disclosure of “virtual machine” in the entire ’397 patent specification is a single reference to the “JAVA Virtual Machine” (’397, 35:34-37). The Java Virtual Machine (JVM) also satisfies each of the proposed definitions of “**virtual machine**” outlined in **Part V** above. Meyer describes the Java Virtual Machine (JVM) as an abstract machine, emulated in software, which executes intermediate code (*i.e.* Java programming instructions). (Meyer, pp.5-6.) Meyer explains that the Java Virtual Machine provides “a special piece of software [that] steps through each of the instructions in the program and carries out the action that the instruction represents.”

Declaration of Vijay K. Madiseti, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 6,546,397 B1

(Meyer, p.5.) Java instructions qualify as “intermediate code” because they are not executed directly by the computer’s processor, but by the software embedded in the virtual machine. “In essence, Java programs are not run directly by the central processor (or CPU) in your system, but are instead run by a ‘virtual’ processor, which is itself a piece of software running on your computer.” (*Id.*)

63. ***Manner, Rationale, and Motivation to Combine:*** It would have been obvious to a person of ordinary skill in the art to combine the disclosures of Reynolds with Meyer identified above. The combination would have predictably resulted in the ibook system of Reynolds in which the ibook client computer included Web browser **82** and Java Virtual Machine software (“**a virtual machine**”), which would have been capable of generating displays for the user. This would have been a straightforward combination, and an ordinarily skilled artisan would have had ample motivation to have made it.

64. At the outset, Reynolds and Meyer are analogous references. The ’397 identifies its “Field of the Invention” as being directed to “methods and apparatus for building a web site using a browser-based build engine.” (’397, 1:5-8.) Reynolds similarly describes techniques for enabling users to create and view web sites, for example, using browser-based authoring tool **86**. (Reynolds, *e.g.*, 1:5-8, 7:2-4.) Reynolds thus falls within the same field of endeavor as the ’397 patent, and is

Declaration of Vijay K. Madiseti, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 6,546,397 B1

reasonably pertinent to a problem faced by the inventors of how to implement a web site building system. With respect to Meyer, it is reasonably pertinent to a problem faced by the '397 inventors in how to use a “virtual machine” to generate displays in connection with a web site building system. Notably, the **only** disclosure of “virtual machine” in the entire '397 patent specification is a single reference to the “JAVA Virtual Machine” ('397, 35:34-37), which is further described in Meyer. Meyer's disclosures would thus have been pertinent to a plethora of issues faced by the inventors in implementing the alleged invention using Java and the Java Virtual Machine. For example, Meyer provides detailed disclosures of how to embed Java programs (applets) into HTML web pages that can be viewed by a standard web browser. (Meyer, pp.36-42.)

65. An ordinarily skilled artisan would have had numerous motivations to combine Reynolds and Meyer in this manner. At the outset, Reynolds itself explains that ibooks can contain Java programs such as Java applets. (Reynolds, 9:52-58 (“When passages **152** are executable modules, navigation tool **84** presents the passages by downloading and executing the passages. Style buttons **180** allow a user to select a desired format for the presentation of information on client **60**. Each style is controlled by an ibook passage (e.g., a Java program) which implements that style.”), 4:48-50 (explaining that ibooks can “contain[] a Java script for an applet”).)

Declaration of Vijay K. Madiseti, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 6,546,397 B1

It thus would have been obvious and natural for an ordinarily skilled artisan to consult the teachings of Meyer when implementing the ibook system described in Reynolds.

66. Meyer itself discloses numerous express motivations to employ Java technologies in a web-based system. As Meyer explains, “Java offers many tantalizing benefits to the application developer, but probably the greatest strength of Java lies not in any single design feature, but in the elegance of the overall package that the Java designers have put together.” (Meyer, p.2.) Meyer goes on to identify and describe several distinct advantages of Java including the ability to run the same Java program on a wide range of computers (portability), “from small personal computers up to million-dollar graphics supercomputers.” (*Id.*) Meyer also touts the execution speed and efficiency of Java programs, the benefits of the dynamic features of the Java programming language, as well as built-in safety and security features “designed to make consumers feel comfortable running programs that have been downloaded over the Internet.” (Meyer, pp.2-3.) Meyer also explains that using Java offered the distinct advantage of “making use of the computing resources available on the user’s local computer,” which reduces the workload requirements of remote servers that were often overloaded or subject to slow connections with clients. (Meyer, p.4.)

67. The benefits of Java recognized in Meyer would have provided a direct advantage to any interactive web-based system, including the web site creation and viewing system in Reynolds. With respect to the claimed “**virtual machine**,” as Meyer explains “you can’t run a Java program without also running an implementation of the Java Virtual Machine.” (Meyer, p.5.) The act of implementing the web site creation system of Reynolds using Java—to achieve any of the benefits above—would have necessarily included “**a browser and a virtual machine capable of generating displays**,” as recited in the preamble.

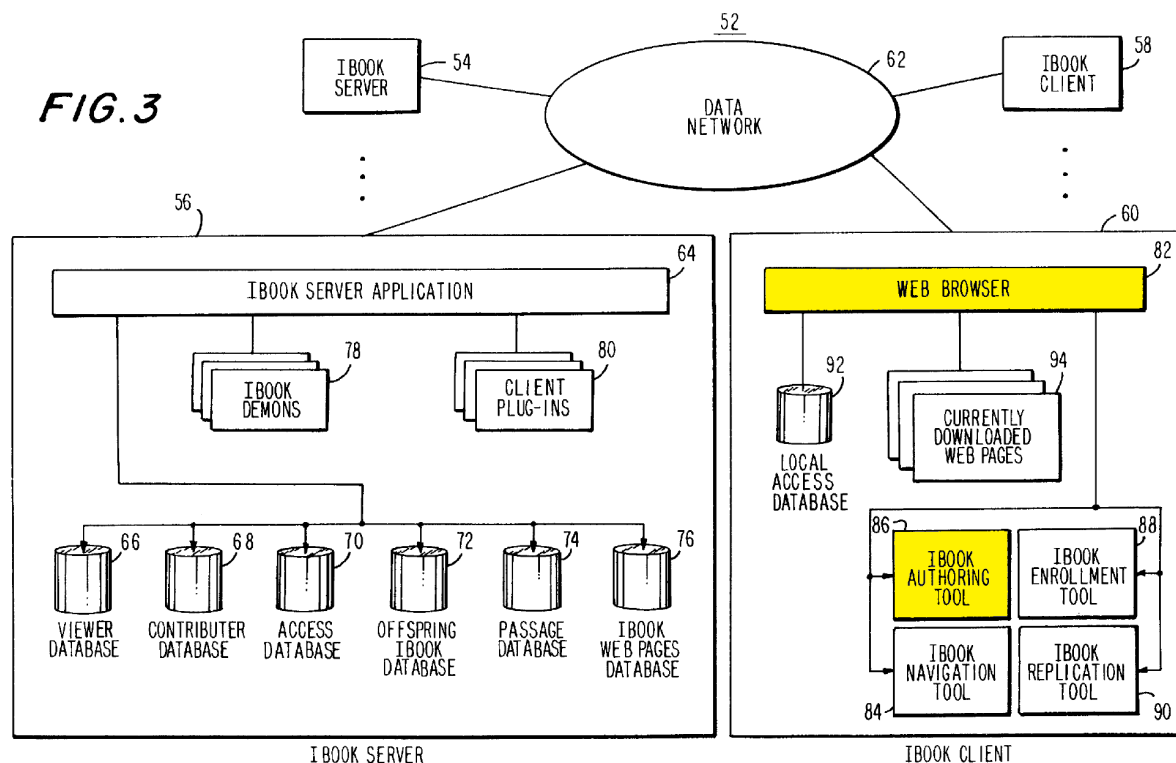
68. An ordinarily skilled artisan would have had more than a reasonable expectation of success and would have perceived no technological obstacles to the proposed combination. Java was a well-known technology by December 1999, and as books like Meyer confirm, was mature and well-documented. Meyer explains that with Java, “[i]t’s now possible to create and deliver dynamic applications (i.e., applets) that run securely, efficiently, and on a wide range of platforms.” (Meyer, p.4.) An ordinarily skilled artisan would have found the disclosures in Reynolds and Meyer more than sufficient to enable a person of ordinary skill in the art to implement their teachings.

- (b) “presenting a viewable menu having a user selectable panel of settings describing elements on a website, said panel of settings being presented through a browser on a computer adapted to accept one or more of said selectable settings in said panel as inputs therefrom, and where at least one of said user selectable settings in said panel corresponds to commands to said virtual machine;” (Claim 1[a])

69. Because of the length of this limitation, I have divided it into two parts, claim 1[a][1] and 1[a][2], below.

- (1) “presenting a viewable menu having a user selectable panel of settings describing elements on a website, said panel of settings being presented through a browser on a computer adapted to accept one or more of said selectable settings in said panel as inputs therefrom...” (Claim 1[a][1])

70. Reynolds in combination with Bernardo discloses and renders obvious claim 1[a][1]. As I noted for the preamble and shown in Figure 3 below, Reynolds discloses an authoring tool **86** that runs within Web browser **82** on ibook client **60**:



(Reynolds, Fig. 3 (highlighting added).) “To create or modify material in an iBook, a contributor uses iBook authoring tool **86** (FIG.3).” (Reynolds, 7:2-3.) The authoring tool **86** in turn includes a number of tools including editing tool **136**:

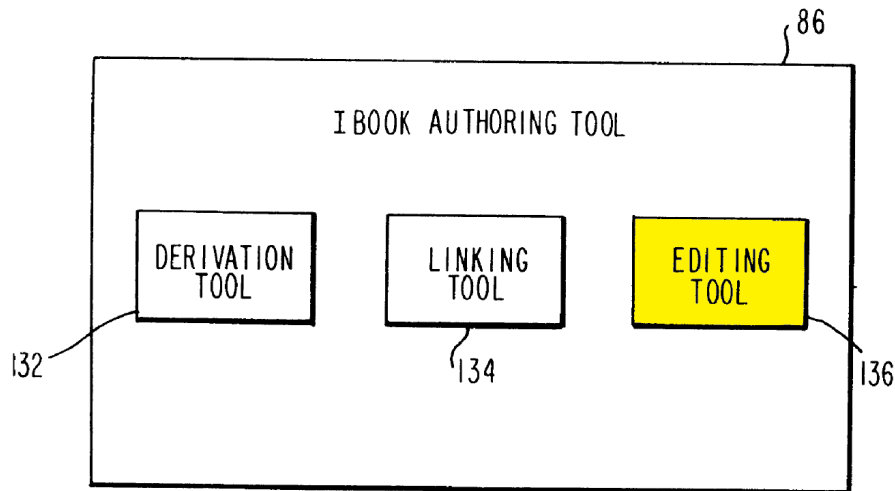


FIG. 6

(Reynolds, Fig. 6 (highlighting added).) “As shown in FIG. 6, ibook authoring tool **86** includes derivation tool **132**, linking tool **134**, and editing tool **136**.” (Reynolds, 7:3-5.) Editing tool **136**, shown above in yellow highlighting above, “is used to enter original material” for an ibook. (Reynolds, 7:5-7.)

71. Reynolds states that editing tool **136** “preferably supports standard word processing functions for entering text and source code and supports graphics, animation, audio, and video editing functions for creating and modifying multimedia material.” (Reynolds, 7:24-28.) But Reynolds does not disclose specific details about the user interface by which the functions of authoring tool **86** or editing tool **136** are provided. Reynolds thus does not expressly disclose “**presenting a viewable menu having a user selectable panel of settings describing elements on**

a website.” But this would have been obvious based on a combination with **Bernardo**. As explained below, under the proposed combination, it would have been obvious to adapt authoring tool **86** of Reynolds to implement the viewable menu claimed.

72. As noted in Part **VI.A** above, like Reynolds, **Bernardo** describes a tool for allowing users to create and edit web pages and web sites. (Bernardo, 1:23-27 (“Field of the Invention [] A tool for creating and modifying a Web site that eliminates the need for a user to write or edit Hypertext Markup Language (HTML) code.”).) Bernardo discloses a website creation software tool that “enables a Web site creator to select from among a plurality of features and options to include within a Web site.” (Bernardo, 6:32-34.) The tool “may be accessed through a suitable network interface.” (Bernardo, 6:57-58.) “For example, a network browser may be used to access the tool.” (Bernardo, 6:58-59; *see also id.*, 6:64-66 (“A site creator may start the process, for example, by directing a network browser to access a server, or other network node, on which the tool is loaded.”).)

73. For purposes of my Declaration, I have relied on the portions of Bernardo that describe user interface aspects of the website creation process, and in particular, the “viewable menus” by which the system in Bernardo collects user input from the website creator. Under the proposed combination with Reynolds, as

explained in detail below, these user interface features from Bernardo would be incorporated into authoring tool **86** (which includes editing tool **136**) of Reynolds.

Bernardo explains that, as part of the website creation process:

At Step **8** [of Figure 3] a site creator may select the desired features/options. Selection may be accomplished in any suitable manner.

(Bernardo, 7:37-39.) Bernardo further explains that selection of website options by the site creator can be accomplished using viewable menus:

To facilitate modifying the site the content composer may be presented various views by the tool to add or modify pages. The view may be any type of graphical or text arrangement that provides a user with several options. For example, view may include a drop down or pop up menu, a list of options to be selected using a radio button selector, a text entry box, or other suitable graphical selection interface . . . For example, options may include choices regarding editing of text, colors, graphics or other objects, as well as, choices regarding positioning of objects, creation of new objects, deleting objects, adding links to other sites, security provisions, and other choices.

(Bernardo, 9:48-61.) The passage above discloses “**presenting a viewable menu having a user selectable panel of settings describing elements on a website.**”

Bernardo specifically mentions a “drop down or pop up menu,” or a “list of options to be selected using a radio button selector,” any one of which qualifies as a

“**viewable menu**” as claimed.

74. Bernardo also provides several concrete examples of screen displays showing viewable menus that satisfy the claim limitation. Figure 5 discloses a first example of the claimed **viewable menu**:

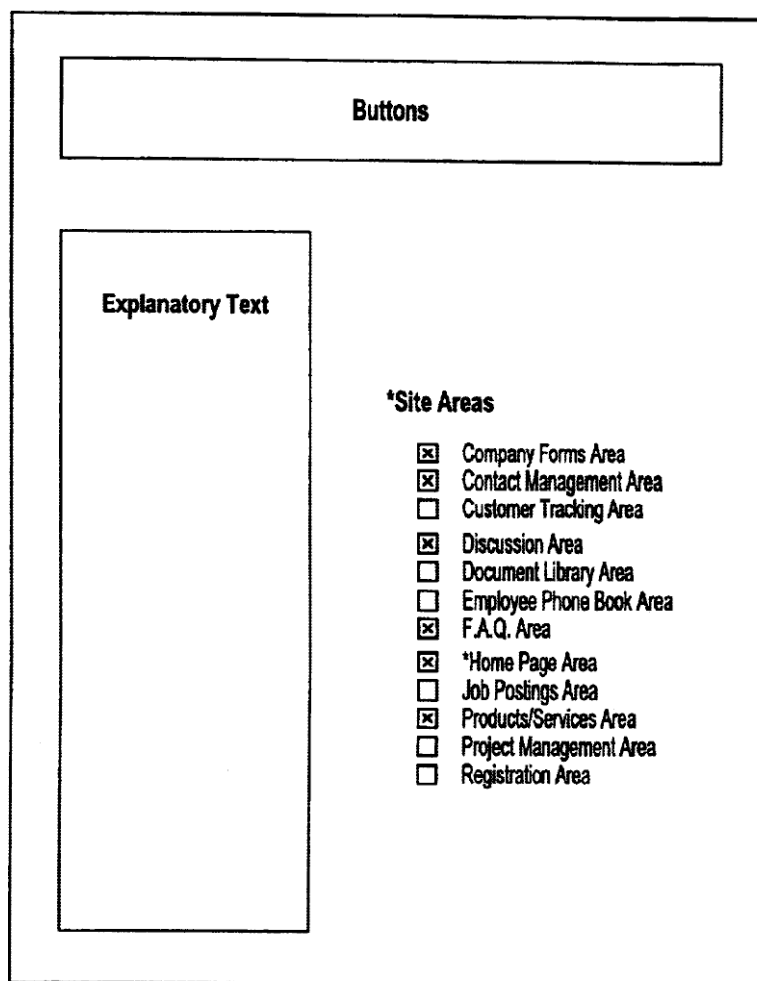


FIG. 5

(Bernardo, Fig. 5.) The screen in Figure 5 discloses a “viewable menu” because it presents a number of user-selectable options for the user. Indeed, Bernardo itself

Declaration of Vijay K. Madiseti, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 6,546,397 B1

describes the screen display in Figure 5 as “a site area selection menu.” (Bernardo, 4:38-39.) Bernardo explains that if the user selects to create a new website, Figure 5 “shows the types of site area options which a user can choose to have included in a Web site.” (Bernardo, 8:38-42.) “As shown in FIG 5, desired site areas may be selected by entering an ‘X’ in a check box.” (Bernardo, 7:42-44.)

