

Declaration of Vijay K. Madisetti, Ph.D. in Support of  
Petition for *Inter Partes* Review of  
U.S. Patent No. 7,594,168 B2

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. 7,594,168 B2  
Issue Date: September 22, 2009

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 '168 PATENT .....                                                                                              | 20   |
| A. Overview of the Specification.....                                                                                  | 20   |
| B. The Challenged Claims .....                                                                                         | 22   |
| V. CLAIM CONSTRUCTION .....                                                                                            | 24   |
| VI. APPLICATION OF THE PRIOR ART TO ASSERTED CLAIMS .....                                                              | 26   |
| A. Brief Summary of Prior Art .....                                                                                    | 27   |
| 1. Reynolds [Ex. 1003] .....                                                                                           | 27   |
| 2. Lemay Web [Ex. 1004].....                                                                                           | 29   |
| 3. Lemay Java [Ex. 1005] .....                                                                                         | 32   |
| 4. Perry [Ex. 1006].....                                                                                               | 33   |
| 5. Witkowski [Ex. 1007].....                                                                                           | 35   |
| 6. Coombs [Ex. 1008] .....                                                                                             | 38   |
| B. Ground 1: Claims 1-4, and 6 are obvious over Reynolds in<br>view of Lemay Web, Lemay Java, Perry and Witkowski..... | 39   |
| 1. Independent Claim 1 .....                                                                                           | 39   |
| (a) “a server comprising a build engine configured<br>to:” (Claim 1[a]) .....                                          | 42   |

## Table of Contents

(continued)

Page

|     |                                                                                                                                                                                                                                                                                                                                                                                                                                  |    |
|-----|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|----|
| (b) | “accept user input to create a web site, the web site comprising a plurality of web pages, each web page comprising a plurality of objects,” (Claim 1[b]) .....                                                                                                                                                                                                                                                                  | 43 |
| (c) | “accept user input to associate a style with objects of the plurality of web pages, wherein each web page comprises at least one button object or at least one image object, and wherein the at least one button object or at least one image object is associated with a style that includes values defining transformations and time lines for the at least one button object or at least one image object;” (Claim 1[c])..... | 50 |
| (d) | “and wherein each web page is defined entirely by each of the plurality of objects comprising that web page and the style associated with the object,” (Claim 1[d]) .....                                                                                                                                                                                                                                                        | 61 |
| (e) | “produce a database with a multidimensional array comprising the objects that comprise the web site including data defining, for each object, the object style, an object number, and an indication of the web page that each object is part of, and” (Claim 1[e]).....                                                                                                                                                          | 64 |
| (f) | “provide the database to a server accessible to [sic; a] web browser;” (Claim 1[f]) .....                                                                                                                                                                                                                                                                                                                                        | 77 |
| (g) | “wherein the database is produced such that a web browser with access to a runtime engine is configured to generate the web-site from the objects and style data extracted from the provided database.” (Claim 1[g]).....                                                                                                                                                                                                        | 80 |
| 2.  | Claim 2: “The system of claim 1, wherein one of said plurality of objects is a child, and wherein the build engine is configured to accept user input to associate a style with child button and child image objects.” .....                                                                                                                                                                                                     | 83 |

**Table of Contents**  
(continued)

|                                                                                                                                                                                                                                | <b>Page</b> |
|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-------------|
| 3. Claim 3: “The system of claim 2, wherein at least one of said styles includes values defining time lines for child button and child image objects.”.....                                                                    | 88          |
| 4. Claim 4: “The system of claim 1, wherein at least one of said styles includes settings for multiple object states.”.....                                                                                                    | 89          |
| 5. Claim 6: “The system of claim 1, where said data is stored as one or more of a Boolean an integer, a string, a floating point variables, or a URL.” .....                                                                   | 89          |
| C. Ground 2: Claims 1-4 and 6 are obvious over Reynolds in view of Lemay Web, Lemay Java, Perry, and Witkowski, in Further View of Coombs.....                                                                                 | 95          |
| 1. Independent Claim 1 .....                                                                                                                                                                                                   | 95          |
| 2. Claim 2: “The system of claim 1, wherein one of said plurality of objects is a child, and wherein the build engine is configured to accept user input to associate a style with child button and child image objects.”..... | 101         |
| 3. Claim 3: “The system of claim 2, wherein at least one of said styles includes values defining time lines for child button and child image objects.”.....                                                                    | 102         |
| 4. Claim 4: “The system of claim 1, wherein at least one of said styles includes settings for multiple object states.”.....                                                                                                    | 102         |
| 5. Claim 6: “The system of claim 1, where said data is stored as one or more of a Boolean an integer, a string, a floating point variables, or a URL.” .....                                                                   | 102         |
| VII. NO SECONDARY CONSIDERATIONS OF NON-OBVIOUSNESS ....                                                                                                                                                                       | 102         |
| VIII. CONCLUSION.....                                                                                                                                                                                                          | 106         |

Declaration of Vijay K. Madiseti, Ph.D. in Support of  
Petition for *Inter Partes* Review of  
U.S. Patent No. 7,594,168 B2

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. 7,594,168 B2

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. 7,594,168 B2

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. 7,594,168 B2

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. 7,594,168 B2

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. 7,594,168 B2

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. 7,594,168 B2

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. 7,594,168 B2

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. 7,594,168 B2 ("168 patent") [**Ex. 1001**] and

Declaration of Vijay K. Madiseti, Ph.D. in Support of  
Petition for *Inter Partes* Review of  
U.S. Patent No. 7,594,168 B2

its prosecution history. The '168 patent states on its face that it issued from an application filed on January 24, 2003, and is a continuation of application no. 09/454,061, filed on December 2, 1999. For purposes of this Declaration, I have assumed December 2, 1999 as the effective filing date for the '168 patent. I have cited to the following documents in my analysis below:

| <b>Exhibit No.</b> | <b>Description of Document</b>                                                                                                                                |
|--------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>1001</b>        | U.S. Patent No. 7,594,168 B2 to Steven R. Rempell (filed January 24, 2003, issued September 22, 2009)                                                         |
| <b>1003</b>        | U.S. Patent No. 6,052,717 to Brian Reynolds et al. (filed October 23, 1996, issued April 18, 2000)                                                            |
| <b>1004</b>        | Excerpts from Laura Lemay et al., <i>Teach Yourself Web Publishing with HTML 3.2 in a Week</i> (1996)                                                         |
| <b>1005</b>        | Excerpts from Laura Lemay et al., <i>Teach Yourself Java in 21 Days</i> (1996)                                                                                |
| <b>1006</b>        | Excerpts from Greg Perry, <i>C++ Programming 101</i> (1992)                                                                                                   |
| <b>1007</b>        | U.S. Patent No. 6,125,360 to Andrew Witkowski et al. (filed July 2, 1998, issued September 26, 2000)                                                          |
| <b>1008</b>        | Excerpts from Jason Coombs et al., <i>Setting up an Internet Site for Dummies</i> (3d ed. 1998)                                                               |
| <b>1009</b>        | Excerpts from <i>Microsoft Press Computer Dictionary</i> (3d ed. 1997)                                                                                        |
| <b>1011</b>        | Prosecution History of U.S. Patent No. 7,594,168 B2                                                                                                           |
| <b>1016</b>        | Amended Order Construing Claims (Dkt. 81) in <i>X Commerce, Inc. d/b/a Magento, Inc. v. Express Mobile, Inc.</i> , No. 17-cv-02605 (N.D. Cal. Sept. 12, 2018) |

Declaration of Vijay K. Madiseti, Ph.D. in Support of  
Petition for *Inter Partes* Review of  
U.S. Patent No. 7,594,168 B2

| <b>Exhibit No.</b> | <b>Description of Document</b>                                                                                                                               |
|--------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>1017</b>        | Oral Order (Dkt. 35) in <i>Express Mobile, Inc. v. eGrove Systems Corporation</i> , No. 1-17-cv-00703 (D. Del. Aug. 22, 2018.)                               |
| <b>1018</b>        | 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)       |
| <b>1020</b>        | 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)                      |
| <b>1021</b>        | Memorandum Opinion (Dkt. 121) in <i>Express Mobile v. GoDaddy.Com, LLC</i> , No. 1-19-cv-01937 (D. Del. June 1, 2021)                                        |
| <b>1025</b>        | Joint Claim Construction Chart (Dkt. 23) in <i>Express Mobile, Inc. v. eGrove Systems Corporation</i> , No. 1-17-cv-00703 (D. Del. April 6, 2018.)           |
| <b>1026</b>        | Joint Claim Construction Brief (Dkt. 64) in <i>Express Mobile, Inc. v. GoDaddy.Com, LLC</i> , No. 1:19-cv-01937 (D. Del. March 19, 2021)                     |
| <b>1027</b>        | Defendant's Amended List of Proposed Claim Constructions in <i>Express Mobile, Inc. v. Facebook, Inc. et al</i> , No. 6-20-cv-00803 (W.D. Tex. May 12, 2021) |

## **II. PERSON OF ORDINARY SKILL IN THE ART**

16. I understand that an assessment of claims of the '168 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

Declaration of Vijay K. Madiseti, Ph.D. in Support of  
Petition for *Inter Partes* Review of  
U.S. Patent No. 7,594,168 B2

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

17. The '168 patent describes its field of invention as generally relating to the creation of web sites. ('168, 1:12-15 (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 '168 patent makes clear, a person

of ordinary skill in the art would have had a working knowledge about various ways to implement websites and providing webpages to browsers upon request. (*See, e.g.*, '168, Abstract (“A method and apparatus for designing and building a web page” including “build tools [that] are operable to construct a single run time file and an associated database that describe, and when executed, produce the web page”).)

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 '168 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

Declaration of Vijay K. Madiseti, Ph.D. in Support of  
Petition for *Inter Partes* Review of  
U.S. Patent No. 7,594,168 B2

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 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

Declaration of Vijay K. Madiseti, Ph.D. in Support of  
Petition for *Inter Partes* Review of  
U.S. Patent No. 7,594,168 B2

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 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

Declaration of Vijay K. Madiseti, Ph.D. in Support of  
Petition for *Inter Partes* Review of  
U.S. Patent No. 7,594,168 B2

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. 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

Declaration of Vijay K. Madiseti, Ph.D. in Support of  
Petition for *Inter Partes* Review of  
U.S. Patent No. 7,594,168 B2

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 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

Declaration of Vijay K. Madiseti, Ph.D. in Support of  
Petition for *Inter Partes* Review of  
U.S. Patent No. 7,594,168 B2

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 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

Declaration of Vijay K. Madiseti, Ph.D. in Support of  
Petition for *Inter Partes* Review of  
U.S. Patent No. 7,594,168 B2

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;

(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 '168 PATENT**

##### **A. Overview of the Specification**

36. The '168 patent, entitled "Browser Based Web Site Generation Tool and Run Time Engine," is generally directed to "[a] method and apparatus for designing and building a web page." ('168, Abstract.) The '168 patent issued on September 22, 2009 from a patent filed on January 24, 2003, and claims priority to December 2, 1999.

37. The web page editor described in the '168 patent includes a "browser based build engine including build tools" that "are operable to construct a single run time file and an associated database that describe, and when executed, produce the web page." ('168, Abstract.) In particular, a 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 "runtime engine." ('168, 2:38-48, 5:47-53.) The database includes the objects of the web pages, as well as settings on "transformation[s]" and "time line[s]" for the web page objects. ('168, 35:19-30 (describing "transformation settings and technology" and stating that "[a] transformation is defined as the changing of an

Declaration of Vijay K. Madiseti, Ph.D. in Support of  
Petition for *Inter Partes* Review of  
U.S. Patent No. 7,594,168 B2

object from one state to another based on a timer control, subject to user settings.”), 36:20-35 (describing “text button and image time lines and technology” and stating that “[a] time line is an independent asynchronous process that defines the existence of a given text button or image object.”).) The specification sets forth as exemplary object transformations “the instantaneous drawing of one object state while erasing the previous object state” and “the gradual fading out of the previous object state, while, simultaneously, fading into the next object state.” (’168, 35:25-30.) The specification further describes various “settings” for transformations of “text” and “image” objects of the web pages, including “[w]hich image states should the transformation be from and into” and “[t]he delay before the next transformation.” (’168, 35:48-67.) In describing “time lines,” the specification explains that “an object time line can be created as an instance of a class, which has a threadable interface” and “its own data structures, which define the animations, and transitions associated with the time line definition.” (’168, 36:26-30.) As the specification further explains, these class files are Java classes that are compiled and “uploaded to the user’s web site” along with the database and run time engine. (’168, 44:3-30 (explaining that “all necessary JAVA class files,” the “customized run time engine file” and the “database file” are compressed into “JAR and CAB files” and “uploaded to the user’s web site,” which “completes the run generation processes”).)

38. When a web surfer views the website, the runtime engine is called by the browser to read the database and “draw” a web page, (’168, 5:53-58), which includes executing instances of Java classes (’168, 22:31-35). 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:

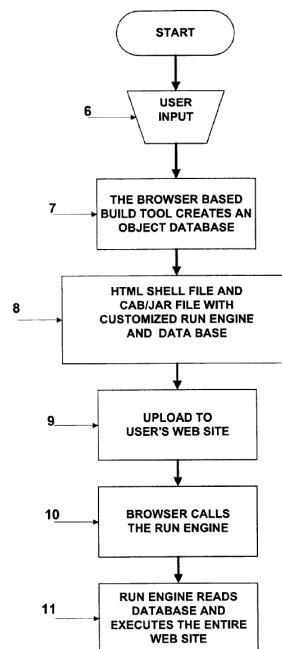


Fig. 2

(’168, Fig. 2.)

## B. The Challenged Claims

39. This Declaration addresses independent claim 1 and its dependent claims 2-4 and 6. Claim 1 recites in its entirety:

Declaration of Vijay K. Madiseti, Ph.D. in Support of  
Petition for *Inter Partes* Review of  
U.S. Patent No. 7,594,168 B2

1. A system for assembling a web site comprising:
  - [a] a server comprising a build engine configured to:
  - [b] accept user input to create a web site, the web site comprising a plurality of web pages, each web page comprising a plurality of objects, accept user input to associate a style with objects of the plurality of web pages,
  - [c] wherein each web page comprises at least one button object or at least one image object, and wherein the at least one button object or at least one image object is associated with a style that includes values defining transformations and time lines for the at least one button object or at least one image object;
  - [d] and wherein each web page is defined entirely by each of the plurality of objects comprising that web page and the style associated with the object,
  - [e] produce a database with a multidimensional array comprising the objects that comprise the web site including data defining, for each object, the object style, an object number, and an indication of the web page that each object is part of, and
  - [f] provide the database to a server accessible to web browser;
  - [g] wherein the database is produced such that a web browser with access to a runtime engine is configured to generate the web-site from the objects and style data extracted from the provided database.

('168, 64:48-65:6 (Claim 1).)

## V. CLAIM CONSTRUCTION

40. In addition to the “*Phillips*” claim construction standard, I have been asked to apply each of the following claim constructions of the terms recited in the claims. I understand that the constructions listed below were proposed and/or adopted in prior litigations involving the ’168 patent. My mapping of the prior art to the challenged claims, 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 the challenged claims obvious under each of these definitions:

| Term                      | Construction(s)                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                        |
|---------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| runtime engine<br>(cl. 1) | <ul style="list-style-type: none"><li>• 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 Express Mobile))</li><li>• 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)</li><li>• 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; EX1021, p.17)</li></ul> |
| database<br>(cl. 1)       | <ul style="list-style-type: none"><li>• an electronic information storage system offering data storage and retrieval (EX1016, p.9)</li><li>• an electronic information storage system offering data storage and retrieval and that stores information on a record-by-record basis, each</li></ul>                                                                                                                                                                                                                                                                                                      |

Declaration of Vijay K. Madiseti, Ph.D. in Support of  
Petition for *Inter Partes* Review of  
U.S. Patent No. 7,594,168 B2

| Term                              | Construction(s)                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                |
|-----------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|                                   | record divided into one or more fields (EX1018, p.19; EX1020, p.2)                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                             |
| multidimensional array<br>(cl. 1) | <ul style="list-style-type: none"> <li>a uniquely identifiable indexed set of related elements, wherein each element is addressed by a set of two or more indices, each index corresponding to a dimension of the array (EX1018, p.17; EX1020, p.1)</li> <li>an indexed set of related elements, wherein each element is addressed by a set of two or more indices (EX1018, p.16 (proposed by Express Mobile))</li> </ul>                                                                                                                                                                      |
| settings<br>(cl. 4)               | <ul style="list-style-type: none"> <li>attributes of an object available for selection (EX1020, p.1)</li> </ul>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                |
| transformations<br>(cl. 1)        | <ul style="list-style-type: none"> <li>the changing of an object from one state to another based on a timer control, subject to user settings (EX1018, p.7; EX1016, p.13; EX1021, p.10)</li> </ul>                                                                                                                                                                                                                                                                                                                                                                                             |
| time lines<br>(cls. 1, 3)         | <ul style="list-style-type: none"> <li>a sequence of changes that define the attributes of a text button or image object as the changes occur (EX1016, p.13)</li> <li>independent process that defines one or more values for an object, including its appearance, animation, speed, and/or resolution over time (EX1016, p.13 (proposed by Express Mobile))</li> <li>independent asynchronous process that defines the values for an object, including its appearance, animation, speed, and/or resolution at one or more defined times (EX1025, p.5 (proposed by Express Mobile))</li> </ul> |
| objects<br>(cls. 1-4)             | <ul style="list-style-type: none"> <li>variable comprising data and routines treated as a discrete entity (EX1027, p.4 (proposed by Facebook))</li> </ul>                                                                                                                                                                                                                                                                                                                                                                                                                                      |

Declaration of Vijay K. Madiseti, Ph.D. in Support of  
Petition for *Inter Partes* Review of  
U.S. Patent No. 7,594,168 B2

| Term                                           | Construction(s)                                                                                                                                                                                                                                                                                                                                      |
|------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| style<br>(cls. 1-4)                            | <ul style="list-style-type: none"> <li>attribute that defines the visual appearance of an object (EX1027, p.5 (proposed by Facebook))</li> </ul>                                                                                                                                                                                                     |
| button<br>(cls. 1-3)                           | <ul style="list-style-type: none"> <li>a graphic object that, when activated through user interaction, performs a specified function (EX1016, p.11)</li> </ul>                                                                                                                                                                                       |
| child/child object/child button<br>(cls. 2, 3) | <ul style="list-style-type: none"> <li>an element with one or more properties that depends on another element (EX1026, pp.35-36 (proposed by Express Mobile))</li> <li>element/object/button that is spawned by another element/object/button (i.e., the parent element/object/button) (EX1026, pp.35-36 (proposed by defendant GoDaddy))</li> </ul> |

## 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 '168 patent are rendered obvious based on the following prior art:

| Ground | Basis for Challenge Under §103(a)                                                                            | Claim(s)  |
|--------|--------------------------------------------------------------------------------------------------------------|-----------|
| 1      | Reynolds (Ex. 1003), Lemay Web (Ex. 1004), Lemay Java (Ex. 1005), Perry (Ex. 1006), and Witkowski (Ex. 1007) | 1-4 and 6 |
| 2      | Reynolds, Lemay Web, Lemay Java, Perry, Witkowski, and Coombs (Ex. 1008)                                     | 1-4 and 6 |

42. I am informed by counsel that each reference listed above qualifies as prior art to the challenged claims because each reference was filed and/or published

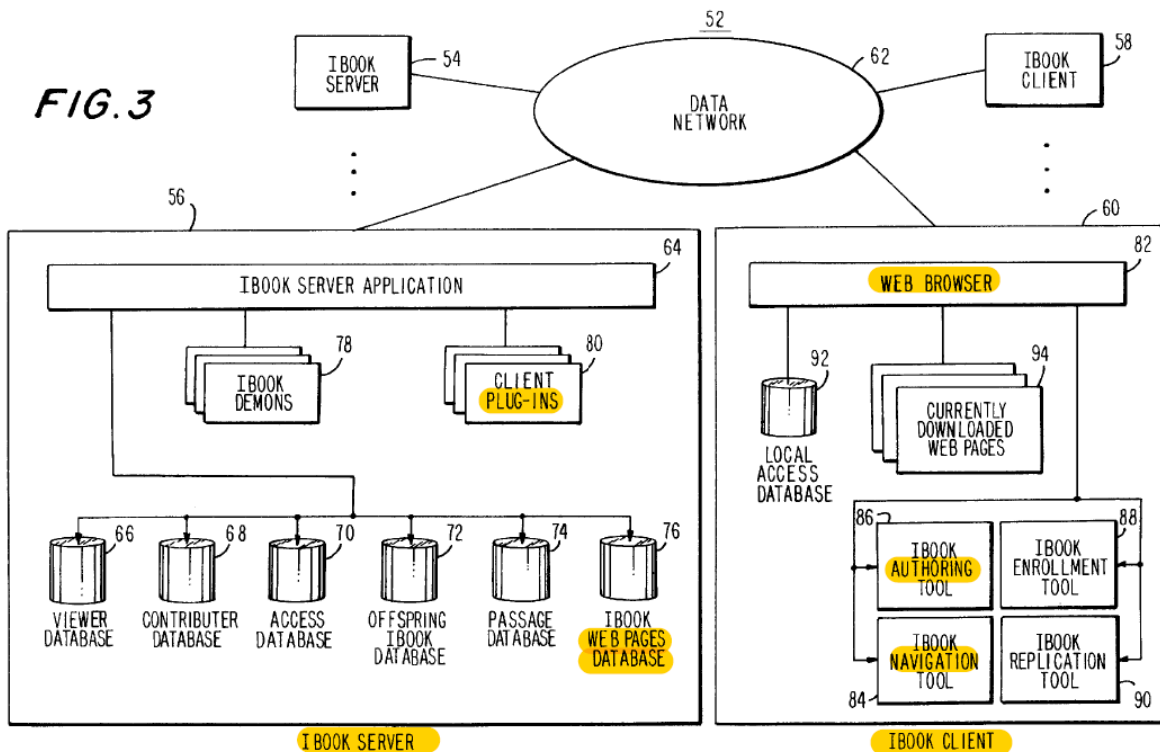
before the earliest claimed priority date for the '168 patent.

**A. Brief Summary of Prior Art**

**1. Reynolds [Ex. 1003]**

43. U.S. Patent No. 6,052,717 to Reynolds et al. is entitled “Interactive Web Book System.” I understand from counsel that Reynolds qualifies as prior art to the '168 patent because it was filed before the effective filing date of the '168 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, 1:24:31; *see also id.*, Abstract.) Reynolds’ system for creating ibooks is depicted in Fig. 3, reproduced and annotated below.



(Reynolds, Fig. 3 (annotation added).)

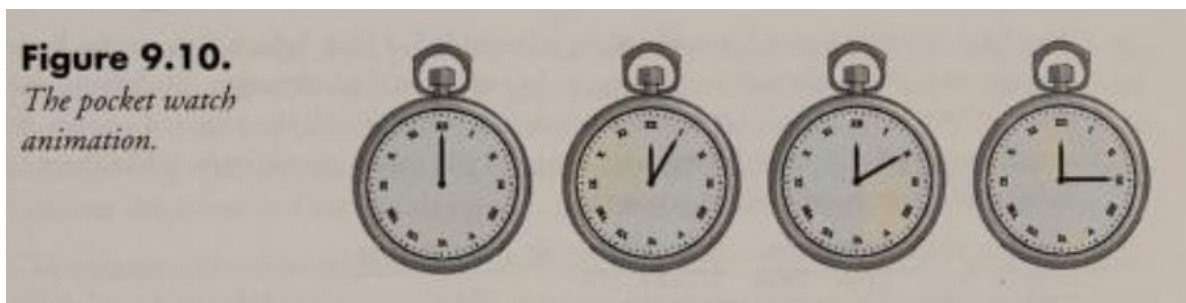
45. Reynolds explains that the material for ibook web pages are stored in an “ibook web pages database 76” at an ibook web server. To edit or view these pages, a user downloads plug-ins 80 from the ibook server that extend the capabilities of the user’s web browser 82. (Reynolds, 5:27-31.) When downloaded, 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 accessing the ibook server to download and display the

content from the database. (Reynolds, 11:24-39.)