75. Figure 8 of Bernardo discloses a second exemplary screen display that also discloses the claimed “**viewable menu**”:

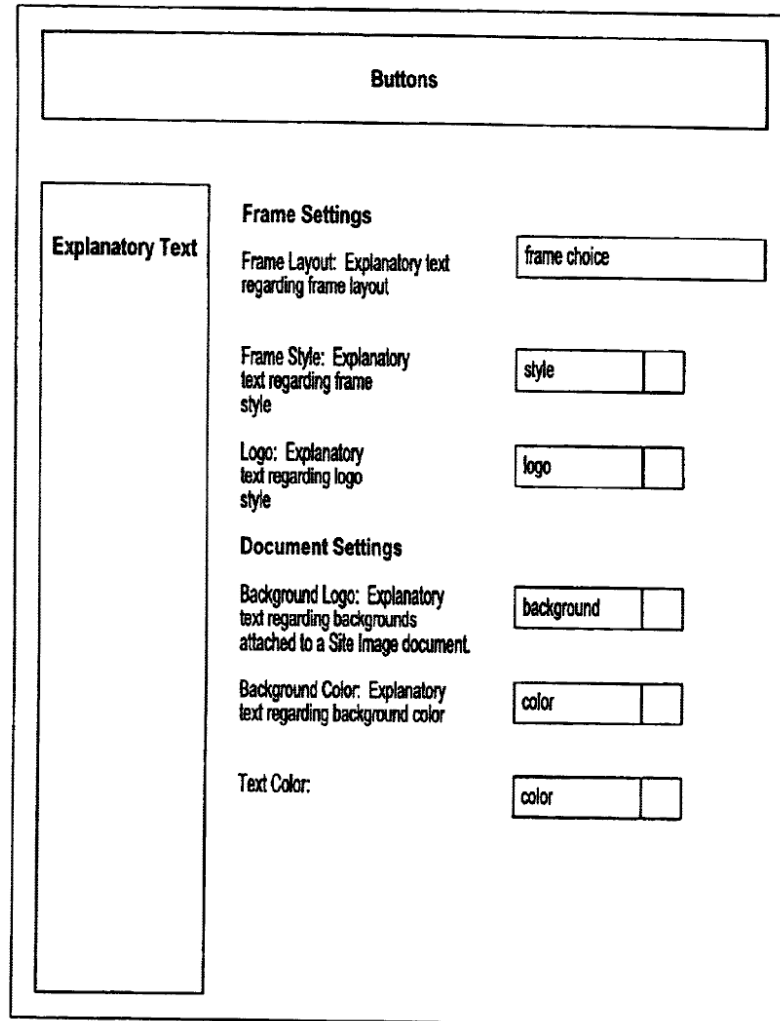


FIG. 8

(Bernardo, Fig. 8.) Figure 8 provides another “illustration depicting a screen showing a selection **menu**.” (Bernardo, 4:45-46.) Bernardo describes Figure 8 as providing a “design center feature [that] provides a screen from which the user is prompted to decide on the options for the design of the web page, for example, by selecting colors, background images, frame styles, and logos.” (Bernardo, 8:56-60.) “Once selections have been made, the user can preview the design and then make

changes as desired.” (Bernardo, 8:60-62.) An ordinarily skilled artisan would have recognized the user interface controls in Figure 8 above as including drop down lists which, as noted previously, qualify as viewable menus.

76. Bernardo provides other examples of screen displays that provide viewable menus. Figure 10 provides another example:

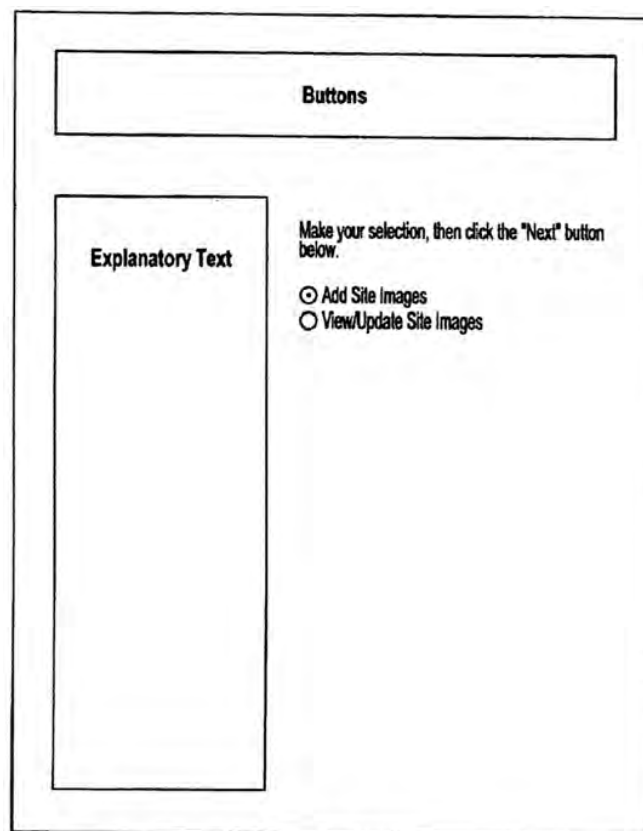


FIG. 10

(Bernardo, Fig. 10.) Figure 10 shows another “selection menu” (Bernardo, 4:49-50), with two menu options to either add new site images or view/update existing site images. “As illustrated the content composer selected the ‘add site images’”

option. (Bernardo, 9:62-64; *see also* Bernardo, 8:67-9:12 & Fig. 9 (showing a further example of a menu for allowing a website creator to select whether to update individual website areas or all website areas).)

77. Bernardo provides further examples of viewable menus. For example, Bernardo explains that in adding content to the main page of an area of the website, one of the steps performed by the user is to “[s]elect a logo or banner for the page from the drop-down list.” (Bernardo, 22:23-30.) Additionally, the user can specify a category of a content page (for indexing purposes) using a drop down list. (Bernardo, 22:56-61.) Although Bernardo does not show actual screen displays of these features, as noted, a “drop-down list” would have been understood by an ordinarily skilled artisan as a “viewable menu” allowing a user to select from a list.

78. Each of the above-mentioned features from Bernardo discloses a separate example of “**presenting a viewable menu having a user selectable panel of settings describing elements on a website,**” as claimed. Each example presents a screen display that provides attributes of an object available for selection by the user. For example, Figure 5 presents a viewable menu with a panel of attributes for the website, in this case, identifying specific areas that the user may wish to add to the website being created. (Bernardo, 8:38-42, 7:42-44, Fig. 5.) Figure 8 presents a viewable menu showing specific attributes for web page objects including colors,

images, styles and logos. (Bernardo, 8:56-60, Fig. 8.) Figure 10 presents a viewable menu showing the ability to specify a particular type of object for addition to the website, *i.e.* a new image. (Bernardo, 9:61-64, Fig. 10.)

79. But even without relying on these specific examples, as noted above, Bernardo plainly discloses that the user “may be presented various views by the tool to add or modify pages,” which “may include a drop down or pop up menu” to “present [the] user with various choices regarding the creation of a Web site,” including “editing of text, colors, graphics or other objects, as well as, choices regarding positioning of objects, creation of new objects, deleting objects, adding links to other sites, security provisions, and other choices.” (Bernardo, 9:48-64; *see also id.*, 7:6-10 (“a site creator is presented with one or more views. The view(s) preferably presents a screen containing a menu of features and/or options for creating and customizing the Web site.”).)

80. Bernardo further discloses that, for each of these screen displays, “**said panel of settings [is] presented through a browser on a computer adapted to accept one or more of said selectable settings in said panel as inputs therefrom.**” As explained above, Bernardo discloses that the website creator accesses the website creation tool by accessing it using a browser. (Bernardo, *e.g.*, 6:58-7:6.) Reynolds similarly discloses, as explained previously, that authoring tool **86** (which includes

editing tool **136**) is downloaded by and executed in Web browser **82**. (Reynolds, 7:2-3, 5:28-37.) Bernardo, alone and in combination with Reynolds, thus discloses that the panel of settings is presented “**through a browser on a computer,**” and because the browser receives the user’s selections through the viewable menus in Bernardo, the browser is “**adapted to accept one or more of said selectable settings in said panel as inputs therefrom.**”

81. ***Manner, Rationale, and Motivation to Combine:*** It would have been obvious to a person of ordinary skill in the art to combine the disclosures of Reynolds with Bernardo. The combination would have predictably resulted in the ibook system of Reynolds in which the Web browser **82**, when executing editing tool **136** (which is part of ibook authoring tool **86**), presents a viewable menu having a user selectable panel of settings describing elements on a website, as disclosed in Bernardo. Likewise, when executing editing tool **136**, Reynolds’ Web browser **82** would be adapted to accept the user’s selections from the panel of settings as inputs. For example, under the proposed combination, editing tool **136** of Reynolds could have been adapted to include menus as disclosed in Bernardo enabling the user to select areas for the website, specify design options for web pages, add images, or make other selections of attributes for web page objects. As additional examples, editing tool 136 of Reynolds could have been adapted to include a “drop down or

pop up menu” enabling the user to make “choices regarding editing of text, colors, graphics or other objects, as well as, choices regarding positioning of objects, creation of new objects, deleting objects, adding links to other sites, security provisions, and other choices.” (Bernardo, 9:48-60.)

82. Bernardo is an analogous reference to the '397 patent for the same reason as Reynolds discussed above. As noted, the '397 identifies its “Field of the Invention” as being directed to “methods and apparatus for building a web site using a browser-based build engine.” ('397, 1:6-8.) Bernardo describes the same type of system, *i.e.*, “[a] tool for creating and modifying a Web site that eliminates the need for a user to write or edit Hypertext Markup Language (HTML) code.” (Bernardo, 1:25-27.) Bernardo thus falls within the same field of endeavor as the '397 patent, and would have been reasonably pertinent to a problem faced by the inventors of the features and capabilities of implementing a web site building system. For example, Bernardo provides specific teachings about user interface techniques for a website creation program that would have been reasonably pertinent to addressing similar user interface implementation challenges facing the '397 patent inventor.

83. An ordinarily skilled artisan would have had numerous motivations to combine Reynolds with Bernardo in this manner. Reynolds provides a suggestion for such a combination by explaining that editing tool **136** “preferably supports

standard word processing functions” for entering text and other information. (Reynolds, 7:24-28.) A person of ordinary skill in the art would have recognized the user interface techniques in Bernardo as resembling “word processing functions” known in the art, such as specifying design and display options. An ordinarily skilled artisan would have found it natural to consult Bernardo in decisions about user interface implementation of the Reynolds editing tool **136**.

84. A person of ordinary skill in the art would also have recognized that a viewable menu with a user-selectable panel of settings, as disclosed through the menus of Bernardo, would have improved the user experience of Reynolds by allowing the user greater control over the appearance of web pages. For example, Figure 8 of Bernardo (as discussed above) allows the user to customize design options for the webpage, “for example, by selecting colors, background images, frame styles, and logos.” (Bernardo, 8:57-62.) These would have provided a benefit to Reynolds of enhancing the user experience—for both the ibook contributor and viewer—by allowing greater customization of the appearance of content displayed in ibook web pages by the browser.

85. As another example, Figure 5 of Bernardo (as discussed above) allows the user to select a number of “predefined site areas” for inclusion on the website. (Bernardo, 7:27-36.) This would have provided a benefit to the system of Reynolds

by allowing an ibook creator to more rapidly and easily create ibook pages based on predefined templates. And although the exemplary site areas listed in Figure 5 of Bernardo resemble portions of a corporate website (*e.g.* “Company Forms Area”), Bernardo makes clear those areas are merely exemplary. (Bernardo, 7:35-37 (“Other areas may be created based on templates for these areas, without requiring the Web site creator to write or edit software code.”).) It would have been obvious that an ordinarily skilled artisan could have adapted the site selection viewable menu of Bernardo to provide predefined site areas appropriate to an ibook implementation.

86. Additionally, as explained, Figure 10 presents a viewable menu allowing the user to either add new website images, or view/update existing images. (Bernardo, 9:61-64, Fig. 10.) This would have presented a benefit to the system of Reynolds by providing a convenient menu for adding images to website content. Reynolds specifically discloses that editing tool **136** can be used to provide multimedia content for an ibook, such as graphic images. (Reynolds, 7:24-31, 8:5-10.) It therefore would have been apparent that any implementation of editing tool **136** would have required a user interface of some type to enable the user to add images to ibook web pages. The viewable menu in Bernardo would have provided a simple and user-friendly way for the user to do so.

87. And at a more general and less granular level, an ordinarily skilled

artisan would have been motivated to implement selection of attributes for web page objects using **viewable menus**, as such menus had known benefits and were standard fixtures of graphical user interface (GUI) implementations since at least the early 1980s. Viewable menus provide a clear user experience advantage by allowing a user to make a selection by simply choosing from a menu of displayed options—thus allowing the user to quickly make a selection without having to type or key in the selection (for example using a keyboard). This not only increases the user’s efficiency but avoids selection errors that might result from keystroke mistakes.

88. This benefit was well-known to ordinarily skilled artisans. For example, U.S. Patent No. 4,464,652, which issued in 1984 to Apple Computer, Inc., provides an early disclosure of on-screen pull-down menus. (Ex. 1009, Figs. 14-15, 6:16-54.) That patent explains that through those menus, “it is possible for a user to select and execute a variety of commands without the necessity of inputting characters on a keyboard,” as is commonly required in the art.” (Ex. 1009, 6:55-58.) Viewable menus also represented one of a limited number of standard graphical user interface controls widely available by December 1999 for obtaining user input.

89. An ordinarily skilled artisan would have had more than a reasonable expectation of success, and would have perceived no technological obstacles to the proposed combination. As explained, the user interface techniques described in

Bernardo were well-known long before December 1999, and could have been implemented using conventional programming techniques. An ordinarily skilled artisan would have found the disclosures in Reynolds and Bernardo more than sufficient to enable a person of ordinary skill in the art to implement the proposed combination without undue experimentation.

(2) **“... and where at least one of said user selectable settings in said panel corresponds to commands to said virtual machine;” (Claim 1[a][2])**

90. Bernardo does not disclose use of a **“virtual machine”** with respect to the viewable menus discussed above because, put simply, Bernardo does not provide details about the implementation of the software tool or its user interface. But as explained below, it would have been obvious to an ordinarily skilled artisan to implement the viewable menu under the combination of Reynolds and Bernardo as a Java applet. Because Java applets contain commands or instructions to **“said virtual machine,”** *i.e.* the Java Virtual Machine, implementing the viewable menu interface as a Java applet would have resulted in **“at least one of said user selectable settings in said panel correspond[ing] to commands to said virtual machine.”**

91. The claimed **“commands to said virtual machine”** correspond to instructions of the Java applet built based on the combination, which are executed by the Java Virtual Machine (JVM) on the client computer. Meyer explains that

“underlying each running Java applet, there is a Java Virtual Machine that is doing all the running.” (Meyer, p.5.) Meyer further explains:

The Java Virtual Machine is at the core of the Java programming language. In fact, you can’t run a Java program without also running an implementation of the Java Virtual Machine. The Java Virtual Machine is the name of the engine that actually executes a Java program . . . Whenever you run a Java program, the instructions that make up the program are not executed directly by the hardware your system uses. Instead, a special piece of software steps through each of the instructions in the program and carries out the action that the instruction represents.