## 2. Lemay Web [Ex. 1004]

46. “Teach Yourself Web Publishing with HTML 3.2 in a Week,” by Lemay is a reference book that describes how use HTML to publish web pages. I am informed that Lemay Web qualifies as prior art to the ’168 patent because it was published at least one year before the earliest effective filing date for the ’168 patent and also published before the invention described in the ’168 patent.

47. I rely on Lemay Web primarily for claim 1, which recites, among other things, that each web page comprises “**at least one button object or at least one image object,**” which is associated with a “**style that includes values defining transformations and time lines . . .**” I rely on Lemay Web in combination with Reynolds to show an example of an object in the form of a Java applet that can be placed on a web page to provide an animated image:



(Lemay Web, Fig. 9.10, p.255.) Figure 9.10 above shows the first four frames of an animated image of a pocket watch, with each frame showing the minute hand pointing to a different position on the dial—in these four examples at 12:00, 12:05,

Declaration of Vijay K. Madiseti, Ph.D. in Support of  
Petition for *Inter Partes* Review of  
U.S. Patent No. 7,594,168 B2

12:10, and 12:15. Lemay Web explains that this example includes 12 image files for the pocket watch (named T1.gif, T2.gif all the way to T12.gif) representing 12 different positions of the second hand. (*Id.*)

48. Adding this image animation to a web page would have been straightforward. Lemay Web explains that the creators of Java (Sun Microsystems) provided a pre-built Java applet named “Animator.class” that allows a web page creator to specify a series of images that will be displayed in sequence on a web page. (Lemay Web, pp.254-55.) To insert the animated image into a page, the creator simply has to insert a few lines of HTML into the page:

```
<APPLET CODE="Animator.class" WIDTH=100 HEIGHT=100>  
<PARAM NAME="STARTIMAGE" VALUE="1">  
<PARAM NAME="ENDIMAGE" VALUE="12">  
<PARAM NAME="PAUSE" VALUE="1000">  
<PARAM NAME="REPEAT" VALUE="TRUE">  
</APPLET>
```

(Lemay Web, p.256.) The HTML code above contains <APPLET> and <PARAM> tags, which are standard tags for inserting a Java applet onto a web page. (Lemay Web, p.255.) These tags would have been well-known to any person of ordinary skill in the art, or anyone else familiar with creating web pages using HTML and Java applets. Lemay Web explains the first <APPLET> tag specifies the name of the Java applet (*i.e.* “Animator.class”), and in this case, the WIDTH and HEIGHT

for the applet on the page (*i.e.* 100-by-100 pixels). (Lemay Web, p.255.)

49. The HTML code then includes four <PARAM> tags, each tag having a NAME and VALUE attribute. (Lemay Web, p.256.) The first and second <PARAM> tags specify the “STARTIMAGE” and “ENDIMAGE,” respectively, indicating that there are 12 images (from 1-12) in the animation. The next <PARAM> tag defines a “PAUSE” value, which specifies how long (in milliseconds) the applet will pause between images in the animation. (Lemay Web, p.256.) Lemay explains that “[y]ou can experiment with the pause between frames until you get an animation you like (here I picked 1000 milliseconds, or an even second) . . . .” (Lemay Web, p.256.) The last <PARAM> tag sets the “REPEAT” parameter to “TRUE,” which tells the applet to loop the animation repeatedly. (Lemay Web, p.256.)

50. Lemay Web makes clear that the <APPLET> and <PARAM> tags shown above provide “only a couple of the Animator applet’s parameters,” as there are other parameters that a developer could specify. (Lemay Web, p.257.) Additional parameters for the Animator include, for example, “parameters that let you . . . add a background to the animation or a soundtrack, and control the order that frames are displayed.” (Lemay Web, p.257.)

51. As I will explain in more detail in my discussion of claim 1 below,

Lemay Web renders the claimed objects and styles obvious in combination with the other prior art cited in this Declaration.

### **3. Lemay Java [Ex. 1005]**

52. “Teach Yourself Java in 21 Days” by Lemay is a reference book describing the Java programming language and how to use it. I am informed that Lemay Java qualifies as prior art to the ’168 patent because it was published at least one year before the earliest effective filing date for the ’168 patent and also published before the invention described in the ’168 patent.

53. My Declaration cites Lemay Java in combination with Lemay Web (mentioned above). Lemay Java provides more detail about the workings of Java applets, including their ability to operate independently through multithreading.

54. Lemay Java teaches how to write “applets” in Java, which are “dynamic and interactive program[s] that can run inside a Web page displayed by a Java-capable browser.” (Lemay Java, 4.) An applet can “run all by itself without interfering with any other part of the system,” using a standard Java feature referred to as “multithreading” which “enables several different execution threads to run at the same time inside the same program, in parallel, without interfering with each other.” (Lemay Java, 178.) Thus, “[u]sing threads, you can have lots of applets running at once on the same page,” i.e., they will “run independently.” (Lemay Java,

178.) As Lemay explains, “[e]ven if you don’t have lots of applets, using threads in your applets is good Java programming practice. The general rule of thumb for well-behaved applets: whenever you have any bit of processing that is likely to continue for a long time (such as an animation loop, or a bit of code that takes a long time to execute), put it in a thread.” (Lemay Java, 178.)

#### 4. Perry [Ex. 1006]

55. Greg Perry, *C++ Programming 101* (1992), is a reference book that discusses fundamental computer programming techniques, including the creation of data structures such as multidimensional arrays to organize data. I am informed that Perry qualifies as prior art to the ’168 patent because it was published at least one year before the earliest effective filing date for the ’168 patent and also published before the invention described in the ’168 patent.

56. My Declaration relies on Perry for the limitation of claim 1 reciting the ability to “**produce a database with a multidimensional array,**” which includes certain information. Perry confirms that the ’168 patent inventors did not invent the concept of a “**multidimensional array**,” and further confirms that the recitation of such an array in claim 1 adds nothing of patentable significance.

57. The term “**array**” in computer science generally refers to a data structure consisting of a number of elements, each element identified by an index

that identifies the location of the element in the array. A simple example of an array is a string of characters—such as the five-character string “hello” containing five elements (textual characters). Each element of this array can be accessed by using an index value that identifies a particular position within the array from zero to four; an index value of zero corresponds to the letter “h” at the front of the string and an index value of four corresponds to the concluding “o” at the end. This is an example of a *one-dimensional* array because only one index value (from zero to four) is needed to access each element in the array. Arrays are basic knowledge of ordinarily skilled artisans and even beginning computer programmers.

58. A multidimensional array, as its name implies, requires at least *two* indices to access each element in the array. As explained in Perry, “[m]ultidimensional arrays, sometimes called *tables* or *matrices*, have at least two dimensions (rows and columns).” (Perry, Ex. 1006, p.455 (italics in original).) “Whereas a single-dimensional array is a list of values, a multidimensional array simulates a table of values, or multiple tables of values. The most commonly used table is a two-dimensional table . . . .” (Perry, p.456.) Table 27.1 in Perry below provides a straightforward example of two-dimensional array for “a football team [that] wanted to track its players’ total yardage. The team has eight games scheduled, and there are six rushing players on the team.” (Perry, p.456)

**Table 27.1. A football team's yardage record.**

| <i>Player Name</i> | <i>Game 1</i> | <i>Game 2</i> | <i>Game 3</i> | <i>Game 4</i> | <i>Game 5</i> | <i>Game 6</i> | <i>Game 7</i> | <i>Game 8</i> |
|--------------------|---------------|---------------|---------------|---------------|---------------|---------------|---------------|---------------|
| Downing            | 1             | 0             | 2             | 1             | 0             | 0             | 0             | 0             |
| Edwards            | 0             | 3             | 6             | 4             | 6             | 4             | 5             | 3             |
| Franks             | 2             | 2             | 3             | 2             | 1             | 0             | 2             | 3             |
| Grady              | 1             | 3             | 2             | 0             | 1             | 5             | 2             | 1             |
| Howard             | 3             | 1             | 1             | 1             | 2             | 0             | 1             | 0             |
| Williams           | 2             | 3             | 1             | 0             | 1             | 2             | 1             | 2             |

(Perry, Table 27.1, p.456.)

59. Perry explains that the Table 27.1 “has rows (the first dimension) and columns (the second dimension). Therefore, this is a two-dimensional table with six rows and eight columns.” (Perry, p.456.) “Each row has a player’s name, and each column has a game number, but these are not part of the actual data. The data consists only of 48 values (six rows by eight columns).” (Perry, p.456.) Table 27.1 provides an example of a “multidimensional array” because it is a collection of elements (in this case yardage values), each element identified by two indices – the player name (row) and the game number (column).

## **5. Witkowski [Ex. 1007]**

60. U.S. Patent No. 6,125,360 to Andrew Witkowski et al., relates to database technologies. My Declaration cites Witkowski for its brief description of

relational databases that I cite in my discussion of claim 1[e] below, and for its exemplary disclosure of relational database tables used to create multidimensional arrays. I am informed that Witkowski qualifies as prior art to the '168 patent because it issued from an application filed before the effective priority date of the patent.

61. 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.)<sup>1</sup> Witkowski also discloses the ability to use these rows and

---

<sup>1</sup> Unless otherwise noted, all underlining and emphasis in quotations from the prior art was added for emphasis.

columns to store information in the form of “a database with a multidimensional array,” as recited in claim 1[e], under the definitions in **Part V**. For example, Figure 4A of Witkowski discloses two exemplary database tables (**TABLE X** and **TABLE Y**), each having rows and columns holding values:

*Fig. 4A*

| TABLE X  |     |     | BASE TABLES AT TIME T1 |          |     | TABLE Y |  |  |
|----------|-----|-----|------------------------|----------|-----|---------|--|--|
| x.row id | X.a | X.b |                        | Y.row id | Y.a | Y.b     |  |  |
| X1       | 1   | 1   |                        | Y1       | 1   | 0       |  |  |
| X2       | 2   | 2   |                        | Y2       | 2   | 1       |  |  |
| X3       | 3   | 3   |                        | Y3       | 4   | 1       |  |  |
| X4       | 4   | 4   |                        | Y3       | 2   | 8       |  |  |
| X5       | 9   | 0   |                        | Yy       | Z   | 8       |  |  |

(Witkowski, Fig. 4A.) “Referring to FIG. 4a,” as explained in Witkowski, “table X includes two columns x.a and x.b and a pseudo-column ‘X.rowid’. The pseudo-column contains a unique identifier for each row in table X.” (Witkowski, 15:54-56.) “Similar to table X, table Y includes two columns y.a and y.b and a pseudo-column Y.rowid.” (Witkowski, 15:59-60.) Witkowski describes various techniques

for “joining” these two tables, but those techniques are not pertinent to the limited purpose for which I have cited Witkowski, *i.e.*, to simply show exemplary database tables organizing information into rows and columns, each table forming a multidimensional array in which each element can be located by using two indices (the identifier of the row id and column identifier). As explained below, it would have been obvious to implement the database in Reynolds as a relational database with a multidimensional array, that would satisfy claim 1[e].

## **6. Coombs [Ex. 1008]**

62. “Setting up an Internet Site for Dummies,” by Coombs et al. 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 ’168 patent because it was published at least one year before the earliest effective filing date for the ’168 patent and also published before the invention described in the ’168 patent.

63. 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.)

**B. Ground 1: Claims 1-4, and 6 are obvious over Reynolds in view of Lemay Web, Lemay Java, Perry and Witkowski**

**1. Independent Claim 1**

64. 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:

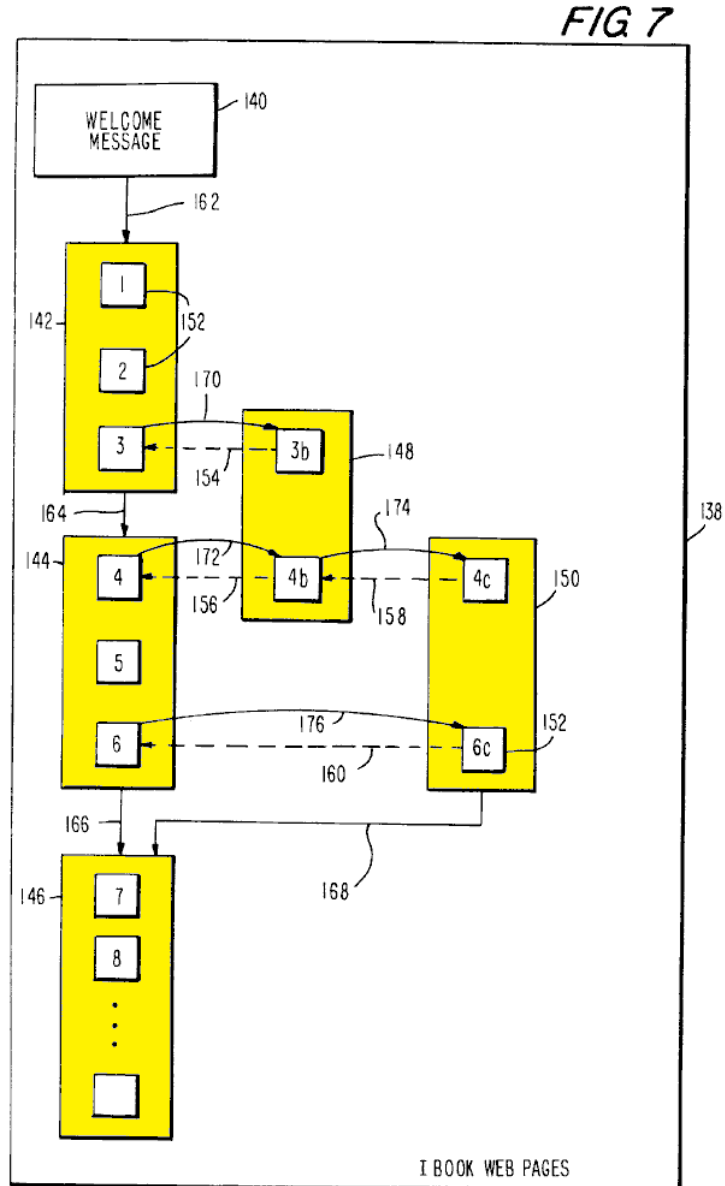
1. A system for assembling a web site comprising:
  - [a] a server comprising a build engine configured to:
  - [b] accept user input to create a web site, the web site comprising a plurality of web pages, each web page comprising a plurality of objects,
  - [c] accept user input to associate a style with objects of the plurality of web pages, wherein each web page comprises at least one button object or at least one image object, and wherein the at least one button object or at least one image object is associated with a style that includes values defining transformations and time lines for the at least one button object or at least one image object;
  - [d] and wherein each web page is defined entirely by each of the plurality of objects comprising that web page and the style associated with the object,
  - [e] produce a database with a multidimensional array comprising the objects that comprise the web site including data defining, for

each object, the object style, an object number, and an indication  
of the web page that each object is part of, and

- [f] provide the database to a server accessible to web browser;
- [g] wherein the database is produced such that a web browser with  
access to a runtime engine is configured to generate the web-site  
from the objects and style data extracted from the provided  
database.

(’168, 64:48-65:6 (Claim 1).) As explained in detail below, the prior art discloses  
and renders obvious each limitation of claim 1.

65. The preamble of claim 1 recites, “[a] **system for assembling a website comprising . . .**” To the extent the preamble is limiting, Reynolds describes a system for assembling a website. The “**website**” in Reynolds takes the form of an “interactive Web book,” or “ibook,” which contains a series of ibook Web pages. (Reynolds, Abstract; *see also id.*, 1:25-27 (“This and other objects of the invention are accomplished in accordance with the principles of the present invention by providing an interactive Web book (‘ibook’) system.”), 1:30-31 (“Material in an ibook is preferably organized in the form of Web pages.”).) Figure 7 shows an exemplary ibook containing a number of ibook web pages:



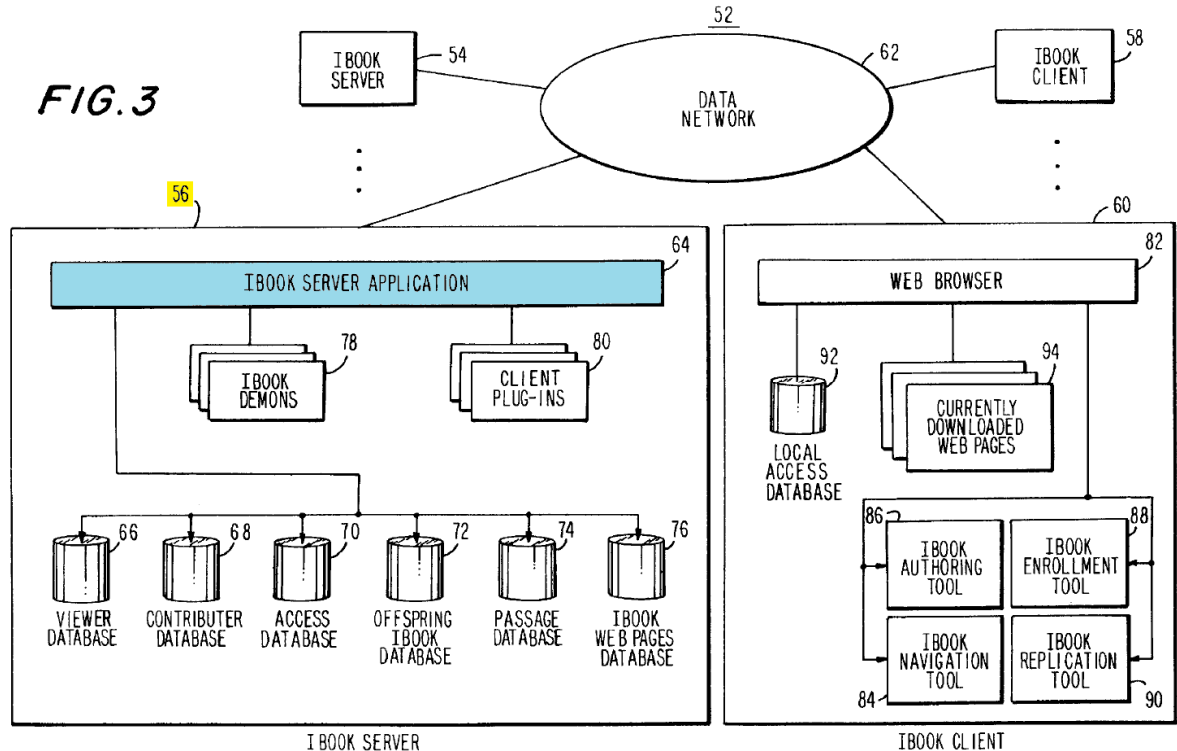
(Reynolds, Fig. 7 (highlighting added).) The exemplary ibook in Figure 7 shows five web pages **142, 144, 146, 148, 150** (shown in yellow highlighting). (Reynolds, 7:57-67.) Figure 7 further confirms that the ibook system in Reynolds “**assembl[es]**” a website, *e.g.*, an assembled ibook as shown for example in Figure 7. The web pages shown in Figure 7 are interconnected via a series of hyperlinks,

e.g., sequence link **164** links web page **142** to **144**, and sequence link **166** links web pages **146** and **148**, and so forth. (Reynolds, 8:53-60.)

66. Each web page in an ibook has one or more “passages” **152** (shown as numbered squares within the yellow highlighted web pages), which provide the content for ibook web pages. (Reynolds, 7:66-67, 1:34-38 (“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.”).) As explained in more detail below, “passages” are downloaded from a server and placed onto the web pages by software running on the web browser on the client device. (Reynolds, 11:24-40.) Reynolds therefore discloses a system for “**assembling a website**,” as recited in the preamble. I will provide additional details about the Reynolds website assembly system in my discussion of the claim limitation that follows.

**(a) “a server comprising a build engine configured to:”  
(Claim 1[a])**

67. The “**server**” takes in Reynolds the form of ibook server **56**, as discussed below, which in turn comprises a “**build engine**” in the form of an ibook server application **64**. Figure 3 shows both of these components:



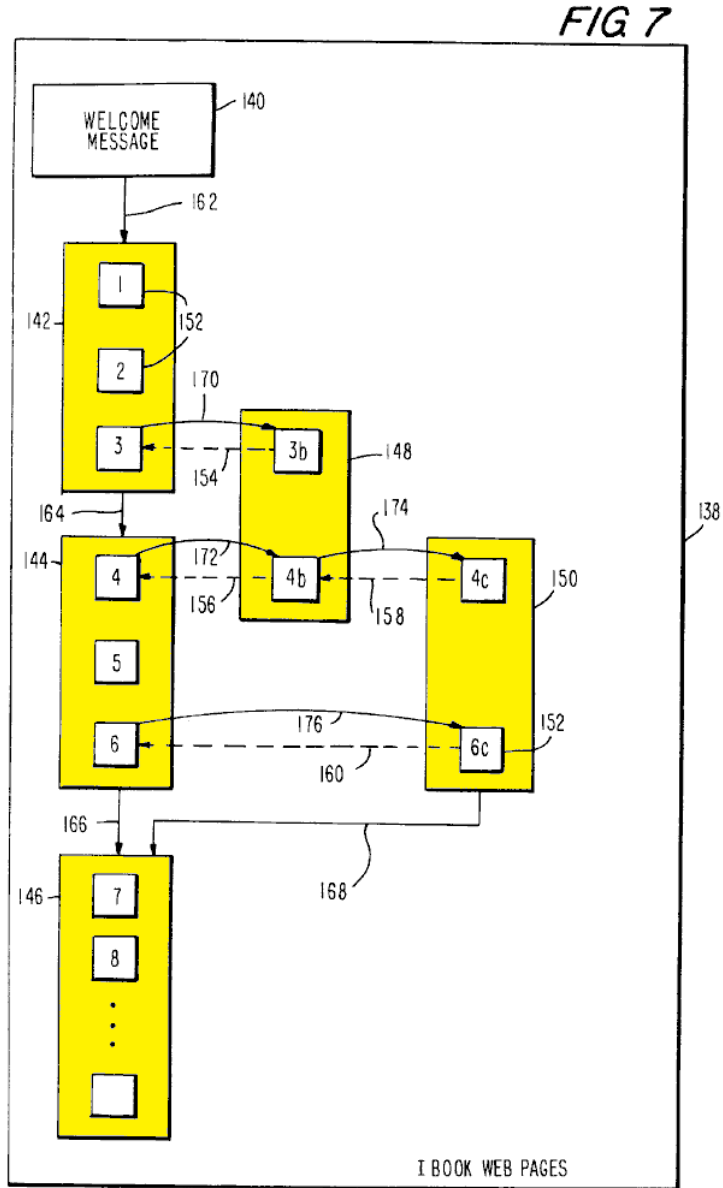
(Reynolds, Fig. 3 (highlighting added).) Figure 3 above shows ibook server **56** which includes an ibook server application **64**, which “directs the overall operation of ibook server **56**.” (Reynolds, 5:18-20.) The server application (“**build engine**”) is configured to perform the functions, described below, recited later in the claim.