(*Id.*) The instructions of a Java applet thus represent “**commands to said virtual machine,**” as claimed, because each instruction provides a command directing the Java Virtual Machine to perform a specific operation.

92. The prior art thus renders obvious that “**at least one of said user selectable settings in said panel corresponds to commands to said virtual machine,**” as claimed. This is because, under the proposed combination, a Java applet would carry out the functionality in Reynolds and Bernardo of (1) presenting the claimed “viewable menu” that contains the user selectable settings, and (2) responding to the user selection by providing the functionality specified by the user’s selection from the viewable menu. The instructions within the Java applet for

providing the functionality specified by the user's selection from the viewable menu correspond to commands to the Java Virtual Machine.

93. ***Rationale and Motivation to Combine:*** As explained previously in connection with the preamble, it would have been obvious to combine the teachings of Reynolds with the Java applet teachings of Meyer. As relevant to this claim limitation the combination would have predictably resulted in implementing authoring tool **86** of Reynolds as a Java applet, the applet adapted to include a viewable menu as disclosed in Bernardo.

94. As explained for the preamble, the motivation to combine would have been straightforward as Meyer identifies several compelling advantages of Java including its portability, execution speed and efficiency, built-in safety and security features, and ability to make efficient use of the computing resources available on the user's local computer. (Meyer, pp.2-4; *see also* analysis accompanying claim 1, preamble.) Bernardo also recognizes that "different Internet Web browser software exists and a given user may attempt to interface a Web site with any one of a number of browser types," but "a given user interface (A) may not display Web site content (e.g., frames) in the same manner as a different user interface (B)." (Bernardo, 12:11-26.) Implementation of Reynolds' authoring tool **86**, including a viewable menu, using a platform-independent programming framework such as Java would

enable different web browsers to display the authoring tool and its viewable menu in the same manner, thus reducing complexity by obviating the need to store “various alternatives for browser specific code that is used depending on the identified browser type.” (Bernardo, 12:34-40; *see also* Meyer, pp.8-9 (“You only have to port the virtual machine and associated support libraries to a new architecture once, and then all of the applications built on top of the virtual machine run unchanged.”) Any one of these advantages would have encouraged an ordinarily skilled artisan to use Java to implement the claimed combination, and collectively these benefits provide an overwhelming motivation to combine. And for the reasons explained above, an ordinarily skilled artisan would have had more than a reasonable expectation of success in implementing the system according to the teachings of Reynolds and Bernardo as a Java applet.

95. Meyer provides a further motivation to combine by stating that Java applets “can be used to create interactive data displays” (Meyer, p.4), which is precisely the type of interactive web-based application contemplated in Reynolds and Bernardo. An ordinarily skilled artisan would have understood that a web site building system, such as the system built based on the proposed combination of Reynolds and Bernardo, would have entailed a high degree of interactivity between the user and the system in presenting information and responding to user input. (*See,*

e.g., Reynolds, 7:24-28 (“For example, editing tool **136** preferably supports standard word processing functions for entering text and source code and supports graphics, animation, audio, and video editing functions for creating and modifying multimedia material.”); Bernardo, 8:56-60 (presenting the user with a viewable menu “to decide on the options for the design of the web page, for example, by selecting colors, background images, frame styles, and logos”).) Meyer confirms that Java would have offered an ideal computing framework for implementing such an interactive system.

96. I further note that the proposed combination is entirely compatible with the disclosure of Reynolds, which explains that authoring tool **86** (which contains editing tool **136**) is implemented in the form of a client plug-in. “Client plug-ins **80** are downloadable executable program modules that can be run on client **60** in conjunction with Web browser **82** to add additional capabilities to Web browser **82**.” (Reynolds, 5:28-31.) The proposed combination would thus have entailed nothing more than a simple substitution of a downloadable browser “plug-in” with a downloadable Java applet. Persons of ordinary skill in the art would have considered the client plug-ins **80** of Reynolds and Java applets as interchangeable technologies with respect to providing software functionality to the client browser to extend the capabilities of a web application. Plug-ins and applets, in fact, were commonly

referenced together when identifying ways to extend the functionality of a web browser. (See, e.g., Ex. 1010, 16:64-67, 17:66-18:1 (explaining that in the context of “extending the functionality of Netscape Navigator” to provide an Internet shopping system, the “presentation of an order form via CGI, JAVA script, plug-in code, etc., is apparent to one of ordinary skill in art.”), 17:55-60 (describing a “JAVA applet” as an “alternative[]” to a “plug-in” to enable “Netscape Navigator to retrieve and display [] production information”).) In fact, Reynolds’ definition of client plug-ins **80**, i.e., “downloadable executable program modules that can be run on client **60** in conjunction with Web browser **82** to add additional capabilities to Web browser **82**,” could precisely describe a Java applet and its relationship to the browser.³

³ Reynolds does not define precisely what a “plug-in” is, but Reynolds is presumably referencing a technology known in the late 1990s as “browser plug-ins,” in which a web browser could download, install and locally execute software programs. One key difference between a browser plug-in and a Java applet is that a browser plug-in included program instructions executed natively (directly) on the client’s computer, whereas Java applet instructions execute through an intermediate Java Virtual Machine as I previously explained. The native execution of browser plug-ins would have created at least two significant drawbacks when compared to Java

(c) “generating a display in accordance with one or more user selected settings substantially contemporaneously with the selection thereof;” (Claim 1[b])

97. This limitation is disclosed by Bernardo, which discloses a “preview”

applets. First, while a Java applet could have run on any computer that had a Java Virtual Machine, a browser plug-in was typically platform-specific thus requiring a different version for each computing platform. (See Ex. 1010, 17:16-18 (“A plug-in is a dynamic code module and unlike JAVA modules which are platform independent, plug-ins are platform dependent.”).) Second, plug-in technology typically lacked the security of Java applets; because they could execute code natively on the user’s computer, they could potentially exert greater control of the computer to carry out unwanted operations. (See Ex. 1010, 17:18-28 (describing various “resources [that] should be made available to the plug-in developer,” including “the web browser,” “the transport data link layer,” “the physical layer,” “the operating system,” and “terminal adaptor hardware”), 17:31-33 (“The plug-in can use the Netscape API functions and can also use the Windows API, WinSock API, WinISDN API, Video or Windows, etc.”).) These two drawbacks would have provided further reasons to implement authoring tool **86** of Reynolds as a Java applet instead of a browser plug-in.

Declaration of Vijay K. Madiseti, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 6,546,397 B1

feature of the software tool. (Bernardo, 8:60-62 (“Once selections have been made, the user can preview the design and then make changes as desired.”).) This preview feature also “permits authorized content creators to preview proposed content as it is being created by rendering pages on-the-fly.” (Bernardo, 4:19-21.) As further explained in Bernardo:

Some embodiments of the invention may include a preview function. A preview function may enable the user to view the Web page as it is created. For example as various options are selected, a preview of the option(s) selected may be displayed for a user to observe. This ability is provided by the tool based on the templates without requiring the user to close the file and reopen it with a browser as would be the case if the user were writing HTML code. Thus, the tool enables pages/objects to be rendered on-the-fly.

(Bernardo, 10:5-13.) Bernardo thus discloses generating a preview of the web page (“**a display**”) based on user selections relating to the web page (“**in accordance with one or more user selected settings**”). Bernardo also discloses that this preview “enable[s] the user to view the Web page as it is created,” in other words, “to be rendered on-the-fly.” (Bernardo, 10:6-7, 10:12-13.) Bernardo does not provide specific timing with respect to the generation of the preview display, but an ordinarily skilled artisan would have understood “on-the-fly” rendering to satisfy any requirement of generating the display “**substantially contemporaneously**

with” the above-mentioned user selections, under any reasonable interpretation.

98. For example, as discussed in **Part V** above, various definitions have been proposed for this term, but Bernardo meets each definition. The “on-the-fly” rendering of the preview display in Bernardo clearly takes place “*at the same period of time from a human perspective,*” “*substantially in the same time period, from a human perspective,*” and “*immediately after,*” the entry of the user selection. This is the entire purpose of the on-the-fly preview in Bernardo—to provide immediate visual feedback to the user showing what the web page will look like, as it is being created by the user. (Bernardo, 10:5-13.) Moreover, the phrase “on-the-fly” as used in Bernardo would have been understood as referring to the immediate generation of the preview display without any significant or noticeable delay to the user.

99. Finally, I note that claim 1**[b]** merely recites “generating a display in accordance with **one or more user selected settings**,” which does not refer back to anything in claim 1**[a]** or otherwise require that the “**one or more user selected settings**” be the same settings presented in the “viewable menu” in claim 1**[a]**. Nevertheless, even if claim 1**[b]** were interpreted to require that the “one or more user selected settings” and the “selection thereof” pertain specifically to a selection from the viewable menu of claim 1**[a]**, Bernardo would still render this limitation obvious. For example, Bernardo confirms that the “preview” can apply in response

Declaration of Vijay K. Madiseti, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 6,546,397 B1

to the user making selections through Figure 8 which, as noted above, provides viewable menu(s) relating to selection of colors, background images, frame styles, and logos. (Bernardo, 8:56-62 (“FIG. 8 illustrates another aspect of the web site development module. This design center feature provides a screen from which the user is prompted to decide on the options for the design of the web page, for example, by selecting colors, background images, frame styles, and logos. Once selections have been made, the user can preview the design and then make changes as desired.”).) It therefore would have been obvious that the “on-the-fly” preview feature in Bernardo could have been applied in connection with selection from the “viewable menu” of Figure 8, or for that matter, in response to any of the other viewable menu selections in Bernardo as discussed above.

100. ***Manner, Rationale and Motivation to Combine:*** It would have been obvious to incorporate the preview feature of Bernardo into the ibook system of Reynolds. The combination would have predictably resulted in authoring tool **86** of Reynolds (which includes editing tool **136**), adapted to provide the further capability of displaying an on-the-fly preview of ibook web pages as they are being created or edited by the user. Although I provided a full description of the rationale and motivation to combine Reynolds with Bernardo for claim 1[a][1], additional motivations can be identified with respect to the Bernardo preview feature.

101. An ordinarily skilled artisan would have been amply motivated to adapt authoring tool **86** of Reynolds to include the Bernardo preview feature. The on-the-fly preview display feature would have improved the user experience of Reynolds by giving ibook creators and contributors immediate visual feedback regarding the appearance of ibook webpages. This would have allowed an ibook creator to immediately spot and address issues regarding the appearance of a web page, as opposed to waiting until after the creation process is finished to view the page as it would appear to a web page visitor. Bernardo explains that because the user can see a preview of the web page as it is being created, there is no need for the user “to close the file and reopen it with a browser as would be the case if the user were writing HTML code.” (Bernardo, 10:9-12.) This is a basic and well-known feature of WYSIWYG (“What You See is What You Get”), a common expression in the field for the ability of graphical user interfaces to provide a display that shows a document under development in the same manner in which it will appear when presented as a final product to a user.

102. As explained for claim 1[a][1], the combination of Reynolds with Bernardo would have been technologically straightforward, and an ordinarily skilled artisan would have had a reasonable expectation of success. For purposes of the preview feature, an ordinarily skilled artisan could have implemented the preview

feature into authoring tool **86** of Reynolds—which under the proposed combination takes the form of a Java applet—using conventional programming techniques. For example, the applet could have been implemented to display the on-the-fly preview using a portion of the same window used by authoring tool **86**, or in a separate window on the client computer display.

(d) “storing information representative of said one or more user selected settings in a database;” (Claim 1[c])

103. The “**database**” in Reynolds takes the form of ibook Web pages database **76**. As Reynolds explains, “[t]he material contributed to an ibook is contained in one or more Web pages. The content of the ibook Web pages is preferably stored in ibook Web pages database **76** (FIG. 3).” (Reynolds, 7:36-39.) As explained in claim limitations 1[d]-[e] below, navigation tool **84** at the client computer accesses requested passages of the ibook from Web page database **76** to display ibook web pages to the user. (Reynolds, *e.g.*, 11:31-40.)

104. It would have been obvious, based on the combination of Reynolds and Bernardo, to “**stor[e] information representative of said one or more user selected settings in a database,**” by storing the particular selections provided by the user through the viewable menus (see claim 1[a] above) in Web pages database **76** of Reynolds. Because “[t]he content of the ibook Web pages is preferably stored in ibook Web pages database **76**” (Reynolds, 7:36-39), it would have been obvious to

store user selections regarding that content in that same database. For example, as explained for claim 1[a][1], under the proposed combination of Reynolds and Bernardo, the user would have made selections relating to settings for elements of the web site through the viewable menus of Bernardo. An ordinarily skilled artisan, as noted, would have been motivated to adapt Reynolds to provide the viewable menus of Bernardo in order to allow the user greater control over the features and appearance of ibook web pages, among other benefits as discussed. It therefore would have been obvious to store those user selections in the database of Reynolds so the system could later build and display ibook web pages consistent with those selections. This is consistent with Bernardo, which discloses storing user selections in database templates for later use in constructing web pages. (Bernardo, *e.g.*, 8:1-10.) Under the combination with Reynolds, the user selections could be stored in Web pages database 76.⁴ I also note that my opinions would not change if the

⁴ I note that claim 1[c] simply recites “storing information *representative* of said one or more user selected settings in a database,” and thus does not require storage in any particular form or using any particular implementation. In an actual implementation with Reynolds, it would have been obvious to incorporate user selections (*e.g.*, colors, images, styles, etc.) into the content stored in web pages

user selections were stored in some other database within Reynolds, as it would have been obvious that user selections would need to be stored so they could be later retrieved to generate web pages consistent with those selections. It would make no difference to a person of ordinary skill in the art whether user selections were stored in Web pages database **76** of Reynolds or some separate database.

105. Reynolds also renders obvious the claimed “**database**” under any of the two prior claim constructions identified in **Part V**, which simply provide different formulations of describing conventional database technologies. The Web pages database **76** qualifies as “*an electronic information storage system offering data storage and retrieval,*” as Reynolds itself confirms that information is both stored in, and retrieved from, database **76** in connection with creating and later displaying ibook web pages. (See, e.g., Reynolds, 7:36-39, 11:31-40.) The longer definition identified in **Part V**, “*an electronic information storage system offering data storage and retrieval and that stores information on a record-by-record basis,*

database **76**. This would have been an obvious design choice because, as explained in more detail for claim 1[d] below, information from database **76** is retrieved by server **56** and sent to client **60** to build ibook web pages for display to a visiting user. (Reynolds, e.g., 11:24-40.)

each record divided into one or more fields,” uses more words but describes nothing more than conventional database technologies well-known to persons of ordinary skill in the art. (See *Microsoft Computer Dictionary* (3d ed. 1997), Ex. 1008, at p.129 (defining “database” as “[a] file composed of records, each containing fields together with a set of operations for searching, sorting, recombining, and other functions”).)

106. To the extent there is any doubt on this point, I have also cited to **Witkowski** (Ex. 1007), a patent assigned to Oracle that I cite merely for its description of prior art database technologies. In particular, the “Background of the Invention” in Witkowski provides the following prior art description:

In a database management system (DBMS), data is stored in one or more data containers, each container contains records, and the data within each record is organized into one or more fields. In relational database systems, the data containers are referred to as tables, the records are referred to as rows, and the fields are referred to as columns.

(Witkowski, 1:20-25.) Witkowski thus discloses a “database” under even the longer and more detailed definition provided in **Part V** above.