(b) “accept user input to create a web site, the web site comprising a plurality of web pages, each web page comprising a plurality of objects,” (Claim 1[b])

68. The build engine in Reynolds is configured to accept user input to create a web site. As explained, an ibook web site “**compris[es] a plurality of web pages,**” *i.e.* ibook web pages. (Reynolds, *e.g.*, 1:29-30 (“Material in an ibook is preferably organized in the form of Web pages.”), 7:36-37.) Each of these web pages

comprises “**a plurality of objects**,” which as explained for the preamble, take the form of passages **152** that contain content for ibook web pages. (Reynolds, 1:34-38 (“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.”).) Reynolds explains that passages **152** can include not just text but also “graphic images, animation sequences, portions of a sound track, still video images, or video clips. If an ibook contains executable material, passages **152** can be modules of code.” (Reynolds, 8:4-11; *see also id.*, 9:52-54 (“When passages **152** are executable modules, navigation tool **84** presents the passages by downloading and executing the passages.”).)

69. As noted above, Figure 7 shows an exemplary ibook comprising a “**plurality of web pages**,” *i.e.* ibook web pages, “**each web page comprising a plurality of objects**,” *i.e.* passages within each web page:



(Reynolds, Fig. 7 (highlighting added).) As noted previously, the exemplary ibook in Figure 7 shows five web pages **142, 144, 146, 148, 150** (shown in yellow highlighting). (Reynolds, 7:57-67.) As also noted previously, each web page has one or more “passages” **152** (shown as numbered squares within the yellow highlighted web pages), which provide the content for ibook web pages. (Reynolds,

Declaration of Vijay K. Madiseti, Ph.D. in Support of  
Petition for *Inter Partes* Review of  
U.S. Patent No. 7,594,168 B2

7:66-67, 1:34-38 (“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.”).)

70. For purposes of claim 1**[b]**, the “**plurality of web pages**” corresponds in Reynolds to at least two ibook web pages containing passages **152**, such as the ones shown in yellow highlighting from Figure 7 above. Claim 1**[b]** recites a “web site comprising a plurality of web pages,” and I am informed by counsel that under the patent laws, the word “comprising” means that the “web site” can include additional web pages beyond the claimed “plurality of web pages.” It is therefore not necessary to include *every* ibook web page to satisfy the claim, but rather, identify at least two web pages that are part of the ibook web site. Because the claim language does not exclude the existence of web pages in addition to those identified as “a plurality of web pages,” it is unnecessary to include welcome message page **140** that provides optional summary and sponsorship information and an ability to enroll in an ibook. (Reynolds, 7:41-56.)

71. Passages in Reynolds also qualify as “**objects**,” as claimed. As explained in **Part V**, one of the definitions provided for “object” in a previous litigation was “*variable comprising data and routines treated as a discrete entity*.” It is unnecessary for me to opine on whether or not I agree with this definition

Declaration of Vijay K. Madiseti, Ph.D. in Support of  
Petition for *Inter Partes* Review of  
U.S. Patent No. 7,594,168 B2

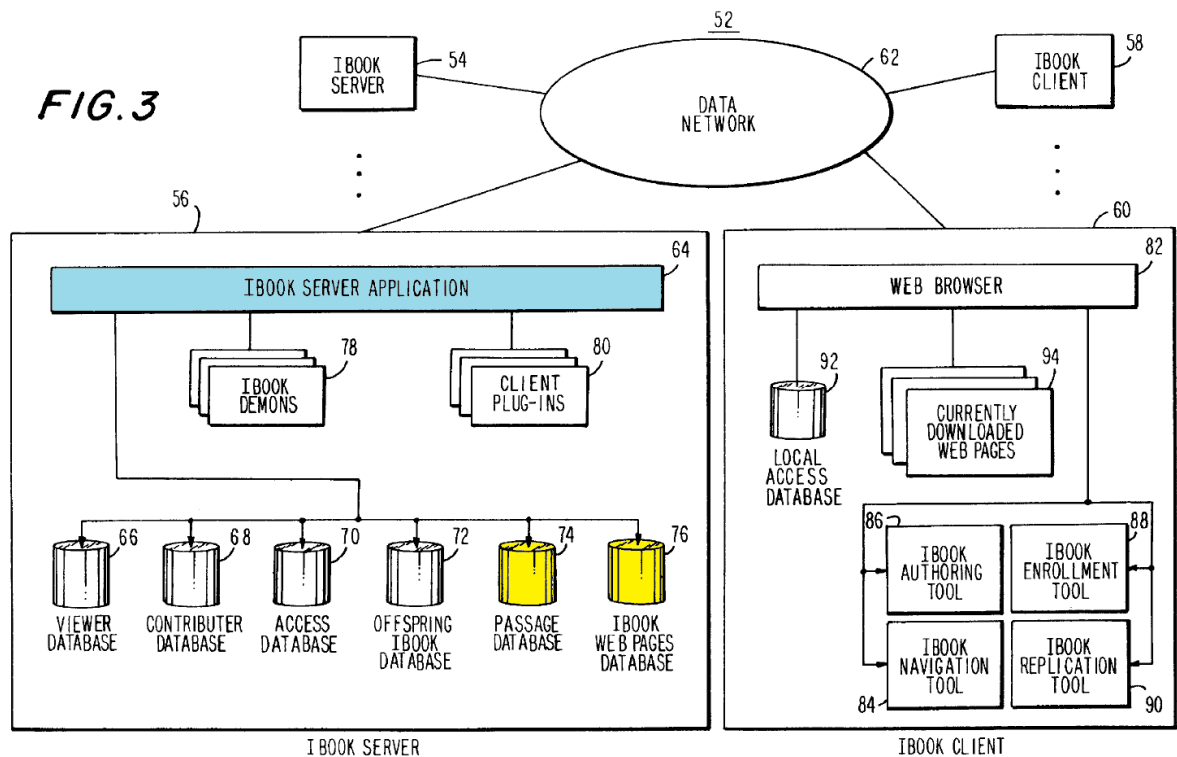
because, even if adopted, Reynolds readily discloses it. Passages in Reynolds comprise a “*variable*” because they are not fixed information; as explained below, the passages can contain data that is selected by the user through authoring tool 86. Reynolds also confirms that passages can contain “*data and routines.*” The term “routine” in computer science generally refers to a sequence of software code or instructions that can be invoked to perform a particular function. As shown above, passages can not only contain data, but executable software or code modules. (Reynolds, *e.g.*, 8:10-11, 9:52-54.) A person of ordinary skill in the art would have understood that executable software modules described in Reynolds would include (at a minimum) software routines and data used to carry out their functions. Under the combination of Reynolds with Lemay Web and Lemay Java, as explained for claim 1[c] below, ibook passages can also include Java applets that contain code and image data for displaying an animated image object. Finally, each passage in Reynolds is “*treated as a discrete entity.*” As explained below, each passage is accessed and retrieved separately from the server when viewing the web page. (Reynolds, 11:24-40.) Reynolds further discloses the ability to keep track of viewer access on a passage-by-passage basis. (Reynolds, 11:7-11, 11:43-55.)

72. Reynolds therefore discloses that “**the web site compris[es] a plurality of web pages, each web page comprising a plurality of objects,**” as

claimed. I will provide further detail about these passages in my discussion of subsequent limitations below.

73. The web server application **64** in Reynolds (the “**build engine**”) is also configured to “**accept user input to create a web site**,” as claimed. Reynolds discloses a series of tools including authoring tool **86**, a plug-in software program downloaded from the server that executes on the client device. (Reynolds, 7:2-5 (“To create or modify material in an ibook, a contributor uses ibook authoring tool **86** (FIG.3). As shown in FIG. 6, ibook authoring tool **86** includes derivation tool **132**, linking tool **134**, and editing tool **136**.”), 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**.”).) Authoring tool **86** is used to create the passages of an ibook discussed above, and “[t]he functions provided by authoring tool **86** may differ depending on the content of a particular ibook passage.” (Reynolds, *e.g.*, 7:23-24.) For example, authoring tool **86** includes editing tool **136** which “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, 7:25-28.)

74. Although authoring tool **86** runs on the client computer **60**, Reynolds makes clear that the “**build engine**” (*i.e.* ibook server application **64**) on the server “**accept[s] user input**” for the passages used to create an ibook. This takes place when the server of Reynolds receives the user input provided through authoring tool **86** and stores it in databases at the server as shown in Figure 3:



(Reynolds, Fig. 3 (highlighting added).) As shown, ibook server application **64** (“**build engine**”) controls a number of databases, including passage database **74** and ibook web pages database **76** that store information relating to an ibook. (Reynolds, *e.g.*, 5:21-23 (“In particular, ibook server application **64** controls databases **66**, **68**,

70, 72, 74, and 76, in which various ibook information is stored.”.)

75. Reynolds further explains that “during the creation of passages **152** authoring tool **86** and ibook server application 64 cause information to be stored in passage database 74 that identifies which Web page **138** contains each passage **152**.” (Reynolds, 9:27-30.) Reynolds also explains that “[t]he content of the ibook Web pages is preferably stored in ibook Web pages database **76** (FIG. 3).” (Reynolds, 7:37-39.) It would have been apparent and obvious to a person of ordinary skill in the art that ibook server application **64** accepts user input provided through authoring tool **86** to create ibook passages—otherwise ibook server application **64** could not store ibook information in its databases to reflect user selections made using authoring tool **86**. Reynolds therefore discloses and renders obvious a “**build engine**” that is configured to “**accept user input to create a web site.**”

- (c) “**accept user input to associate a style with objects of the plurality of web pages, wherein each web page comprises at least one button object or at least one image object, and wherein the at least one button object or at least one image object is associated with a style that includes values defining transformations and time lines for the at least one button object or at least one image object;**” (Claim 1[c])

76. Reynolds renders this limitation obvious in view of Laura Lemay, *Teach Yourself Web Publishing with HTML 3.2 in a Week* (1996) (**Ex. 1004**), referred to throughout this Declaration as “**Lemay Web.**” As will be explained

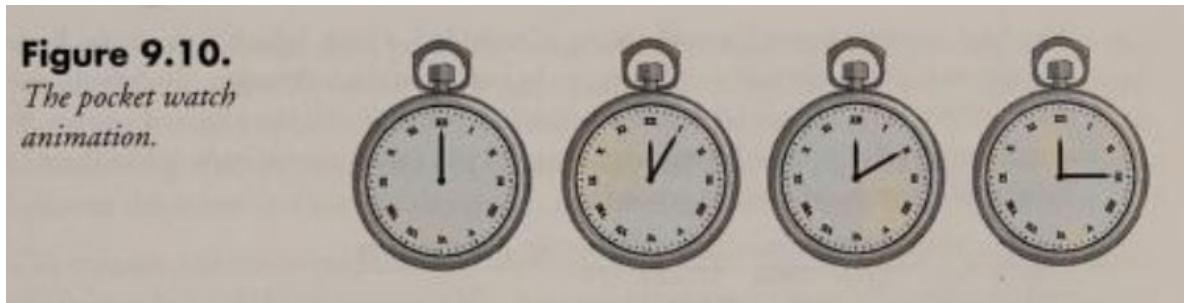
below, Reynolds and Lemay Web render obvious the ability of the build engine to associate a style with objects on an ibook web page, such as the style associated with a Java animation object (“**at least one button object or at least one image object**”).

77. As noted previously, Reynolds explains that passages **152** of ibook web pages (“**objects of the plurality of web pages**”) can include not just text but also graphic images, animation sequences, and executable code modules. (Reynolds, 8:4-11, 9:52-54.) Reynolds explains that “[t]he functions provided by authoring tool **86** may differ depending on the content of a particular ibook passage.” (Reynolds, 7:23-24.) For example, “editing tool **136** [part of authoring tool **86**] 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, 7:25-28.)

78. Reynolds thus contemplates that authoring tool **86** will have functionality allowing the user to place content on an ibook web page, but Reynolds does not provide details regarding the specific user inputs provided through authoring tool **86**. I have therefore cited to Lemay Web which describes how to add “**at least one button object or at least one image object**,” to a web page, and specify a style for the object as recited in the claim.

79. Lemay Web discloses an example of a simple Java applet that can be

placed on a web page to provide an animated image of a pocket watch:



(Lemay Web, Fig. 9.10, p.255.) Figure 9.10 above shows the first four frames of an animated image of a pocket watch, with each frame showing the minute hand pointing to a different position on the dial—in these four examples at 12:00, 12:05, 12:10, and 12:15. Lemay Web explains that this example includes 12 image files for the pocket watch (named T1.gif, T2.gif all the way to T12.gif) representing 12 different positions of the second hand. (*Id.*)

80. Adding this image animation to a web page would have been exceedingly straightforward. Lemay Web explains that the creators of Java (Sun Microsystems) provided a pre-built Java applet named “Animator.class” that allows a web page creator to specify a series of images that will be displayed in sequence on a web page. (Lemay Web, pp.254-55.) To insert the animated image into a page, the creator simply has to insert a few lines of HTML into the page:

```
<APPLET CODE="Animator.class" WIDTH=100 HEIGHT=100>  
<PARAM NAME="STARTIMAGE" VALUE="1">  
<PARAM NAME="ENDIMAGE" VALUE="12">  
<PARAM NAME="PAUSE" VALUE="1000">  
<PARAM NAME="REPEAT" VALUE="TRUE">  
</APPLET>
```

(Lemay Web, p.256.) The HTML code above contains <APPLET> and <PARAM> tags, which are standard tags for inserting a Java applet onto a web page. (Lemay Web, p.255.) These tags would have been well-known to any person of ordinary skill in the art, or anyone else familiar with creating web pages using HTML and Java applets. Lemay Web explains the first <APPLET> tag specifies the name of the Java applet (*i.e.* “Animator.class”), and in this case, the WIDTH and HEIGHT for the applet on the page (*i.e.* 100-by-100 pixels). (Lemay Web, p.255.)

81. The HTML code then includes four <PARAM> tags, each tag having a NAME and VALUE attribute. (Lemay Web, p.256.) The first and second <PARAM> tags specify the “STARTIMAGE” and “ENDIMAGE,” respectively, indicating that there are 12 images (from 1-12) in the animation. The next <PARAM> tag defines a “PAUSE” value, which specifies how long (in milliseconds) the applet will pause between images in the animation. (Lemay Web, p.256.) Lemay explains that “[y]ou can experiment with the pause between frames until you get an animation you like (here I picked 1000 milliseconds, or an even

second) . . . .” (Lemay Web, p.256.) The last <PARAM> tag sets the “REPEAT” parameter to “TRUE,” which tells the applet to loop the animation repeatedly. (Lemay Web, p.256.)

82. Lemay Web makes clear that the <APPLET> and <PARAM> tags shown above provide “only a couple of the Animator applet’s parameters,” as there are other parameters that a developer could specify. (Lemay Web, p.257.) Additional parameters for the Animator include, for example, “parameters that let you . . . add a background to the animation or a soundtrack, and control the order that frames are displayed.” (Lemay Web, p.257.)

83. Lemay Web thus discloses “**objects of the plurality of web pages,**” in the form of image animations that include both graphical images as well as executable code (*i.e.* the Java animation applet) to display the images in sequence. Lemay Web further discloses the ability to “**accept user input to associate a style**” with these objects. Lemay Web discloses many examples of attributes that define the visual appearance of these objects, including the width and height on the page, the sequence of the images, a “PAUSE” value indicating how long each image is displayed before advancing to the next image, and a “REPEAT” value indicating whether or not the image animation will occur once or cycle in a loop.

84. Lemay Web also discloses that “**each web page comprises at least one**

**button object or at least one image object,”** as claimed. The animated image object in Lemay Web, in fact, qualifies as both. It obviously qualifies as an “**image object**” because, as noted, it includes a number of graphical images for display on the web page. It also qualifies as a “**button object**” because, as explained in Lemay Web, “clicking on the animation will start and stop it.” (Lemay Web, p.256.) The animated image, in other words, is a button object in that performs the function of starting and stopping the animation when activated by the viewing user’s interaction through a mouse click.

85. And regardless of whether an animated image is viewed as an image object, a button object, or both, it is “**associated with a style that includes values defining transformations and time lines for the at least one button object or at least one image object,”** as claimed. With respect to “**transformations,”** as explained, the <PARAM> tags that define the style as specified above define how the image object changes from one state to another. The “STARTIMAGE” and “ENDIMAGE” tags, as mentioned, specify the images that define where the animation begins and ends; in the pocket watch example, how the appearance of the pocket watch will be transformed as the animation advances. The animation is also subject to a user-selectable timer control because, as mentioned, the “PAUSE” attribute specifies the amount of time before advancing to the next image.

86. For the same reasons, the <PARAM> tags that define the style also define “**time lines**” for the image or button object. As explained in **Part V**, three different definitions have been provided in prior litigations, the simpler ones being “*a sequence of changes that define the attributes of a text button or image object as the changes occur,*” and “*independent process that defines one or more values for an object, including its appearance, animation, speed, and/or resolution over time.*” The styles discussed above readily satisfy both definitions because they specify attributes that define the appearance of the image and how it changes at defined times (e.g., the resolution of the pocket watch image, where the watch hour hand points and how often it changes), as discussed. The style also defines an “*independent process*” because the parameters are entirely responsible for specifying the sequence of images and speed of the animation for the object on the web page—and are thus not dependent on other objects on the web page.

87. The third definition from previous litigation for “**time lines**” requires an “*independent asynchronous process that defines the values for an object,*” which in my opinion adds nothing material to my analysis above. In computing, an asynchronous process is generally understood as one that can occur at the same time as (or concurrently with) other processes. An asynchronous process is often contrasted with a “synchronous” process that “blocks” or otherwise prevents other

processes from executing until the synchronous process completes. In any case, any “independent asynchronous process” requirement is satisfied by the built-in multithreading features built-in to Java, as explained below.

88. This is confirmed by Laura Lemay, *Teach Yourself Java in 21 Days* (1996) (**Ex. 1005**), which I will refer to as “**Lemay Java**.” Lemay Java was written by the same author as Lemay Web discussed above, but is a different book.

89. Lemay Java explains that “[m]ultithreading, as in Java, enables several different execution threads to run at the same time inside the same program, in parallel, without interfering with each other.” (Lemay Java, p.178.) Thus, “[u]sing threads, you can have lots of applets running at once on the same page,” *i.e.*, they can run “independently.” (*Id.*) As Lemay explains, “[e]ven if you don’t have lots of applets, using threads in your applets is good Java programming practice. The general rule of thumb for well-behaved applets: whenever you have any bit of processing that is likely to continue for a long time (such as an animation loop, or a bit of code that takes a long time to execute), put it in a thread.” (*Id.*)

90. It thus would have been obvious to implement the animated Java image object discussed in Lemay Web as its own process thread, as disclosed in Lemay Java. This would have predictably resulted in the animated image object in Lemay Web being implemented in its own process or thread, thus making it both

“independent” and “asynchronous” of other processes and satisfying even the most restrictive definition of “**time lines**” from prior litigations. The motivation to combine would have been straightforward; Lemay Java confirms that it was a “general rule of thumb” in order to create “well-behaved” “animation loop[s]” and to enable multiple applets to run at the same time on the same page without interfering with each other. (Lemay Java, p.178.)

91. *Manner, Rationale, and Motivation to Combine Reynolds with Lemay Web and Lemay Java:* It would have been obvious to a person of ordinary skill in the art to combine Reynolds with Lemay Web and Lemay Java, as explained above. The combination would have predictably resulted in the ibook creation system of Reynolds further adapted to allow the user to specify for each web page, during the process of creating an ibook passage, an animated image object based on the teachings of Lemay Web (“**wherein each web page comprises at least one button object or at least one image object**”). An exemplary implementation of this combination would have allowed a user of authoring tool **86** in Reynolds to provide user input defining an image object for each page by specifying the “Animator” Java applet in Lemay Web, and also providing the user-defined HTML parameters specifying the visual appearance of the object including by identifying the images, their resolution, their sequence and timing, and other visual characteristics as

discussed above (“**wherein the at least one button object or at least one image object is associated with a style that includes values defining transformations and time lines for the at least one button object or at least one image object**”).

As explained for claim 1[b] above, such user input would have been accepted by the ibook server application 64 (“**build engine**”) for storage in its databases. A person of ordinary skill in the art would have had many motivations to combine.

92. To begin with, Reynolds, Lemay Web and Lemay Java are analogous references in the same field as the ’168 patent. The ’168 patent identifies its “Field of the Invention” as being directed to “methods and apparatus for building a web site using a browser-based build engine.” (’168, 1:13-15.) 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 ’168 patent, and is reasonably pertinent to a problem faced by the inventors of how to implement a web site building system. Lemay Web and Lemay Java also fall within the same field of endeavor as they describe two of the basic building blocks for building websites at the time of the alleged invention, *i.e.* HTML and Java. Lemay Web and Lemay Java would also have been pertinent to a plethora of issues faced by the inventors in implementing the alleged invention, including how to define and implement objects and specifying

attributes defining how those objects will appear and behave on a web page.

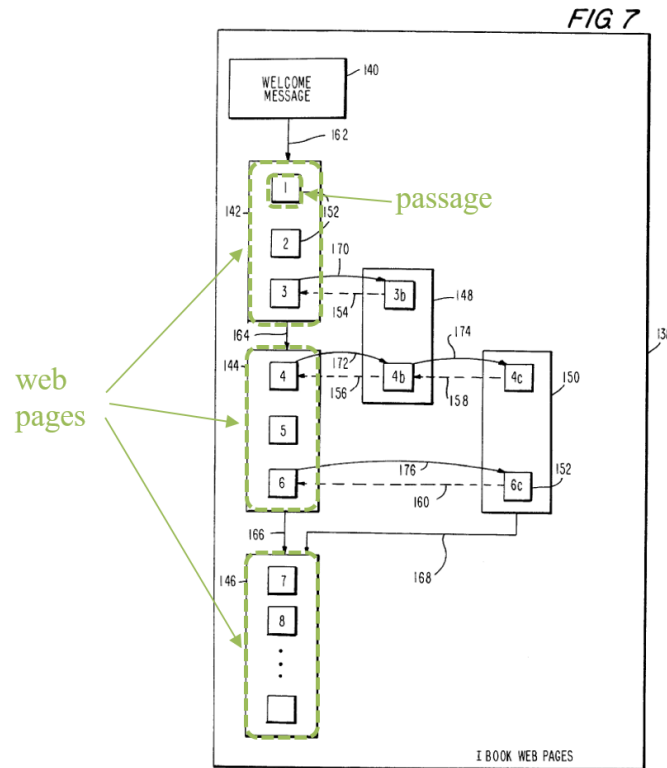
93. An ordinarily skilled artisan would have found this combination straightforward. As noted, Reynolds itself already discloses that passages **152** of an ibook can include “graphic images,” “animation sequences,” and “executable material” including “modules of code.” (Reynolds, 8:7-11; *see also id.*, 9:52-54 (“When passages **152** are executable modules, navigation tool **84** presents the passages by downloading and executing the passages.”).) Reynolds itself also confirms that passages can be implemented as Java programs. (Reynolds, 9:56-58 (“Each style is controlled by an ibook passage (e.g., a Java program) which implements that style.”).) It would therefore have been obvious, in view of Lemay Web, to implement an ibook in Reynolds such that each web page includes at least one image or button object using the Java techniques described above. Lemay Web provides an express motivation to do just that by explaining that the Java applet discussed above (the Animator) is a pre-built applet provided by the makers of Java. (Lemay Web, p.254.) An ordinarily skilled artisan would have been motivated to use these Animator techniques in Lemay Web in order to speed application development; as Lemay Web explains, “you can use pre-built Java applets on your pages to create animation effects without touching a line of Java code.” (Lemay Web, p.254.) “All you have to do is download the Java applet to your system,

include a few lines of HTML on your page, and everything works just fine . . . .”  
(Lemay Web, p.254.) An ordinarily skilled artisan would also have been motivated to adapt the Reynolds ibook creation system to support the animated image objects of Lemay Web in order to enrich user experience, as animated images are likely to be viewed as more engaging than textual content. And as explained previously Lemay Java provides a further motivation to use the techniques discussed above to harness the advantages of Java multithreading, resulting in an image object that is both “independent” and “asynchronous” of other processes.