107. ***Manner, Rationale and Motivation to Combine:*** It would have been obvious to a person of ordinary skill in the art to combine Reynolds with the teachings of Witkowski discussed above, predictably resulting in the database of

Reynolds being implemented using relational database technologies. The database in Reynolds would thus have provided an electronic storage system offering data storage and retrieval in which information is stored in a series of rows (“on a record-by-record basis”), with each row divided into one or more columns (“fields”).

108. An ordinarily skilled artisan would have regarded this as a straightforward, and in fact trivial, combination as relational databases were well-known to ordinary skilled artisans—in fact, Reynolds expressly discloses the storage of passage information in a “relational database”:

Although various information concerning the ibook passages, such as derivation linking information, derivation type information, passage characteristics, and attribution information have been described as being stored in passage databases, **a suitable alternative is to store this information in the Web pages that contain the passages or as a set of relations and tables in a suitable relational database**, or as a set of objects in an object database.

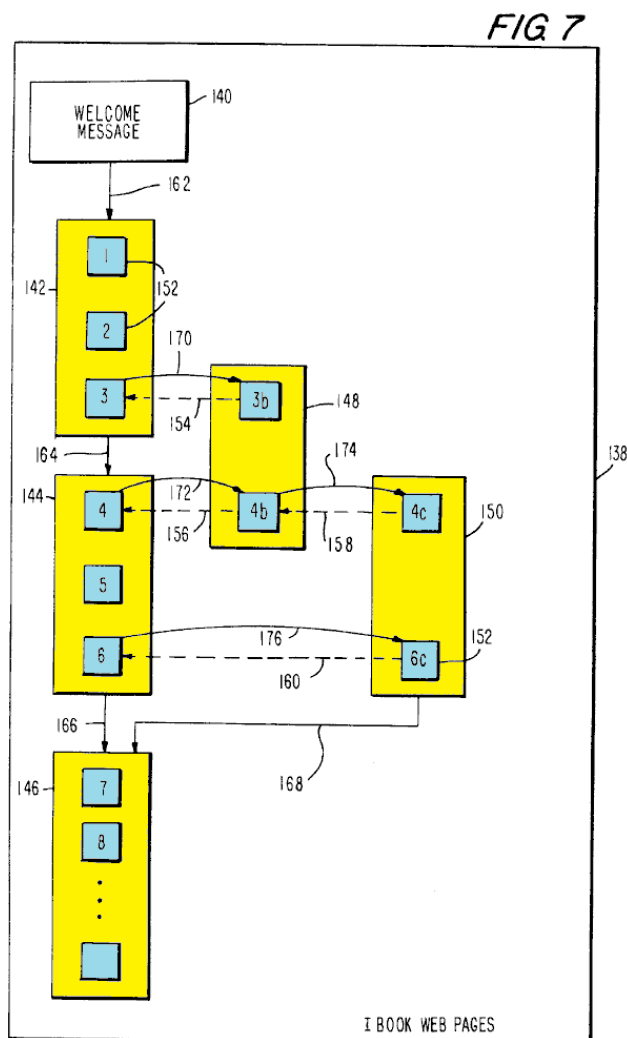
(Reynolds, 15:30-37.) Storing data in the form of tables using relational databases was by far the most popular form of database storage by December 1999. Relational databases were commercially available through off-the-shelf tools from third party vendors such as Oracle (the assignee of Witkowski), as well as freely available open source databases such as MySQL. An ordinarily skilled artisan would have been motivated to use relational database technologies to implement the database in

Reynolds because they were technologically mature, well-documented, stable and could be readily incorporated into existing applications. An ordinarily skilled artisan would thus have had every expectation of success in implementing the database of Reynolds as a relational database that organizes data into rows (records) and columns (fields). Harnessing preexisting relational database technologies would have enabled the ordinarily skilled artisan to more quickly implement the ibook system described in Reynolds, as it would have obviated the need to undertake the potentially time-consuming and costly process of designing and developing a “home-grown” data storage technique. It would also have increased the reliability of the database in Reynolds because, by December 1999, relational databases represented a reliable, decades-old technology. Developing a brand new and untested home-grown data storage technique, on the other hand, would have required considerably more testing in order to expose and correct any potential programming bugs, which would further increase delay and costs associated with the application development.

- (e) **“generating a website at least in part by retrieving said information representative of said one or more user selected settings stored in said database; and” (Claim 1[d])**

109. Reynolds discloses the step of **“generating a website,”** as claimed, by generating the ibook website, which in turn is made up of a number of ibook Web

pages. Reynolds explains that “[t]he material contributed to an ibook is contained in one or more Web pages.” (Reynolds, 7:36-37.) “Ibook Web pages **138** may contain both static Web pages and Web pages with content that changes dynamically whenever referenced.” (Reynolds, 7:39-41; *see also id.*, 15:42-48.) An exemplary ibook website is shown in Figure 7:



Declaration of Vijay K. Madiseti, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 6,546,397 B1

(Reynolds, Fig. 7 (highlighting added).) Figure 7 shows a number of ibook web pages **142, 144, 144, 146, 148** and **150** shown in **yellow** highlighting. (Reynolds, 7:57-65.) “Web pages **142, 144, 146, 148,** and **150** are preferably organized in the form of one or more passages **152,**” shown in **blue** highlighting within the web pages. (Reynolds, 7:66-67.) “For example, ibook Web page **142** contains passages **1, 2,** and **3** and ibook Web page **150** contains passages **4c** and **6c.**” (Reynolds, 7:67-8:2.) Reynolds makes clear that “passages” within ibook Web pages can contain any kind of data, including text, multimedia material (such as graphic images, animation sequences, audio or video data), and even executable materials. (Reynolds, 8:2-11, 7:24-31.) Figure 7 also shows that the ibook Web pages are connected via hyperlinks. For example, “[s]equence links **162, 164,** and **166** are hypertext links that link welcome message page **140** and ibook pages **142, 144,** and **146.**” (Reynolds, 8:53-55.) An ibook in Reynolds discloses a “**website**” because (among other things) it discloses a set of related and interconnected web pages relating to a particular ibook.

110. Reynolds also discloses generating the ibook website “**at least in part by retrieving said information representative of said one or more user selected settings stored in said database,**” as claimed. This is because, under the combination of Reynolds and Bernardo, the web pages for the ibook website are

generated by retrieving the user selections from ibook Web pages database **76**. A skilled artisan would have been motivated to build and display the web page “representative of said one or more user selected settings” for the same reasons as previously explained for claim 1[d]—to allow the user greater control over the features and appearance of the website and its pages. Thus, just as a skilled artisan would have been motivated to store user selections in the database of Reynolds, a skilled artisan would have been motivated to later build and display ibook web pages consistent with those user selections.

111. Reynolds explains that “[a]s the user navigates through an ibook using navigation tool **84**, navigation tool **84** transmits certain access information to server **56**, making requests to view material.” (Reynolds, 11:24-26.) After the user “determines which passage is to be viewed,” for example using buttons, navigation tool **84** “generates a request to view a desired passage and transmits this request to server **56** at step 200 [of Figure 9].” (Reynolds, 11:28-33.) The web server then “requests and locates the requested passage in ibook Web pages database **76**,” before transmitting that passage back to the client, which “displays the passage.” (*Id.*, 11:33-39.) Thus, Reynolds’ navigation tool **84** generates an ibook website by retrieving passages (containing the content reflecting the selections made by the user) from Web pages database **76**.

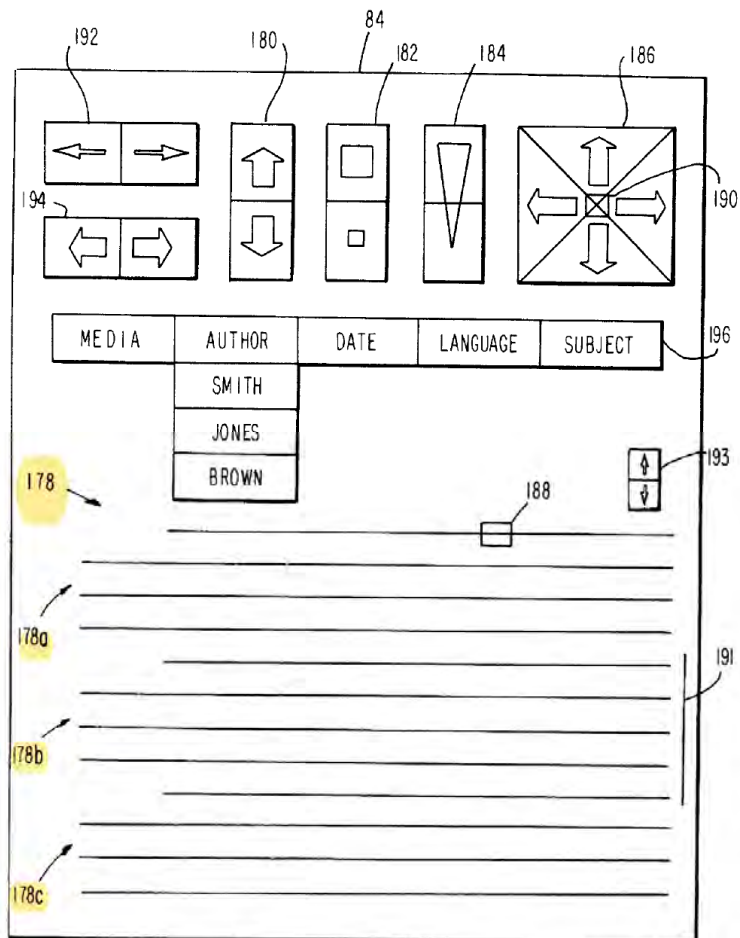
- (f) **“building one or more web pages to generate said website from at least a portion of said database and at least one run time file, where said at least one run time file utilizes information stored in said database to generate virtual machine commands for the display of at least a portion of said one or more web pages.”**
(Claim 1[e])

112. As I explained for claim 1[d] above, Reynolds discloses **“building one or more web pages to generate said website from at least a portion of said database,”** as claimed. This is because an ibook in Reynolds (**“a website”**) includes a number of individual web pages, which are **built** by Reynolds’ navigation tool by combining passages retrieved from web pages database **76**. (Reynolds, 1:29-30 (**“Material in an ibook is preferably organized in the form of Web pages.”**), 7:36-37, 11:24-39 (**“Navigation tool 84 preferably generates a request to view a desired passage . . . Server 56 transmits the requested passage to client 60 . . . Client 60 displays the passage”**).) The process for generating the website, as I described for claim 1[d], thus includes building the one or more web pages that make up that website. As I also explained, the web pages are built and generated **“from at least a portion of said database,”** because the client retrieves passages from ibook Web pages database **76** and combines the retrieved passages to produce those pages. (Reynolds, Fig. 7 (depicting web page **142** as the combination of passages **1**, **2**, and **3**), Fig. 8 (displaying a web page as the combination of passages **178a**, **178b**, and

178c).)

113. Reynolds also renders obvious building the one or more web pages “**from . . . at least one run time file,**” as claimed. The “**run time file**” in Reynolds takes the form of navigation tool **84**, which is one of the executable client plug-in modules **80** that are downloaded onto and executed by the browser on the client. (Reynolds, 5:28-37 (“Client plug-ins **80** are downloadable executable program modules that can be run on client **60** in conjunction with Web browser **82** to add additional capabilities to Web browser **82** . . . When client plug-ins **80** are downloaded onto client **60**, they form tools **84**, **86**, **88**, and **90**.”).) Figure 8 of Reynolds shows an illustrative navigation tool **84**:

FIG. 8



(Reynolds, Fig. 8 (highlighting added).) Figure 8 shows navigation tool **84** that displays “passages **152** [which] are illustrated as paragraphs of text **178**. In a multimedia environment, passages **152** would be multimedia segments presented on a computer monitor as video accompanied by sound.” (Reynolds, 9:48-52.) Navigation tool **84** also includes “navigation controls” that allow the viewing user to navigate the ibook website, including cursor keys **186** for navigating within a

passage, magnification buttons **184**, and back and forward buttons **194** for navigating between different Web pages. (Reynolds, 9:48-10:16, 10:27-31.)

114. As explained previously, navigation tool **84**, which was downloaded by and executed at Web browser **82** (“**at least one run time file**”), is responsible for downloading passages from the ibook Web pages database **76** and “**building**” web pages by combining the downloaded passages and presenting the web pages to the user. (Reynolds, 11:24-40.) Reynolds therefore also discloses that the run time file, which is executed at runtime, “**utilizes information stored in said database . . . for the display of at least a portion of said one or more web page,**” as explained in connection with claim 1[d] above.

115. Reynolds does not expressly disclose that navigation tool **84** utilizes information stored in said database “**to generate virtual machine commands,**” but this would have been obvious for substantially the same reasons provided in connection with claim 1[a][2] and the preamble above. It would have been obvious to implement navigation tool **84** of Reynolds as a Java applet. As with claim 1[a][2], the “**virtual machine commands**” under this combination would have corresponded to instructions of the Java applet implementing the navigation tool **84**, which are executed by the Java Virtual Machine (JVM) on the client computer. As explained previously, Meyer explains that “underlying each running Java applet, there is a Java

virtual machine that is doing all the running.” (Meyer, p.5.) Meyer further explains:

The Java Virtual Machine is at the core of the Java programming language. In fact, you can’t run a Java program without also running an implementation of the Java Virtual Machine. The Java Virtual Machine is the name of the engine that actually executes a Java program . . . Whenever you run a Java program, the instructions that make up the program are not executed directly by the hardware your system uses. Instead, a special piece of software steps through each of the instructions in the program and carries out the action that the instruction represents.

(Meyer, p.5.) The instructions of a Java applet thus represent “**virtual machine commands**,” as claimed, because they direct the Java Virtual Machine as to the operations it must perform—in this case the functions of navigation tool **84**.

116. The prior art thus renders obvious that “**said at least one run time file utilizes information stored in said database to generate virtual machine commands for the display of at least a portion of said one or more web pages**,” as claimed. This is because, under the proposed combination, a Java applet would implement the functionality of navigation tool 84 of Reynolds which, as noted, utilizes information in the form of passages stored in and obtained from iBook Web pages database 76 (“**information stored in said database**”), to generate and issue instructions to the Java Virtual Machine (JVM) (“**to generate virtual machine**

commands”), for building web pages by combining those passages and displaying the web pages in the browser (“**for the display of at least a portion of said one or more web pages**”).

117. ***Rationale and Motivation to Combine:*** The rationale and motivation to combine was fully explained for claim 1[a][2], and applies fully here. As explained, it would have been obvious to combine the teachings of Reynolds with the Java applet teachings of Meyer. As relevant to this claim limitation the combination would have predictably resulted in implementing navigation tool **84** of Reynolds as a Java applet.