**(d) “and wherein each web page is defined entirely by each of the plurality of objects comprising that web page and the style associated with the object,” (Claim 1[d])**

94. Reynolds discloses and renders obvious that each web page is defined entirely by each of its passages and the associated style; as noted above, it would have been obvious that each passage under the combination of Reynolds with Lemay Web and Lemay Java would have included an animated image object. (Reynolds, 8:7-10 (“If an ibook contains multimedia material, passages **152** can be graphic images, animation sequences, portions of a sound track, still video images, or video clips.”).) Figure 7 of Reynolds, reproduced and annotated below, illustrates an ibook website comprising various web pages (**142, 144, 146, 148, and 150**), each web page in turn composed of passages **152** (which are retrieved from Web pages database

76):

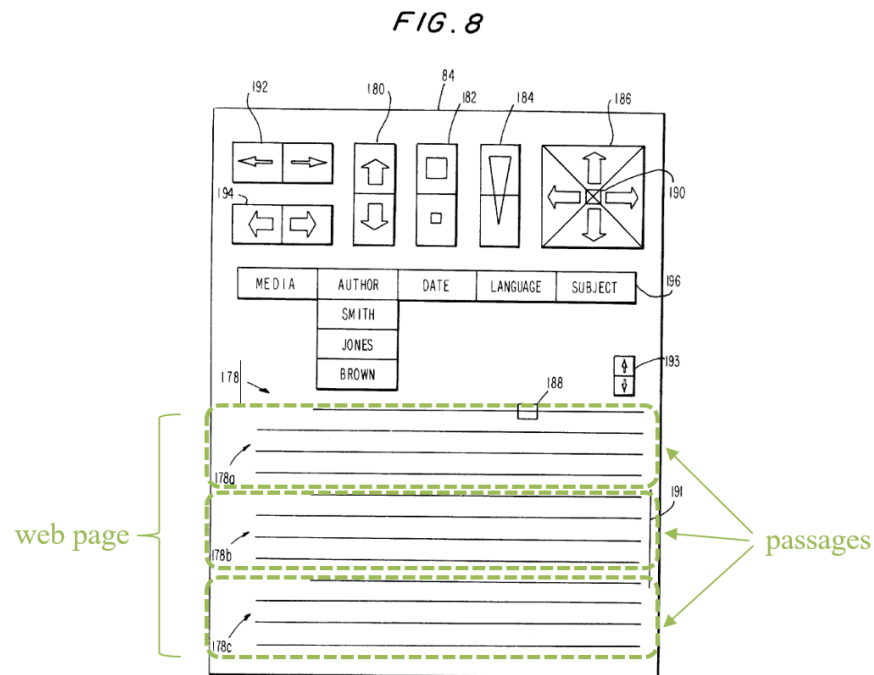


(Reynolds, Fig. 7 (annotations added); *see also id.*, 7:66-67 (“Web pages **142**, **144**, **146**, **148**, and **150** are preferably organized in the form of one or more passages **152**.”), 7:37-39.)

95. Reynolds makes clear that each web page in an ibook is defined **entirely** by passages **152** (“**each of the plurality of objects**”) that make up that page. This is because, as Reynolds confirms, each ibook web page is nothing more than a collection of passages. This is shown by Figure 8 of Reynolds below, disclosing an exemplary ibook web page as shown in a web browser using

Declaration of Vijay K. Madiseti, Ph.D. in Support of  
Petition for *Inter Partes* Review of  
U.S. Patent No. 7,594,168 B2

navigation tool **84**, displaying multiple passages **178** (**178a**, **178b**, and **178c**) in accordance with the style associated with those passages:



(Reynolds, Fig. 8; *see also id.*, 9:27-30 (“during the creation of passages **152** authoring tool **86** and ibook server application **64** cause information to be stored in passage database **74** that identifies which Web page **138** contains each passage **152**”).) Although navigation tool **84** in Figure 8 also displays various buttons and other on-screen tools (*e.g.*, **180**, **182**, **184**, **186**, **188**, **192**, **193**) and menus **196**, these are not part of the actual web page but user interface features of navigation tool **84**. (Reynolds, *e.g.*, 9:54-56, 9:62-10:16, 10:27-38.) All of the content displayed for an ibook web page, in other words, is based on passages **152** associated with the page.

Under the combination of Reynolds with Lemay Web and Lemay Java, therefore, each web page in an ibook is defined entirely by the passages (“**the plurality of objects**”) comprising that web page and the styles associated with those passages.

96. Under the proposed combination, “each web page is defined entirely by each of the plurality of objects comprising that web page **and the style associated with the object**,” as claimed. As previously explained in connection with element 1[c], each passage (object) based on the proposed combination with Lemay Web and Lemay Java can also include an associated style specifying how an animated image object is presented to a viewing user. Because each style specifies how its corresponding object will appear on the page, the prior art renders obvious that each web page is also defined entirely by “**the style associated with the object.**”

- (e) “**produce a database with a multidimensional array comprising the objects that comprise the web site including data defining, for each object, the object style, an object number, and an indication of the web page that each object is part of, and**” (Claim 1[e])

97. Reynolds discloses and renders obvious that its ibook server application produces a database comprising the objects that make up the web site. As explained for claim 1[a]-[b] above, ibook server application 64 receives user input defining passages via authoring tool 86 and stores data in its databases. For convenience, I will first explain how database storage of the particular information items in claim

1[e] (*i.e.* “the objects that comprise the web site,” “the object style,” “an object number,” “an indication . . .”) is disclosed and rendered obvious by Reynolds. I will then explain how it would have been obvious, based on Reynolds in combination with Perry and Witkowski, to store this information in “**a database with a multidimensional array**,” as recited in claim 1[e].

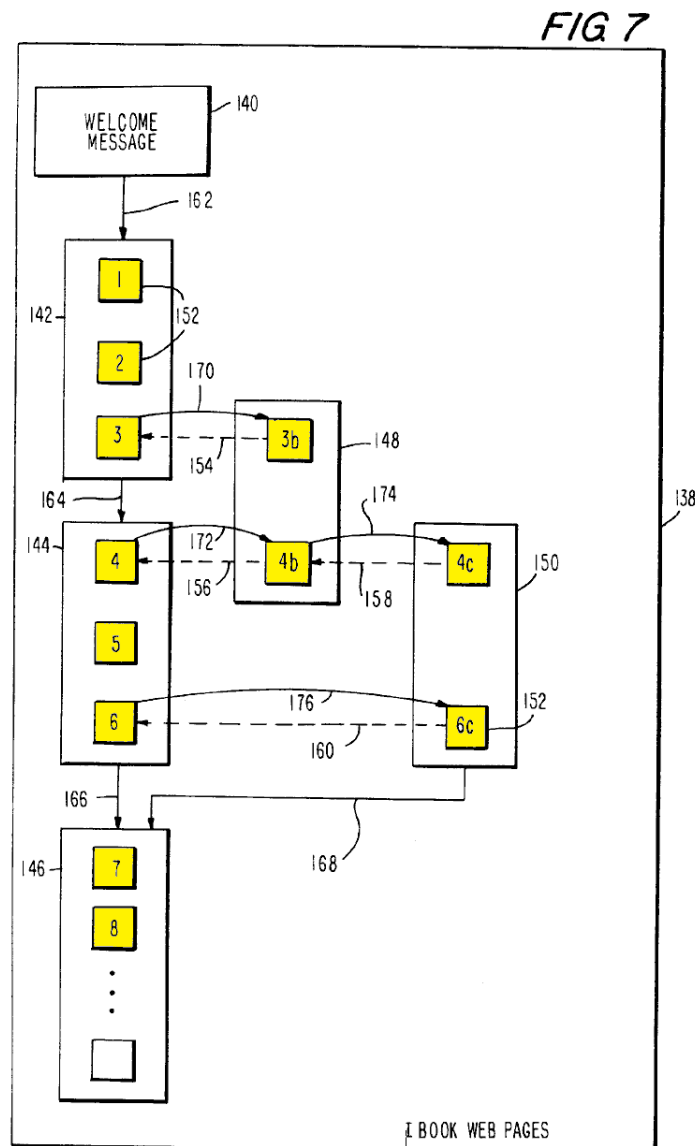
98. Reynolds discloses databases that store “**the objects that comprise the web site**,” as claimed. As explained, the content of ibook web pages is contained in passages which are stored in database 76. (Reynolds, *e.g.*, 1:34-38 (“Within each Web page, material may be organized in passages. Passages 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.”), 7:37-39 (“The content of the ibook Web pages is preferably stored in ibook Web pages database 76 (FIG. 3).”).) As also explained for claim 1[b], each passage of an ibook web page in Reynolds is stored as a discrete entity. When viewing an ibook web page, in fact, navigation tool 84 individually requests passages from the server (which in turn retrieves them from database 76) on a passage-by-passage basis. (Reynolds, *e.g.*, 11:28-40.) Reynolds therefore discloses and renders obvious “**a database . . . comprising the objects that comprise the web site**,” as claimed.

99. The databases under the combination of Reynolds and Lemay Web and

Lemay Java also “**includ[e] data defining, for each object, the object style,**” as claimed. Under the proposed combination as discussed above, data defining the style for each passage (“**for each object, the object style**”) would be provided by the user through authoring tool **86** of Reynolds. It therefore would have been obvious that the same databases in Reynolds that store passages would store their associated styles. Using the animated image object discussed at length in the discussion of claim 1[c], the style is provided within the HTML code for the web page on which the animated image will appear. The styles would thus be stored in the same web pages database **76** of Reynolds that stores page content.

100. The databases in Reynolds also store data defining, for each passage (“**for each object**”) in an ibook, “**an indication of the web page that each object is part of.**” As Reynolds explains, “during the creation of passages **152** authoring tool **86** and ibook server application **64** cause information to be stored in passage database **74** that identifies which Web page **138** contains each passage **152.**” (Reynolds, *e.g.*, 9:27-30.) Reynolds therefore discloses a database stores an indication of the web page(s) that each passage is a part of.

101. Reynolds also renders obvious storing “**an object number,**” as claimed. Figure 7 in Reynolds shows an example ibook with web pages including passages, each passage (“**object**”) having a number:



(Reynolds, Fig. 7 excerpt (highlighting added).) “Web pages **142, 144, 146, 148** and **150** are preferably organized in the form of one or more passages **152.**” (Reynolds, 7:66-67.) Each of these web pages includes, as shown in highlighting, passages identified in Figure 7 with numbers. “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 (highlighting added).) Reynolds therefore discloses the  
association of a passage with a numeric value (“**object number**”).

102. Reynolds does not state whether the passage numbers discussed above  
are stored in the database (as opposed to being used for identification in the figure),  
but in any case, Figure 7 confirms that assigning a number to each passage (“**object  
number**”) would have been obvious to a person of ordinary skill in the art. As noted  
previously, Reynolds explains that during the creation of passages **152**, information  
is stored in passage database **74** “that identifies which Web page **138** contains each  
passage **152**.” (Reynolds, 9:27-30.) It would have been obvious that in order to  
keep track of the web pages containing a particular passage, the system must assign  
a value to each passage in the database in order to allow it to be identified. A person  
of ordinary skill in the art would have appreciated that a number is the simplest and  
most easily-implemented way of representing such an identifier—as confirmed by  
its use in Figure 7 to identify which passages are assigned to which ibook web pages.

103. Reynolds therefore discloses and renders obvious storing in a database  
“**the objects that comprise the web site including data defining, for each object,  
the object style, an object number, and an indication of the web page that each  
object is part of.**” Reynolds also discloses that the ibook server application **64** (the

“**build engine**”) “**produce[s] a database**,” as claimed. As explained previously, Reynolds explains that ibook server application **64** “directs the overall operation of ibook server **56**,” and in particular, “controls databases **66, 68, 70, 72, 74, and 76**, in which various ibook information is stored.” (Reynolds, 5:20-23.) It thus would have been obvious that the “**build engine**” in Reynolds would be configured to produce the database recited in claim 1[e].