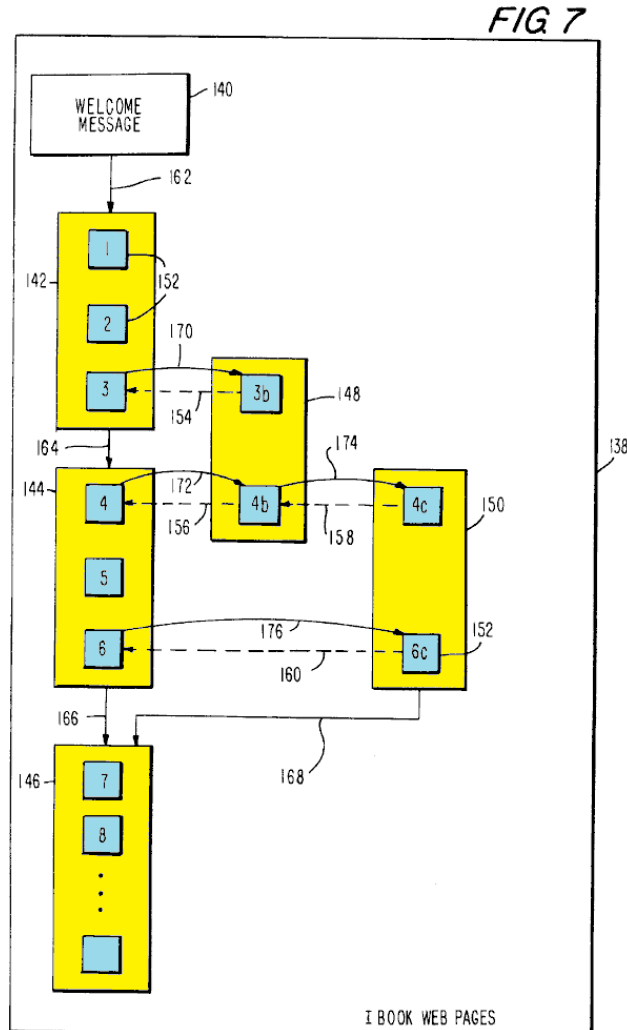
118. I explained for claim 1[a][2] why it would have been obvious to an ordinarily skilled artisan to implement authoring tool **86** (another plug-in module in Reynolds) as a Java applet, and those same reasons apply to navigation tool **84**. As explained, Meyer identifies many compelling advantages of Java applets including portability, execution speed and efficiency, the benefits of the dynamic features of the Java programming language, built-in safety and security features, and ability to make efficient use of the computing resources available on the user’s local computer. (Meyer, pp.2-4; *see also* analysis accompanying claim 1, preamble.) And for the same reasons explained above, an ordinarily skilled artisan would have had more than a reasonable expectation of success.

119. As explained for claim 1[a][2], Meyer provides a further motivation to combine by stating that Java applets “can be used to create interactive data displays” (Meyer, p.4), which is precisely the type of functionality provided by navigation tool **84**. An ordinarily skilled artisan would have understood that navigation tool **84** in Reynolds would have benefited from implementation as a Java applet in order to facilitate the high degree of interactivity between the user and the system in requesting user input and retrieving and displaying information in response.

C. Ground 2: Claim 1 is obvious over Reynolds, Bernardo, Meyer, and Witkowski, in Further View of Coombs

120. As explained above for Ground 1, Reynolds discloses a method that allows users to provide **Internet websites**, in the form of interactive Web books or “ibooks” accessible via the World Wide Web. (Reynolds, Abstract; *see also id.*, 1:5-8 (“This invention relates to the Internet, and more particularly, to techniques for creating and viewing material on the World Wide Web in the form of an interactive Web book.”).) Reynolds explains that each “Web book,” or “ibook,” is composed of a collection of web pages. (Reynolds, 3:46-47 (“Contributed material is preferably stored in the form of ibook web pages 42”), 7:36-37 (“The material contributed to an ibook is contained in one or more Web pages”).).

121. In my opinion, Reynolds discloses a “**website**” because it discloses a collection of related web pages, as shown in Figure 7:



(Reynolds, Fig. 7 (highlighting added).) Figure 7 shows a number of iBook web pages **142, 144, 144, 146, 148** and **150** shown in **yellow** highlighting. (Reynolds, 7:57-65.) “Web pages **142, 144, 146, 148** and **150** are preferably organized in the form of one or more passages **152**,” shown in **blue** highlighting within the web pages. (Reynolds, 7:66-67.) Figure 7 also shows that the iBook Web pages are connected via hyperlinks. For example, “[s]equence links **162, 164**, and **166** are hypertext links

that link welcome message page **140** and ibook pages **142**, **144**, and **146**.”
(Reynolds, 8:53-55.) In my opinion, therefore, an ibook in Reynolds discloses a
“**website**” because (among other things) it discloses a set of related (and
interconnected) web pages all relating to a particular ibook.

122. But Reynolds does not expressly use the word “**website**” to describe
the collection of ibook web pages. Because the ’397 patent does not provide a
definition of “**website**,” I have been asked to provide my opinion in the event the
Patent Owner were to argue that “**website**” requires web pages be accessed using a
separate and dedicated domain name (such as “*www.domainname.com*”). In my
opinion, even if “**website**” were construed in this fashion, and Patent Owner were to
argue that Reynolds does not disclose a website under this definition, such a
definition would not provide a non-obvious distinction over the prior art. As I will
explain below, it would have been obvious to an ordinarily skilled artisan to register
a domain name for an ibook such that viewing users would access the ibook web
pages through a URL defined by a separate and dedicated domain name. For all
other limitations of claim 1, my analysis from Ground 1 applies here.

123. Registering a domain name for a website was (and remains to this day)
one of the most basic and easy initial steps one can undertake to create a website.
As Coombs explains, obtaining a domain name for a website was as simple as

confirming that the domain name was not already taken, and if it was not, submitting an application for the domain name with the appropriate registrar and paying the appropriate fee. (Coombs, pp.29-39.) The “first steps that everyone needs to take in setting up an Internet site,” such as a web site, are “getting a domain name and establishing a presence on the Internet.” (Coombs, p.3.)

124. ***Rationale and Motivation to Combine:*** It would have been obvious to implement the ibook system according to the combined teachings of Reynolds, Bernardo, Meyer, and Witkowski—as described in Ground 1—under a separate domain name pursuant to the teachings of Coombs. This would have predictably resulted in the ibook system in which the web pages for an ibook would have been accessible under a separate and dedicated domain name, for example, a URL in a form such as <http://www.reynoldsibook.com>. This URL is composed of various sections identifying the website: http:// identifies the HTTP protocol for the browser to communicate with Reynolds’ web server, “www” identifies that the ibook is located on the “World Wide Web,” and reynoldsibook.com identifies the domain which is translated to the IP address of Reynolds’ web server. (See Coombs, p.351 (defining the “Hypertext Transfer Protocol (HTTP)” as “[a] system for transferring files across the World Wide Web”); p.358 (defining the “World Wide Web (WWW)” as “[t]he total selection of Web Sites interconnected via the Internet”),

Declaration of Vijay K. Madiseti, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 6,546,397 B1

p.357 (explaining that the “http://” prefix in a URL is used for the “World Wide Web”).) Under this combination, therefore, all claim limitations requiring a “**website**” in claim 1 would have been satisfied by the ibook of Reynolds, even under the narrow interpretation discussed above, as the ibook web pages would have been available under a separate domain name dedicated to that ibook. For all other limitations of claim 1, my analysis from Ground 1 applies here.

125. An ordinarily skilled artisan would have had many motivations to apply the teachings of Coombs in the manner suggested. As noted, Coombs confirms that domain name registration was a common first step in establishing a separate website. The teachings of Coombs are thus analogous because they are reasonably pertinent to the ability to establish a separate domain name for a website, a common technique of enabling the website to be accessed by visitors.

126. Reynolds and Coombs both provide express motivations to combine. First, as any user of web pages would appreciate, a domain name provides a simple and easy way for visitors to reach a website. Coombs explains that obtaining a domain name enables Internet users to access a website without “worry[ing] about translating computer names into IP addresses.” (Coombs, p.25.) It would have been straightforward for a “sponsor who sets up an ibook on an ibook server” (Reynolds, 3:17-21), to register a desired domain name, such as reynoldsibook.com. This in

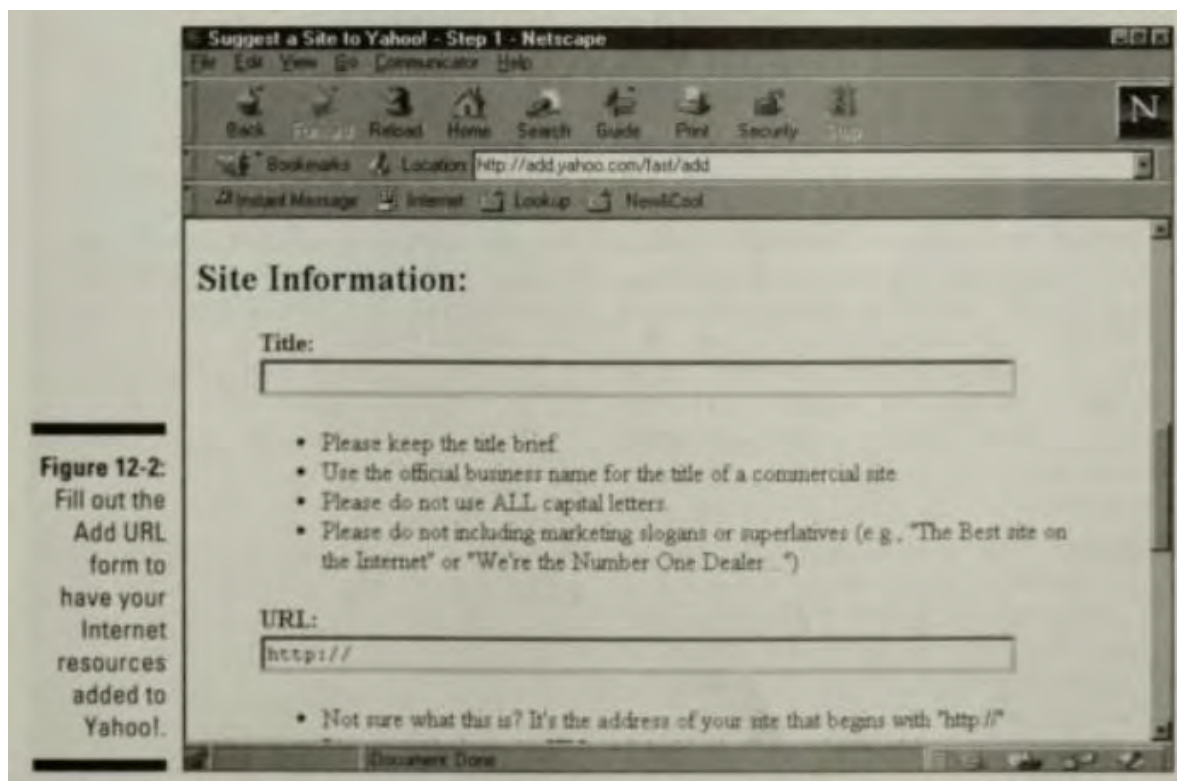
Declaration of Vijay K. Madiseti, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 6,546,397 B1

turn would have made it easier to access the website.

127. Second, Coombs explains that registering a website domain name would have enabled a website developer to register the website with popular search engines such as Yahoo!. Reynolds itself explains the desirability for advertising the ibook's presence by adding the ibook to a "Web index" maintained by a search engine such as "Yahoo, Alta Vista, and Lycos." (See Reynolds, 3:17-21 ("Ibooks may initially be supported by a sponsor who sets up an ibook on an ibook server and who may, for example, advertise that ibook's presence and chosen theme by arranging to have suitable information regarding the ibook included in Web index 40."), 14:15-21 ("ibooks preferably support various functions for advertising their presence to popular search engine databases. Suitable search engines to which the ibook can advertise its presence include Yahoo, Alta Vista, and Lycos.").) Coombs confirms that having a separate domain and URL would have made this a simple process. As Coombs explains, "[t]aking advantage of search engines in promoting your site is the single most important thing you can do in marketing your site," and thus "[h]aving Yahoo! include your Internet site in its directory is very important." (Coombs, p.246.)

128. Coombs illustrates how a skilled artisan would have registered an ibook website to the Yahoo! directory, by submitting a form to Yahoo! at

<http://add.yahoo.com/fast/add:>



(Coombs, Fig. 12-2, p.249.) Thus, Reynolds' ibook sponsor could easily advertise an ibook web site on the Yahoo! web index by submitting the ibook's title and URL under the registered domain name. This would have provided a further motivation to combine.

VII. NO SECONDARY CONSIDERATIONS OF NON-OBVIOUSNESS

129. I understand from counsel that the Patent Owner in the underlying district court litigation has not yet identified any evidence with respect to secondary considerations of non-obviousness.

Declaration of Vijay K. Madiseti, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 6,546,397 B1

130. To the extent Patent Owner cites any evidence of sales or any praise or any industry recognition of products that Patent Owner asserts to implement the claimed invention, I am not aware of any information demonstrating that any increased sales, commercial success, praise, or any other secondary factor (that Patent Owner may assert) was a result of *the particular features recited in the '397 patent's claims*. Since Patent Owner has not yet identified any evidence of secondary considerations, Patent Owner cannot demonstrate that the limitations of the claimed invention in particular, as opposed to other features of the products at issue, were the factors that caused any increased sales, praise, or any other asserted secondary considerations.

131. Thus, based on my review of the evidence to date, I can summarize my opinions regarding any alleged secondary considerations of non-obviousness relating to the '397 patent, as follows:

132. *No commercial success of the claimed invention.* Patent Owner has not cited any evidence of particular commercial success of products embodying the '397 patent as opposed to products that do not embody the '397 patent. Patent Owner has not cited any evidence that any commercial success of any products is particularly a result of the claimed inventions recited in the '397 patent's claims and not due to any other facts.

Declaration of Vijay K. Madiseti, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 6,546,397 B1

133. *No long-felt but unsolved need.* Patent Owner has not cited any evidence of any long-felt need that remained unsolved in the prior art before the '397 patent. To the contrary, as discussed above, the prior art solved the problems that the '397 patent purported to address.

134. *No failure of others.* Patent Owner has not cited any evidence of anyone who tried, but failed, to solve the problems addressed by the '397 patent. As shown by my analysis above, there existed prior art references that successfully disclosed and rendered obvious the subject matter claimed by the '397 patent.

135. *No copying of the claimed invention.* Patent Owner has not cited any evidence that any other party (including Facebook or third parties) ever copied from the '397 patent and its claimed invention.

136. *No unexpected results of the claimed invention.* Patent Owner has not cited any evidence of unexpected results achieved by the '397 patent's claimed invention. To the contrary, the prior art disclosed the predictable, expected results that show why the '397 patent's claims are obvious as discussed in my Declaration.

137. *No praise for the claimed invention.* Patent Owner has not cited any evidence of praise for the claimed invention recited in the '397 patent.

138. *No surprise or skepticism at the claimed invention.* Patent Owner has not cited any evidence that observers were surprised by, or skeptical of, the claimed

Declaration of Vijay K. Madiseti, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 6,546,397 B1

invention recited in the '397 patent.

139. *No departure from the wisdom of the prior art.* Patent Owner has not cited any evidence that the claimed inventions of the '397 patent departed from the wisdom of the prior art. The '397 patent claims subject matter that was already present in the prior art, including in the references discussed in my Declaration.

140. Moreover, with respect to the considerations discussed above, I also refer to and incorporate my opinions stated throughout this Declaration, including my analysis showing that the '397 patent is directed to techniques known in the prior art and does not provide any inventive technology.

141. To the extent Patent Owner at a later date cites or provides any other evidence regarding secondary considerations, including any expert opinions, I reserve the right to supplement my analysis and opinions to comment on it.

VIII. CONCLUSION

142. In signing this Declaration, I recognize that the Declaration will be filed as evidence in a contested case before the Patent Trial and Appeal Board of the United States Patent and Trademark Office. I also recognize that I may be subject to cross-examination in this proceeding. If required, I will appear for cross-examination at the appropriate time. I reserve the right to offer opinions relevant to the invalidity of the '397 patent claims at issue and/or offer testimony in support of this Declaration.

143. I hereby declare that all statements made herein of my own knowledge are true and that all statements are made on information and belief are believed to be true, and further 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 18 U.S.C. § 1001.

Dated: July 1, 2021

Respectfully submitted,


Vijay K. Madiseti, Ph.D

EXHIBIT A

Dr. Vijay K. Madiseti
Fellow, IEEE
vkm@madiseti.com
Cell: 770-527-0177
Address:
56 Creekside Park Drive
Johns Creek, GA 30022

Employment:

- 1989-present: Full Professor of Electrical & Computer Engineering (Georgia Tech, Atlanta, GA 30332).

Areas of Technical Interest – Embedded Systems, Wireless & Mobile Communications, Computer Engineering, Circuit Design (Analog/Digital), Biomedical Instrumentation Hardware/Software, Software Engineering, Digital Signal Processing, Sensors, Wireline & Wireless Computer Networks, Software Systems, Control Systems, Chip Design, Cloud Computing.

Education: PhD (EECS, University of California at Berkeley, 1989), B.Tech (Hons) in Electronics and Electrical Communications Engineering (Indian Institute of Technology, Kharagpur, 1984).

Startup Companies:

Director, VP Technologies, Inc. (1995-): A startup commercialized through Georgia Tech's Advanced Technology Development Corporation (ATDC) focusing on digital software and hardware design services for military market. <http://www.vptinc.net>

Director, Soft Networks, LLC (2001-2007): A startup commercialized through Georgia Tech support focusing on software development tools and compilers for Cellular/WiFi/VOIP/telecommunication products. <http://www.soft-networks.com>

Director, Elastic Video Inc. (2007- 2009): A startup commercialized through Georgia Tech's VentureLab (<http://venturelab.gatech.edu>) development image and video processing software for wireless & IP networking.

Litigation Experience (2015-2021) With Testimony

(Note: There may be multiple cases between the parties, e.g., District Court v. ITC, US versus Foreign Cases)

Chrimar v. Adtran, Alcatel, et al.

Technology: Power over Ethernet (2015-2017)

Submitted reports & Deposition & Trial

Chamberlain v. Ryobi/TTI

Case No: 1:16-cv-06097 (ND Illinois)

Expert for Ryobi

Technology: Wireless/IoT/Barrier Movement (2016 – 2019)

Submitted reports & deposition & trial testimony

IOEngine v. IMC/Imation

Case No: cv-14-1572-GMS (US Delaware)

Expert for IMC/Imation

Technology: Networked Storage Device (2016-2017)

Submitted reports & deposition & trial testimony

Huawei v. Samsung

Case No: 3:16-cv-2787-WHO (ND Cal)

Expert for Samsung

Technology: 4G/LTE Random Access Protocols (2016-2019)

Submitted reports & deposition

Hitachi Maxell v. ZTE/Huawei

Case No: 5:16-cv-00178-RWS (ED Texas)

Expert for Hitachi Maxell

Technology: Digital Cameras

Submitted reports & deposition (2017 – 2018)

Hitachi Maxell v. Apple

Case No: 5:19-cv-00036-RWS

Expert for Hitachi Maxell

Technology: Digital Cameras

Submitted Expert Reports (2020-)

Qualcomm v. Apple

Case No: 17-ccv-0108-GPC-MDD (SD Cal) Also, Related ITC/FTC Matters

Expert for Qualcomm

Technology: 4G/Wireless Communications/Smartphones (2017-2019)

Submitted Reports and Deposition

Qualcomm v. Apple

Case No: 3:17-cv-01375-DMS-MDD (SD Cal)

Expert for Qualcomm
Technology: 4G/Wireless Communications/Smartphones (2017-2019)
Submitted Reports and Deposition

Optis v. Huawei

Case No: 2:17-cv-123 (E.D. Texas)
Expert for Optis Wireless
Technology: 4G/Video (2017-2019)
Submitted reports & deposition

Beckman Coulter v. Sysmex

Case No: 1:17-cv-24049-DPG (ND Illinois)
Expert for Sysmex
Technology: Medical Instrumentation Automation (2017-present)
Testifying Expert

TQ Delta v. Xyxel/Adtran

Expert for TQ Delta
Technology: DSL Technologies (2018-present)
Testifying Expert

3GL/KPN v. LG, Blackberry, HTC

Expert for 3GL/KPN
Technology: 4G/LTE Protocols (2018-present)
Reports and Deposition / Testifying Expert

Rovi v. Comcast

ITC No. 337-TA-1103 (ITC)
Expert for Rovi
Technology: Digital Video & interactive GUI (Jan 2018-May 2018)
Reports and Deposition

Cirba/Densify v. VMWare

Case No: 1:19-cv-00742-LPS
Expert for Cirba/Densify
Technology: Virtualization (2019-present)
Reports Deposition/PI Hearing / Testifying Expert

Power Integrations v. On Semiconductor

Case No: 17-cv-03189-BLF
Expert for On Semiconductor
Technology: Power Electronics (2018-2019)
Reports & Deposition/Testifying Expert.

Pathway v. Lumens, IPEVO, Aver JDA

Case No: ITC 337-TA-1045

Expert for Lumens JDA

Technology: Document Cameras (2016-2019)

Reports & Deposition (for IPR/ITC, no trial)

Wilan v. Apple

Case No: 3:14-cv-2235

Expert for WiLan

Technology: Voice over LTE/LTE Protocols (2017-2020)

Reports and Testified through Deposition/Trials

St. Lawrence Comm. v. Amazon

Case No. 2:19-CV-00027-JRG

Expert for Amazon

Technology: Multimedia Codecs

Submitted Expert Reports/Deposition (2020-2020)

PanOptis v. Apple

Case No. 2:19-CV-00066-JRG

Expert for PanOptis

Technology: LTE

Submitted Expert Reports/Deposition/Trial (2019-2020)

GAS v. Sprint Communications

Case No. 2:20-cv-00007-RWS

Expert for General Access (GAS)

Technology: LTE

Submitted Expert Reports/Deposition (9/21-)

Additional matters include declarations supporting IPRs at the PTAB for Google (US Patent 8,601,154), On Semiconductor (US Patent 6,212,079), Ubisoft (US Patent 5,490,216), Lumens JDA (2017-2019), Broadcom (WiFi), Sony (US Patent 6,101,534), Kia, (US Patent 5,530,431), Qualcomm, Ericsson, Daimler Benz, VW, Estech, Amazon, Ring, Digital Ally, On Semiconductor, United Patents, Lenovo, BMC Software, BMW, Daimler and Dolby.

Earned Degrees

1. B. Tech (Hons), Electronics & Electrical Comm. Engineering
Indian Institute of Technology (IIT), Kharagpur, India
1984.

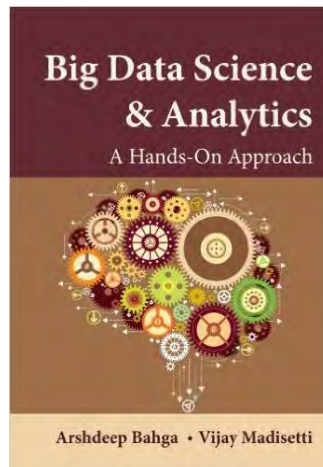
2. Ph.D., Electrical Engineering & Computer Sciences (EECS)
University of California (UC), Berkeley. CA
1989.

Books

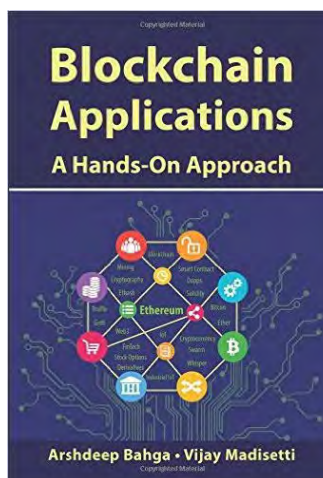
1. VLSI Digital Signal Processors
Madiseti, V.K.;
Boston: MA, IEEE Press: Butterworth Heinemann, 1995, 525 pp.
2. Quick-Turnaround ASIC Design in VHDL
Romdhane, M., Madiseti, V.K., Hines, J.
Boston: MA, Kluwer Academic Press, 1996, 190 pp.
3. The Digital Signal Processing Handbook (First Edition)
Madiseti, V. K., Williams, D. (Editors)
CRC Press, Boca Raton, Fla, 1998, 2500 pp.
4. VHDL: Electronics Systems Design Methodologies.
Madiseti, V. K. (Editor)
Boston: MA, IEEE Standards Press, 2000, ISBN 0-7381-1878-8.
5. Platform-Centric Approach to System-on-Chip (SoC) Design.
Madiseti, V. K., Arpnikanondt, A.
Springer, Boston: MA, Springer, 2004, 280 pp.
6. The Digital Signal Processing Handbook – Second Edition.
Madiseti, V. K. (2009)
CRC Press, Boca Raton, Fla.
7. Cloud Computing: A Hands-On Approach
A Bahga, V. Madiseti (2013)
Amazon CreateSpace Publishing, 2013, 454 pp.
8. Internet of Things: A Hands-On Approach
A Bahga, V. Madiseti (2014)
Amazon CreateSpace Publishing, 2014, 450 pp.



9. Big Data Science & Analytics: A Hands-On Approach
A Bahga, V. Madiseti (2016)
Amazon CreateSpace Publishing, 2016, 542 pp.



10. Blockchain Applications: A Hands-On Approach
A Bahga, V. Madiseti (2017)
Amazon CreateSpace Publishing, 2017, 380 pp.



Edited Books & Collection of Papers

1. Advances in Parallel & Distributed Simulation
Madiseti, V.K.; Nicol, D., Fujimoto, R. (Editors)
San Diego, CA: SCS Press, 1991, 200 pp.
2. Modeling, Analysis, Simulation of Computer & Telecommunications Systems
Madiseti, V., Gelenbe, E., Walrand, J. W. (Editors)
Los Alamitos: CA, IEEE Computer Society Press, 1994, 425 pp.

3. Modeling & Simulations on Microcomputers
Madiseti, V.K. (Editor)
San Diego, CA: SCS Press, 138 pp. 1990.

Editorship of Journals & Transactions

1. IEEE Design & Test of Computers
Special Issue: Reengineering Digital Systems
April – June 1999 (Vol 16, No 2)
Madiseti, V.K (Editor)
Los Alamitos: CA, IEEE Computer Society Press, 1999.
2. IEEE Design & Test of Computers
Special Issue: Rapid Prototyping of Digital Systems
Fall 1996 (Vol 13, No 3)
Madiseti, V., Richards, M. (Editors)
Los Alamitos: CA, IEEE Computer Society Press, 1994, 425 pp.
3. IEEE Transactions on Circuits & Systems II
Associate Editor: 1993-1995.
4. International Journal in Computer Simulation
Associate Editor: 1990-1993
5. International Journal in VLSI Signal Processing
Editorial Board: 1995 - Present

Recent Issued US Patents

- [1] [10,503,927](#) [Method and system for securing cloud storage and databases from insider threats and optimizing performance](#), Issued Dec 10, 2019
- [2]. [10,460,283](#) [Smart contract optimization for multiparty service or product ordering system](#), Issued Oct 29, 2019
- [3]. [10,459,946](#) [Method and system for tuning blockchain scalability, decentralization, and security for fast and low-cost payment and transaction processing](#), Issued Oct 29, 2019

- [4]. [10,402,589](#) [Method and system for securing cloud storage and databases from insider threats and optimizing performance](#), Issued Sep 3, 2019
- [5]. [10,394,845](#) [Method and system for tuning blockchain scalability for fast and low-cost payment and transaction processing](#), Issued Aug 27, 2019
- [6]. [10,289,631](#) [Method and system for tuning blockchain scalability for fast and low-cost payment and transaction processing](#), Issued May 24, 2019
- [7]. [10,255,342](#) [Method and system for tuning blockchain scalability, decentralization, and security for fast and low-cost payment and transaction processing](#), Issued April 9, 2019
- [8]. [10,243,743](#) [Tokens or crypto currency using smart contracts and blockchains](#), Issued March 26, 2019
- [9]. [10,204,339](#) [Method and system for blockchain-based combined identity, ownership, integrity and custody management](#), Issued February 12, 2019
- [10]. [10,204,148](#) [Method and system for tuning blockchain scalability, decentralization, and security for fast and low-cost payment and transaction processing](#), Issued Feb 12, 2019
- [11]. [10,121,143](#) [Method and system for blockchain-based combined identity, ownership, integrity and custody management](#), Issued Nov 6, 2018
- [12]. [10,102,526](#) [Method and system for blockchain-based combined identity, ownership, integrity and custody management](#), October 16, 2018
- [13]. [10,102,265](#) [Method and system for tuning blockchain scalability for fast and low-cost payment and transaction processing](#), October 16, 2018
- [14]. [9,935,772](#) [Methods and systems for operating secure digital management aware applications](#), April 3, 2018
- [15]. [9,769,213](#) [Method and system for secure digital object management](#), Issued Sep 19, 2017

Refereed Journal Publications

1. Trends in the Electronic Control of Mine Hoists
Madiseti, V. and Ramlu, M.,
IEEE Transactions on Industry Applications, Vol IA-22, No. 6,
November/December 1986. Pages 1105-1112
2. Multilevel range/NEXT performance in digital subscriber loops
Brand, G.; Madiseti, V.; Messerschmitt, D.G.;
Communications, Speech and Vision, IEE Proceedings I [see also IEE

Proceedings-Communications] ,Volume: 136 , Issue: 2 , April 1989
Pages:169 – 174

3. Seismic migration algorithms on parallel computers
Madiseti, V.K.; Messerschmitt, D.G.;
Signal Processing, IEEE Transactions on [see also Acoustics, Speech, and
Signal Processing, IEEE Transactions on] ,Volume: 39 , Issue: 7 , July 1991
Pages:1642 – 1654
4. Asynchronous algorithms for the parallel simulation of event-driven
dynamical systems
Madiseti, V.K.; Walrand, J.C.; Messerschmitt, D.G.:
ACM Transactions on Modeling and Computer Simulation, v 1, n 3, July 1991,
Pages: 244-74
5. Synchronization mechanisms for distributed event-driven
computation
Madiseti, V.K.; Hardaker, D.:
ACM Transactions on Modeling and Computer Simulation, v 2, n 1, Jan. 1992,
Pages: 12-51
6. Efficient VLSI Architectures for the Arithmetic Fourier Transform
(AFT)
Kelley, B.T.; Madiseti, V.K.;
Signal Processing, IEEE Transactions on [see also Acoustics, Speech, and
Signal Processing, IEEE Transactions on] ,Volume: 41 , Issue: 1 , January
1993
Pages:365-378
7. The fast discrete Radon transform. I. Theory
Kelley, B.T.; Madiseti, V.K.;
Image Processing, IEEE Transactions on ,Volume: 2 , Issue: 3 , July 1993
Pages:382 – 400
8. The Georgia Tech digital signal multiprocessor
Barnwell, T.P., III; Madiseti, V.K.; McGrath, S.J.A.;
Signal Processing, IEEE Transactions on [see also Acoustics, Speech, and
Signal Processing, IEEE Transactions on] ,Volume: 41 , Issue: 7 , July 1993
Pages:2471 – 2487
9. The MIMDIX Environment for Parallel Simulation
Madiseti, V.K.; Hardaker, D.; Fujimoto, R.M.:
Journal of Parallel and Distributed Computing, v18, no. 4, August 1993,
Pages: 473-83.

10. LMSGEN: a prototyping environment for programmable adaptive digital filters in VLSI
Romdhane, M.S.B.; Madiseti, V.K.;
Chapter in VLSI Signal Processing, VII, 1994.,
Pages: 33 – 42
11. Fixed-point co-design in DSP
Egolf, T.W.; Famorzadeh, S.; Madiseti, V.K.;
Chapter in VLSI Signal Processing, VII, 1994.,
Pages: 113 - 126
12. A fast spotlight-mode synthetic aperture radar imaging system
Madiseti, V.K.;
Communications, IEEE Transactions on , Volume: 42 , Issue: 234 , February-April 1994
Pages: 873 – 876
13. Rapid prototyping on the Georgia Tech digital signal multiprocessor
Curtis, B.A.; Madiseti, V.K.;
Signal Processing, IEEE Transactions on [see also Acoustics, Speech, and Signal Processing, IEEE Transactions on] , Volume: 42 , Issue: 3 , March 1994
Pages: 649 – 662
14. Low-power signaling in asymmetric noisy channels via spectral shaping
Sipitca, M.; Madiseti, V.K.;
Signal Processing Letters, IEEE, Volume: 1 , Issue: 8 , Aug 1994
Pages: 117 – 118
15. A quantitative methodology for rapid prototyping and high-level synthesis of signal processing algorithms
Madiseti, V.K.; Curtis, B.A.;
Signal Processing, IEEE Transactions on [see also Acoustics, Speech, and Signal Processing, IEEE Transactions on] , Volume: 42 , Issue: 11 , Nov. 1994
Pages: 3188 – 3208
16. Computer Simulation of Application-Specific Signal Processing Systems
Casinovi, G.; Madiseti, V.K.;
International Journal in Computer Simulation, Vol. 4, No. 4, Nov 1994.
17. System partitioning of MCMs for low power
Khan, S.A.; Madiseti, V.K.;
Design & Test of Computers, IEEE , Volume: 12 , Issue: 1 , Spring 1995
Pages: 41 – 52

18. Error correcting run-length limited codes for magnetic recording
Jaejin Lee; Madiseti, V.K.;
Magnetics, IEEE Transactions on ,Volume: 31 , Issue: 6 , Nov. 1995
Pages:3084 – 3086
19. Virtual prototyping of embedded microcontroller-based DSP systems
Madiseti, V.K.; Egolf, T.W.;
Micro, IEEE ,Volume: 15 , Issue: 5 , Oct. 1995
Pages:9 – 21
20. Constrained multitrack RLL codes for the storage channel
Lee, J.; Madiseti, V.K.;
Magnetics, IEEE Transactions on ,Volume: 31 , Issue: 3 , May 1995
Pages:2355 – 2364
21. Rapid digital system prototyping: current practice, future challenges
Madiseti, V.K.;
Design & Test of Computers, IEEE ,Volume: 13 , Issue: 3 , Fall 1996
Pages:12 – 22
22. Conceptual prototyping of scalable embedded DSP systems
Dung, L.-R.; Madiseti, V.K.;
Design & Test of Computers, IEEE ,Volume: 13 , Issue: 3 , Fall 1996
Pages:54 – 65
23. Advances in rapid prototyping of digital systems
Madiseti, V.K.; Richards, M.A.;
Design & Test of Computers, IEEE ,Volume: 13 , Issue: 3 , Fall 1996
Pages:9
24. Combined modulation and error correction codes for storage channels
Jaejin Lee; Madiseti, V.K.;
Magnetics, IEEE Transactions on ,Volume: 32 , Issue: 2 , March 1996
Pages:509 – 514
25. Model-based architectural design and verification of scalable embedded DSP systems-a RASSP approach
Dung, L.-R.; Madiseti, V.K.; Hines, J.W.;
Chapter in VLSI Signal Processing, IX, 1996.
Pages:147 – 156

26. Low-power digital filter implementations using ternary coefficients
Hezar, R.; Madiseti, V.K.;
Chapter in VLSI Signal Processing, IX, 1996.,
Pages:179 – 188
27. All-digital oversampled front-end sensors
Romdhane, M.S.B.; Madiseti, V.K.;
Signal Processing Letters, IEEE, Volume: 3 , Issue: 2 , Feb. 1996
Pages:38 – 39
28. Modeling COTS components in VHDL
Calhoun, S., Reese, R; Egolf, T., Madiseti, V.K.;
Journal of VLSI Signal Processing, Volume: 14 , Issue: 2 , Nov 1996
Pages: 24 – 31
29. VHDL-Based Rapid Systems Prototyping
Egolf, T.; Madiseti, V.K.;
Journal of VLSI Signal Processing, Volume: 14 , Issue: 2 , Nov 1996
Pages: 40-52
30. Interface design for core-based systems
Madiseti, V.K.; Lan Shen;
Design & Test of Computers, IEEE ,Volume: 14 , Issue: 4 , Oct.-Dec. 1997
Pages:42 - 51
31. Incorporating cost modeling in embedded-system design
Debardelaben, J.A.; Madiseti, V.K.; Gadiant, A.J.;
Design & Test of Computers, IEEE ,Volume: 14 , Issue: 3 , July-Sept. 1997
Pages:24 – 35
32. On homomorphic deconvolution of bandpass signals
Marenco, A.L.; Madiseti, V.K.;
Signal Processing, IEEE Transactions on [see also Acoustics, Speech, and
Signal Processing, IEEE Transactions on] ,Volume: 45 , Issue: 10 , Oct. 1997
Pages:2499 – 2514
33. A case study in the development of multi-media educational material:
the VHDL interactive tutorial
Gadiant, A.J.; Stinson, J.A., Jr.; Taylor, T.C.; Aylor, J.H.; Klenke, R.H.;
Salinas, M.H.; Madiseti, V.K.; Egolf, T.; Famorzadeh, S.; Karns, L.N.; Carter,
H.W.;
Education, IEEE Transactions on ,Volume: 40 , Issue: 4 , Nov. 1997
Pages:17 pp.

34. Adaptive mobility management in wireless networks
Jeongwook Kim; Madisetti, V.K.;
Electronics Letters ,Volume: 34 , Issue: 15 , 23 July 1998
Pages:1453 – 1455
35. Efficient implementation of two-band PR-QMF filterbanks
Hezar, R.; Madisetti, V.K.;
Signal Processing Letters, IEEE ,Volume: 5 , Issue: 4 , April 1998
Pages:92 – 94
36. On fast algorithms for computing the inverse modified discrete cosine transform
Yun-Hui Fan; Madisetti, V.K.; Mersereau, R.M.;
Signal Processing Letters, IEEE ,Volume: 6 , Issue: 3 , March 1999
Pages:61 – 64
37. System on chip or system on package?
Tummala, R.R.; Madisetti, V.K.;
Design & Test of Computers, IEEE ,Volume: 16 , Issue: 2 , April-June 1999
Pages:48 – 56
38. Reengineering legacy embedded systems
Madisetti, V.K.; Jung, Y.-K.; Khan, M.H.; Kim, J.; Finnessy, T.;
Design & Test of Computers, IEEE ,Volume: 16 , Issue: 2 , April-June 1999
Pages:38 – 47
39. Reengineering digital systems
Madisetti, V.K.;
Design & Test of Computers, IEEE ,Volume: 16 , Issue: 2 , April-June 1999
Pages:15 – 16
40. Parameter optimization of robust low-bit-rate video coders
Sangyoun Lee; Madisetti, V.K.;
Circuits and Systems for Video Technology, IEEE Transactions on, Volume: 9
Issue: 6 , Sept. 1999
Pages:849 – 855
41. Closed-form for infinite sum in bandlimited CDMA
Jatunov, L.A.; Madisetti, V.K.;
Communications Letters, IEEE ,Volume: 8 , Issue: 3 , March 2004
Pages:138 – 140
42. A new protocol to enhance path reliability and load balancing in mobile ad hoc networks
Argyriou, A.; Madisetti, V.K.;

Journal of Ad Hoc Networks, Elsevier Press, 2004

43. Closed-form analysis of CDMA systems using Nyquist pulse
Jatunov, L.A.; Madisetti, V. K.;
Communications Letters, IEEE (Under Revision), 2005.
44. Systematic Design of End-to-End Wireless Mobility Management
Protocols,
Argyriou, A.; Madisetti, V. K.;
ACM/Springer Wireless Networks (WINET), Accepted 2005.
45. A Novel End-to-End Approach for Video Streaming Over the Internet,
Argyriou, A.; Madisetti, V. K.;
Kluwer Telecommunications Systems, Vol. 28, No. 2, Pages 133-150, Jan
2005. Special Issue on Multimedia Streaming.
46. An Analytical Framework of RD Optimized Video Streaming with TCP,
Argyriou, A.; Madisetti, V. K.;
IEEE Transactions on Multimedia, Submitted for review in March 2005.
47. Modeling the Effect of Handoffs on Transport Protocol Performance,
Argyriou, A.; Madisetti, V. K.;
IEEE Transactions on Mobile Computing, Submitted for review in March 2005
48. Throughput Models for Transport Protocols with CBR and VBR Traffic
Workloads",
Argyriou, A.; Madisetti, V. K.;
ACM Transactions on Multimedia Computing, Communications & Applications,
Submitted for review in April 2005.
49. "Electronic System, Platform & Package Codesign",
Madisetti, V. K.
IEEE Design & Test of Computers, Volume 23, Issue 3, June 2006. pages
220-233.
50. "The Design of an End-to-End Handoff Management Protocol",
A. Argyriou, Madisetti, V. K.
Wireless Networks, Springer, May 2006.
51. "A Soft-Handoff Transport Protocol for Media Flows in Heterogeneous
Mobile Networks ",
A. Argyriou, Madisetti, V. K.
Computer Networks, Vol 50, Issue 11, Pages 1860-1871, August 2006.

52. "Computationally Efficient SNR Estimation for Bandlimited WCDMA Systems"
L. Jatunov, Madiseti, V. K.
IEEE Transactions on Wireless Communications, Volume 5, Issue 13,
December 2006, Pages 3480-3491.
53. "Space-Time Codes for Wireless & Mobile Applications",
M. Sinnokrot, Madiseti, V.K.
DSP Handbook, Second Edition, 2009 (to be published)
- 54.A. Bahga, V. Madiseti, "Rapid Prototyping of Advanced Cloud-Based Systems", IEEE Computer, vol. 46, no. 11, Nov 2013, pp 76-83, 2013
- 55.A. Bahga, V. Madiseti, "Cloud-Based Information Integration & Informatics Framework for Healthcare Applications", IEEE Computer, February 2015.
- 56.A. Bahga, V. Madiseti, "A Cloud-based Approach for Interoperable EHRs", IEEE Journal of Biomedical and Health Informatics, vol. 17, no. 5, Sep 2013, pp. 894-906, 2013
- 57.A. Bahga, V. Madiseti, "Cloud-Based Information Technology Framework for Data Driven Intelligent Transportation Systems", Journal of Transportation Technologies, vol.3 no.2, April 2013
- 58.A. Bahga, V. Madiseti, "Performance Evaluation Approach for Multi-tier Cloud Applications", Journal of Software Engineering and Applications, vol. 6, no. 2, pp. 74-83, Mar 2013.
- 59.Yusuf, A., V. Madiseti, "Configuration for Predicting Travel Time Using Wavelet Packets and Support Vector Regression", Journal of Transportation Technologies, vol 3, no. 3, June 2013.
- 60.A. Bahga, V. Madiseti, "Analyzing Massive Machine Maintenance Data in a Computing Cloud", IEEE Transactions on Parallel and Distributed Systems, vol. 23, no. 10, pp. 1831 - 1843, 2012.
- 61.N. Radia. Y. Zhang, M. Tatimapula, V. Madiseti, "Next Generation Applications on Cellular Networks: Trends, Challenges, and Solutions," Proceedings of the IEEE, Vol 100, Issue 4, pp. 841-854, 2012.
- 62.A. Bahga, V. Madiseti, "Rapid Prototyping of Advanced Cloud-Based Systems", IEEE Computer, vol. 46, no. 11, Nov 2013, pp 76-83, 2013

- 63.A. Bahga, V. Madiseti, "A Cloud-based Approach for Interoperable EHRs", IEEE Journal of Biomedical and Health Informatics, vol. 17, no. 5, Sep 2013, pp. 894-906, 2013
- 64.A. Bahga, V. Madiseti, "Cloud-Based Information Integration & Informatics Framework for Healthcare Applications", IEEE Computer, February 2015.

Peer Reviewed Conference Publications

1. Dynamically-reduced complexity implementation of echo cancelers
Madiseti, V.; Messerschmitt, D.; Nordstrom, N.;
Acoustics, Speech, and Signal Processing, IEEE International Conference on
ICASSP '86. ,Volume: 11 , Apr 1986
2. Seismic migration algorithms using the FFT approach on the NCUBE
multiprocessor
Madiseti, V.K.; Messerschmitt, D.G.;
Acoustics, Speech, and Signal Processing, 1988. ICASSP-88., 1988
International Conference on , 11-14 April 1988
3. Seismic migration algorithms on multiprocessors
Madiseti, V.K.; Messerschmitt, D.G.;
Acoustics, Speech, and Signal Processing, 1988. ICASSP-88., 1988
International Conference on , 11-14 April 1988
Pages:2124 - 2127 vol.4
4. WOLF: A rollback algorithm for optimistic distributed simulation
systems
Madiseti, V.; Walrand, J.; Messerschmitt, D.;
Simulation Conference Proceedings, 1988 Winter , December 12-14, 1988
Pages:296 – 305
5. Efficient distributed simulation
Madiseti, V.; Walrand, J.; Messerschmitt, D.;
Simulation Symposium, 1989. The 22nd Annual , March 28-31, 1989
Pages:5 - 6
6. High speed migration of multidimensional seismic data
Kelley, B.; Madiseti, V.;

- Acoustics, Speech, and Signal Processing, 1991. ICASSP-91., 1991
International Conference on , 14-17 April 1991
Pages:1117 - 1120 vol.2
7. Performance of a fast analog VLSI implementation of the DFT
Buchanan, B.; Madiseti, V.; Brooke, M.;
Circuits and Systems, 1992., Proceedings of the 35th Midwest Symposium on
, 9-12 Aug. 1992
Pages:1353 - 1356 vol.2
8. Task scheduling in the Georgia Tech digital signal multiprocessor
Curtis, B.A.; Madiseti, V.K.;
Acoustics, Speech, and Signal Processing, 1992. ICASSP-92., 1992 IEEE
International Conference on ,Volume: 5 , 23-26 March 1992
Pages:589 - 592 vol.5
9. The fast discrete Radon transform
Kelley, B.T.; Madiseti, V.K.;
Acoustics, Speech, and Signal Processing, 1992. ICASSP-92., 1992 IEEE
International Conference on ,Volume: 3 , 23-26 March 1992
Pages:409 - 412 vol.3
10. Yield-based system partitioning strategies for MCM and ASEM design
Khan, S.; Madiseti, V.;
Multi-Chip Module Conference, 1994. MCMC-94, Proceedings., 1994 IEEE, 15-
17 March 1994
Pages:144 - 149
11. Multitrack RLL codes for the storage channel with immunity to
intertrack interference
Lee, J.; Madiseti, V.K.;
Global Telecommunications Conference, 1994. GLOBECOM '94.
'Communications: The Global Bridge'. IEEE, Volume: 3 , 28 Nov.-2 Dec. 1994
Pages:1477 - 1481 vol.3
12. A parallel mapping of backpropagation algorithm for mesh signal
processor
Khan, S.A.; Madiseti, V.K.;
Neural Networks for Signal Processing [1995] V. Proceedings of the 1995
IEEE Workshop , 31 Aug.-2 Sept. 1995
Pages:561 - 570
13. Virtual prototyping of embedded DSP systems
Madiseti, V.K.; Egolf, T.; Farnzadeh, S.; Dung, L.-R.;
Acoustics, Speech, and Signal Processing, 1995. ICASSP-95., 1995

International Conference on ,Volume: 4 , 9-12 May 1995
Pages:2711 - 2714 vol.4

14. Assessing and improving current practice in the design of application-specific signal processors
Shaw, G.A.; Anderson, J.C.; Madisetti, V.K.;
Acoustics, Speech, and Signal Processing, 1995. ICASSP-95., 1995
International Conference on ,Volume: 4 , 9-12 May 1995
Pages:2707 - 2710 vol.4
15. Introduction to ARPA's RASSP initiative and education / facilitation program
Corley, J.H.; Madisetti, V.K.; Richards, M.A.;
Acoustics, Speech, and Signal Processing, 1995. ICASSP-95., 1995
International Conference on ,Volume: 4 , 9-12 May 1995
Pages:2695 - 2698 vol.4
16. DSP design education at Georgia Tech
Madisetti, V.K.; McClellan, J.H.; Barnwell, T.P., III;
Acoustics, Speech, and Signal Processing, 1995. ICASSP-95., 1995
International Conference on ,Volume: 5 , 9-12 May 1995
Pages:2869 - 2872 vol.5
17. Rapid prototyping of DSP systems via system interface module generation
Famorzadeh, S.; Madisetti, V.K.;
Acoustics, Speech, and Signal Processing, 1996. ICASSP-96. Conference Proceedings., 1996 IEEE International Conference on ,Volume: 2 , 7-10 May 1996
Pages:1256 - 1259 vol. 2
18. Rapid prototyping of DSP chip-sets via functional reuse
Romdhane, M.S.B.; Madisetti, V.K.;
Acoustics, Speech, and Signal Processing, 1996. ICASSP-96. Conference Proceedings., 1996 IEEE International Conference on ,Volume: 2 , 7-10 May 1996
Pages:1236 - 1239 vol. 2
19. A constructive deconvolution procedure of bandpass signals by homomorphic analysis
Marenco, A.L.; Madisetti, V.K.;
Geoscience and Remote Sensing Symposium, 1996. IGARSS '96. 'Remote Sensing for a Sustainable Future.', International ,Volume: 3 , 27-31 May 1996
Pages:1592 - 1596 vol.3

20. BEEHIVE: an adaptive, distributed, embedded signal processing environment
Famorzadeh, S.; Madisetti, V.; Egolf, T.; Nguyen, T.;
Acoustics, Speech, and Signal Processing, 1997. ICASSP-97., 1997 IEEE
International Conference on ,Volume: 1 , 21-24 April 1997
Pages:663 - 666 vol.1
21. Target detection from coregistered visual-thermal-range images
Perez-Jacome, J.E.; Madisetti, V.K.;
Acoustics, Speech, and Signal Processing, 1997. ICASSP-97., 1997 IEEE
International Conference on ,Volume: 4 , 21-24 April 1997
Pages:2741 - 2744 vol.4
22. Variable block size adaptive lapped transform-based image coding
Klausutis, T.J.; Madisetti, V.K.;
Image Processing, 1997. Proceedings., International Conference on ,Volume:
3 , 26-29 Oct. 1997
Pages:686 - 689 vol.3
23. A Rate 8/10 (0, 6) MTR Code And Its Encoder / decoder
Jaejin Lee; Madisetti, V.K.;
Magnetics Conference, 1997. Digests of INTERMAG '97., 1997 IEEE
International , 1-4 April 1997
Pages:BS-15 - BS-15
24. VHDL models supporting a system-level design process: a RASSP
approach
DeBardelaben, J.A.; Madisetti, V.K.; Gadiant, A.J.;
VHDL International Users' Forum, 1997. Proceedings , 19-22 Oct. 1997
Pages:183 – 188
25. A performance modeling framework applied to real time infrared
search and track processing
Pauer, E.K.; Pettigrew, M.N.; Myers, C.S.; Madisetti, V.K.;
VHDL International Users' Forum, 1997. Proceedings , 19-22 Oct. 1997
Pages:33 – 42
26. System design and re-engineering through virtual prototyping: a
temporal model-based approach
Khan, M.H.; Madisetti, V.K.;
Signals, Systems & Computers, 1998. Conference Record of the Thirty-
Second Asilomar Conference on ,Volume: 2 , 1-4 Nov. 1998
Pages:1720 - 1724 vol.2

27. A debugger RTOS for embedded systems
Akgul, T.; Kuacharoen, P.; Mooney, V.J.; Madiseti, V.K.;
Euromicro Conference, 2001. Proceedings. 27th , 4-6 Sept. 2001
Pages:264 - 269
28. Adaptability, extensibility and flexibility in real-time operating systems
Kuacharoen, P.; Akgul, T.; Mooney, V.J.; Madiseti, V.K.;
Digital Systems, Design, 2001. Proceedings. Euromicro Symposium on , 4-6 Sept. 2001
Pages:400 – 405
29. Effect of handoff delay on the system performance of TDMA cellular systems
Turkboylari, M.; Madiseti, V.K.;
Mobile and Wireless Communications Network, 2002. 4th International Workshop on , 9-11 Sept. 2002
Pages:411 – 415
30. Enforcing interdependencies and executing transactions atomically over autonomous mobile data stores using SyD link technology
Prasad, S.K.; Bourgeois, A.G.; Dogdu, E.; Sunderraman, R.; Yi Pan; Navathe, S.; Madiseti, V.;
Distributed Computing Systems Workshops, 2003. Proceedings. 23rd International Conference on , 19-22 May 2003
Pages:803 – 809
31. Performance evaluation and optimization of SCTP in wireless ad-hoc networks
Argyriou, A.; Madiseti, V.;
Local Computer Networks, 2003. LCN '03. Proceedings. 28th Annual IEEE International Conference on , 20-24 Oct. 2003
Pages:317 - 318
32. Implementation of a calendar application based on SyD coordination links
Prasad, S.K.; Bourgeois, A.G.; Dogdu, E.; Sunderraman, R.; Yi Pan; Navathe, S.; Madiseti, V.;
Parallel and Distributed Processing Symposium, 2003. Proceedings. International , 22-26 April 2003
Pages:8 pp.
33. Bandwidth aggregation with SCTP
Argyriou, A.; Madiseti, V.;
Global Telecommunications Conference, 2003. GLOBECOM '03. IEEE Volume: 7 , 1-5 Dec. 2003
Pages:3716 - 3721 vol.7

34. Software streaming via block streaming
Kuacharoen, P.; Mooney, V.J.; Madisetti, V.K.;
Design, Automation and Test in Europe Conference and Exhibition, 2003
, 2003
Pages:912 – 917
35. Frequency-dependent space-interleaving for MIMO OFDM systems
Mohajerani, P.; Madisetti, V.K.;
Radio and Wireless Conference, 2003. RAWCON '03. Proceedings , Aug. 10-13, 2003
Pages:79 - 82
36. A media streaming protocol for heterogeneous wireless networks
Argyriou, A.; Madisetti, V.;
Computer Communications, 2003. CCW 2003. Proceedings. 2003 IEEE 18th Annual Workshop on , 20-21 Oct. 2003
Pages:30 – 33
37. Realizing load-balancing in ad-hoc networks with a transport layer protocol
Argyriou, A.; Madisetti, V.;
Wireless Communications and Networking Conference, 2004. WCNC. 2004 IEEE ,Volume: 3 , 21-25 March 2004
Pages:1897 - 1902 Vol.3
38. Streaming H.264 / AVC video over the Internet
Argyriou, A.; Madisetti, V.;
Consumer Communications and Networking Conference, 2004. CCNC 2004. First IEEE , 5-8 Jan. 2004
Pages:169 – 174

Other Publications

1. A Transport Layer Technology for Improving the QoS of Networked Multimedia Applications <draft-madisetti-argyriou-qos-sctp-00.txt>. Madisetti, V., Argyriou, A. IETF Internet-Draft, Jul 25, 2002.
2. Voice & Video over Mobile IP Networks <draft-madisetti-argyriou-voice-video-mip-00.txt> Madisetti, V., Argyriou, A. IETF Internet-Draft, Nov 20, 2002.
3. Enhancements to ECRTTP with Applications to Robust Header Compression for Wireless Applications. <draft-madisetti-rao-suresh-rohc-00.txt> Madisetti, V.; Rao, S., Suresh, N. IETF Internet-Draft, June 30, 2003.

Ph.D. Students Graduated

1. Brian T. Kelley, 1992
VLSI Computing Architectures for High Speed Signal Processing
Member of Technical Staff, Motorola.

Winner of Dr. Thurgood Marshall Dissertation Fellowship Award
2. Bryce A. Curtis, 1992
Special Instruction Set Multiple Chip Computer for DSP
Member of Technical Staff, IBM
3. Jaejin Lee, 1994
Robust Multitrack Codes for the Magnetic Channel
Professor, Yonsei University, Korea
4. Mohamed S. Ben Romdhane, 1995
Design Synthesis of Application-Specific IC for DSP
Director of IP, Rockwell.
5. Shoab A. Khan, 1995
Logic and Algorithm Partitioning on MCMs
Professor, National University of Science & Technology, Pakistan
6. Lan-Rong Dung, 1997
VHDL-based Conceptual Prototyping of Embedded DSP Architectures
Professor, National Chaio Tung University, Taiwan.

Winner of VHDL International Best PhD Thesis Award, 1997

7. Thomas W. Egolf, 1997
Virtual Prototyping of Embedded DSP Systems
Distinguished Member of Technical Staff, Agere
8. Alvaro Marenco, 1997
On Homomorphic Deconvolution of Bandpass Signals
Professor, Texas A&M University.

Winner of GIT ECE Outstanding Teaching Assistant Award

9. Shahram Famorzadeh, 1997
BEEHIVE: A Distributed Environment for Adaptive Signal Processing
Member of Technical Staff, Rockwell.
10. Timothy J. Klausutis, 1997
Adaptive Lapped Transforms with Applications to Image Coding.
US Air Force/Univ. of Florida.
11. Lan Shen, 1998
Temporal Design of Core-Based Systems
Member of Technical Staff, IBM
12. James DeBardelaben, 1998
Optimization Based Approach to Cost Effective DSP Design
Research Scientist, Johns Hopkins University

Georgia Tech ECE Faculty Award

13. Sangyoun Lee, 1999
Design of Robust Video Signal Processors
Professor, Yonsei University

US Army Sensors Lab Research Excellence Award, 1999

14. Rahmi Hezar, 2000
Oversampled Digital Filters
Member of Technical Staff, Texas Instruments
15. Yong-kyu Jung, 2001
Model-Based Processor Synthesis
Professor, Texas A&M University
16. Mustafa Turkboylari, 2002
Handoff Algorithms for Wireless Applications
Member of Technical Staff, Texas Instruments
17. Yun-Hui Fan, 2002
A Stereo Audio Coder with Nearly Constant Signal to Noise Ratio
Post-Doctoral Research Associate, Northeastern University
18. Subrato K. De, 2002

Design of a Retargetable Compiler for DSP
Member of Technical Staff, Qualcomm

US Army Sensors Lab Research Excellence Award, 1999

19. Chonlameth Aripnikanondt, 2004
System-on-Chip Design with UML
Professor, King Mongkut's University, Thailand.

US Army Sensors Lab Research Excellence Award, 1999

20. Loran Jatunov, 2004
Performance Analysis of 3G CDMA Systems
Senior Research Scientist, Soft Networks, LLC.

21. Antonios Argyriou, 2005, Serving in Hellenic Army.

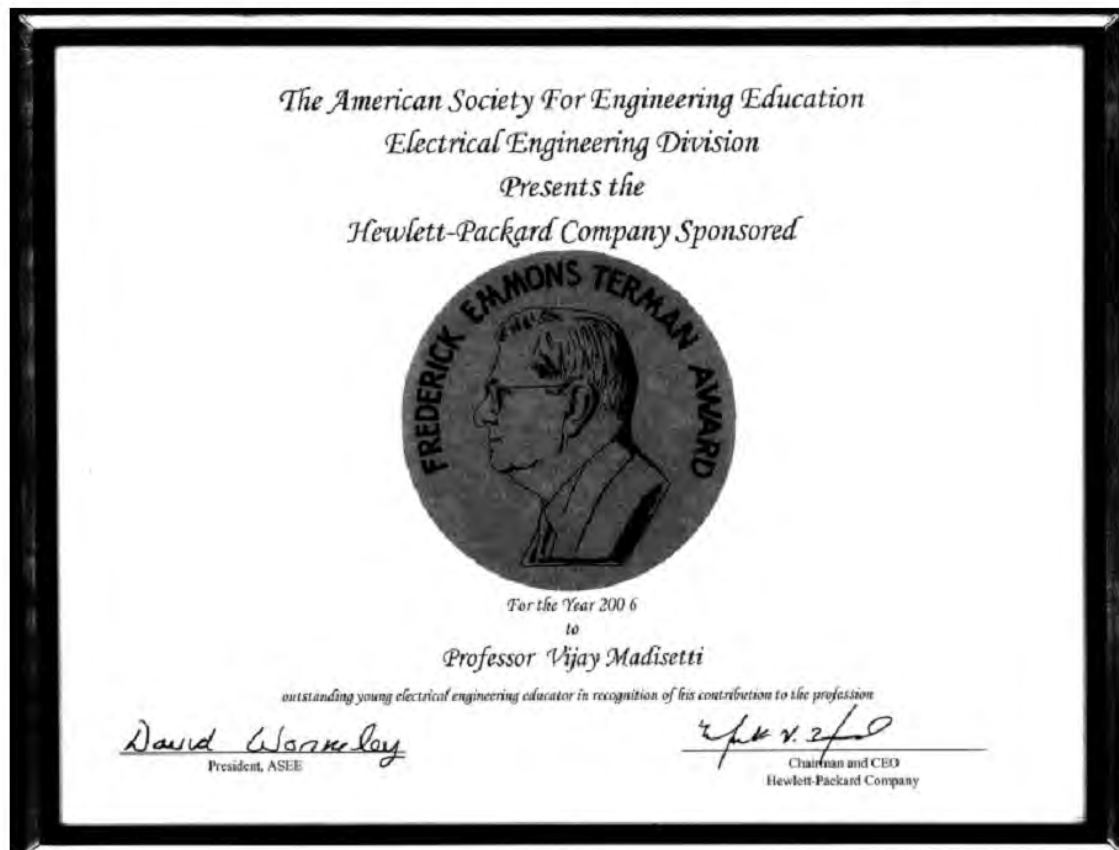
22. Pilho Kim, 2009, Scientist, VP Technologies, Inc.

23. M. Sinnokrot, 2009, Staff Engineer, Qualcomm.

Awards & Honors

1. **Jagadis Bose National Science Talent Fellowship**, Indian Institute of Technology, Kharagpur, 1980-1984.
2. **General Proficiency Prize**, Indian Institute of Technology, Kharagpur, 1984.
3. **Demetri Angelakos Outstanding Graduate Student Award**, Univ. of California, Berkeley, 1989
4. **Ira Kay IEEE/ACM Best Paper Award** for Best Paper presented at IEEE Annual Simulation Symposium, 1989
5. **IBM Faculty Development Award** 1990
6. **Technical Program Chair**, IEEE Workshop on Parallel and Distributed Simulation. 1990.
7. **Technical Program Chair**, IEEE MASCOTS'94
8. **NSF RI Award**, 1990
9. **VHDL International Best PhD Dissertation Advisor Award**, 1997

10. Georgia Tech Outstanding Doctoral Dissertation Advisor Award, 2001.
11. ASEE 2006 Frederick Emmons Terman Medal, 2006.
12. Fellow of IEEE



Intellectual Property Disclosures (Georgia Tech)

<u>Patent</u>	<u>Date</u>	<u>Description</u>
2843	2004	Method and Apparatus for Improving the Performance of Wireless LANs
2825	2003	Method and Apparatus for Optimal Partitioning and Ordering of Antennas for Layered Space-Time Block Codes in MIMO Communications Systems
2815	2003	How to Rapidly Develop a SyD Application
GSU-023	2003	Rapid Development of SyD Applications

2810	2003	System on Mobile Devices Middleware Design
2718	2003	A Transport Layer Algorithm for Improved Anycast Communication
2717	2003	A Novel Transport Layer Load-Balancing Algorithm
2716	2003	A Transport Layer QoS Algorithm
2715	2003	A Novel Transport Layer Algorithm for MPLS Performance
2659	2002	A New Algorithm and Technology for Implementing Mobile IP with Applications to Voice and Video over Mobile IP
2656	2002	Debugging with Instruction-Level Reverse Execution
2655	2002	Embedded Software Streaming
2539		System of Databases: An Enabling Technology for Programming
2517	2002	A Dynamic Instantiated Real-Time Operating System Debugger
2516	2002	A Dynamic Real-Time Operating System
GSU-009	2001	System of Databases: Architecture,, Global Queries, Triggers and Constraints
2480	2001	Mobile Fleet Application based on SyD Technology
2479	2001	System of Databases: A model with coordination links and a calendar application
1893	1999	Beehive
1726	1995	Very High Scale Integrated Circuit Hardware Description Language Models (VHDL Models)
1401	1995	Self-Compensation Receiver (SCR)

Issued Patents: 9,935,772, 9,769,213