104. Reynolds does not itself disclose the ability to “**produce a database with a multidimensional array**,” comprising the information identified above. But the claimed database with a multidimensional array would have been obvious to a person of ordinary skill in the art in view of Perry and Witkowski.

105. Perry confirms that the ’168 patent inventors did not invent the concept of a “**multidimensional array**.” In computer science, the term “**array**” refers to a well-known data structure consisting of a number of elements, each element identified by an index that identifies the location of the element in the array. A simple example of an array is a string of characters—such as the five-character string “hello” containing five elements (textual characters). Each element of this array can be accessed by using an index value that identifies a particular position within the array from zero to four; an index value of zero corresponds to the letter “h” at the front of the string and an index value of four corresponds to the concluding “o” at

the end. This is an example of a *one-dimensional* array because only one index value (from zero to four) is needed to access each element in the array.

106. A multidimensional array, as its name implies, requires at least *two* indices to access each element in the array. As explained in Perry, “[m]ultidimensional arrays, sometimes called *tables* or *matrices*, have at least two dimensions (rows and columns).” (Perry, Ex. 1006, p.455 (italics in original).) “Whereas a single-dimensional array is a list of values, a multidimensional array simulates a table of values, or multiple tables of values. The most commonly used table is a two-dimensional table . . . .” (Perry, p.456.) Table 27.1 in Perry below provides a straightforward example of two-dimensional array for “a football team [that] wanted to track its players’ total yardage. The team has eight games scheduled, and there are six rushing players on the team.” (Perry, p.456.)

| <b>Table 27.1. A football team’s yardage record.</b> |               |               |               |               |               |               |               |               |
|------------------------------------------------------|---------------|---------------|---------------|---------------|---------------|---------------|---------------|---------------|
| <i>Player Name</i>                                   | <i>Game 1</i> | <i>Game 2</i> | <i>Game 3</i> | <i>Game 4</i> | <i>Game 5</i> | <i>Game 6</i> | <i>Game 7</i> | <i>Game 8</i> |
| Downing                                              | 1             | 0             | 2             | 1             | 0             | 0             | 0             | 0             |
| Edwards                                              | 0             | 3             | 6             | 4             | 6             | 4             | 5             | 3             |
| Franks                                               | 2             | 2             | 3             | 2             | 1             | 0             | 2             | 3             |
| Grady                                                | 1             | 3             | 2             | 0             | 1             | 5             | 2             | 1             |
| Howard                                               | 3             | 1             | 1             | 1             | 2             | 0             | 1             | 0             |
| Williams                                             | 2             | 3             | 1             | 0             | 1             | 2             | 1             | 2             |

(Perry, Table 27.1, p.456.)

107. Perry explains that Table 27.1 “has rows (the first dimension) and columns (the second dimension). Therefore, this is a two-dimensional table with six rows and eight columns.” (Perry, p.456.) “Each row has a player’s name, and each column has a game number, but these are not part of the actual data. The data consists only of 48 values (six rows by eight columns).” (Perry, p.456.)

108. Table 27.1 provides an example of a “multidimensional array” because it provides a uniquely identifiable indexed set of related elements (in this case yardage values), where each element is addressed by a set of two or more indices—the player name (row) and the game number (column)—each index corresponding to a dimension of the array.

109. The basic concept behind a multidimensional array—the ability to store data in tables accessible by two or more dimensions (such as rows and columns)—also underlies some relational database implementations. This is confirmed by **Witkowski** (Ex. 1007), a patent assigned to Oracle that I cite for a 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 provides several examples of simple database tables that are organized into rows and columns. Figure 4a provides an example of a simple database table called TABLE X:

**TABLE X**

| x.row id | X.a | X.b |
|----------|-----|-----|
| X1       | 1   | 1   |
| X2       | 2   | 2   |
| X3       | 3   | 3   |
| X4       | 4   | 4   |
| X5       | 9   | 0   |

(Witkowski, Fig 4A excerpt (highlighting added).) “Referring to FIG. 4a,” Witkowski explains, “table X includes two columns x.a and x.b and a pseudo-column ‘X.rowid’. The pseudo column contains a unique identifier for each row in table X. The pseudo-column may, for example, simply represent the addresses of

the actual storage locations at which the rows are stored.” (Witkowski, 15:54-59.)<sup>2</sup>

Witkowski describes various techniques for “joining” Table X with Table Y (also shown in Figure 4A), but those techniques are not pertinent to the limited purpose for which I have cited Witkowski, *i.e.*, to show an exemplary database table that stores information into rows and columns. This table shows an example of a “**multidimensional array**,” as claimed, because it provides a uniquely identifiable indexed set of related elements (Table X), in which each element is addressed by two indices—the row id (*i.e.* x.rowid) and the column (*i.e.* x.a or x.b). For example, if one wanted to obtain the value shown in yellow highlighting above, one could obtain that value by asking the database to retrieve the value in column x.a within the row whose id is X1. For example, in a relational database based on the structured query language (SQL), this value could be obtained with an SQL statement such as “SELECT X.A FROM TABLE X WHERE X.ROWID = X1.” In any case,

---

<sup>2</sup> Witkowski uses Figure 4A to show an example of a “base table,” the values of a table at a particular time. My analysis relies on Figure 4A of Witkowski for basic disclosure of the structure of a table in a relational database system, which would have been general knowledge to an ordinarily skilled artisan, and my analysis does not rely on the specific table joining techniques disclosed in Witkowski.

Witkowski discloses the ability to store data in a database through a “multidimensional array” by using a database table that stores data in rows and columns, with indices for addressing dimensions (*i.e.* row or column).

110. ***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 Perry and Witkowski discussed above. The combination would have predictably resulted in the system of Reynolds further adapted to include **a database with a multidimensional array**, which includes the specific information recited in the claim and stored in Reynolds in passage database 74 and/or web page database 76. This combination would thus have resulted in a relational database that stores the information recited in claim 1[e], in a database table with rows for each record, and each row having columns or fields (“a database with a multidimensional array”). As applied to Reynolds, this could have been implemented in any number of ways, but the most straightforward would have been for the rows to correspond to ibook passages (“**the objects that comprise the web site**”), and for each passage, a number of columns (“**including data defining, for each object**”) containing the data relating to passages as discussed above. This would have provided a **multidimensional array** because the table would have provided a uniquely identifiable indexed set of related elements, with each element identified by two

indices—an identifier of the row (passage) and an identifier of the column within the row where the desired information is stored.

111. An ordinarily skilled artisan would have regarded this as a straightforward, and in fact trivial, combination. Implementation of multidimensional arrays is often taught in introductory undergraduate computer programming courses and is considered elementary programming knowledge well within the knowledge of ordinarily skilled artisans. The ability to store data using two dimensions and accessible using two indices (*e.g.* through rows and columns) is also a natural and fundamental way that human beings organize information, as evidenced by its widespread use with spreadsheets, bookkeeping ledgers, and numerous other real-world applications as shown for example in the football player table in Perry. (Perry, Table 27.1, p.456.) In fact, any situation that requires storage of (a) a number of pieces of information about (b) a number of items lends itself easily to a two-dimensional array with information stored in rows and columns. It thus would have been obvious to implement a database in Reynolds, which must store a number of pieces of information about each passage, through a two-dimensional array such as a table with rows and columns.

112. And like multidimensional arrays, relational databases were also 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.

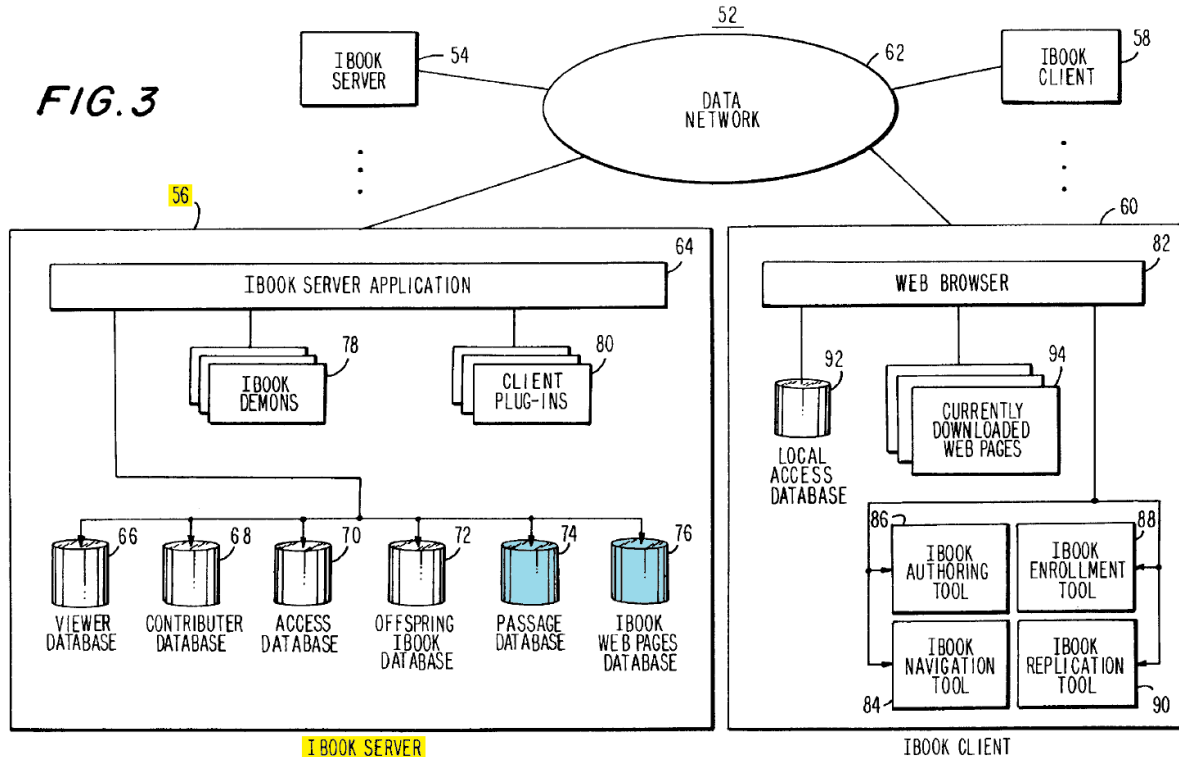
(Reynolds, 15:30-37.) Storing data in the form of tables using relational databases was a popular (if not the most popular) form of database storage by December 1999. Relational databases were commercially available through off-the-shelf tools from third party vendors such as Oracle (the assignee of 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 table-based database storage of the passage-related information of Reynolds because relational databases were technologically mature, well-documented, stable, and could have been 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 a multidimensional array organized into rows and columns.

113. 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 enhanced 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.

**(f) “provide the database to a server accessible to [sic; a] web browser;” (Claim 1[f])**

114. Reynolds discloses that web pages database **76** is provided to a server accessible to web browser **82**. In particular, web pages database **76** is provided to web server **56**, as shown in Fig. 3:



(Reynolds, Fig. 3 (highlighting added).) As shown, passage database **74** and ibook web pages database **76** are provided to ibook server **56** and ibook server application **64**. Reynolds makes clear that ibook server application **64** (the “**build engine**”) **provides** these databases to server **56**. (Reynolds, 5:19-22 (“Ibook server application **64** directs the overall operation of ibook server **56**. In particular, ibook server application **64** controls databases **66**, **68**, **70**, **72**, **74**, and **76**, in which various ibook information is stored.”).) Under the proposed combination of Reynolds with Perry and Witkowski, as explained in claim 1[e] above, the information in these databases would be stored in “**a database with a multidimensional array,**” as

claimed. The proposed combination thus renders obvious the build engine being configured to “**provide the database to a server.**”

115. Reynolds further confirms that the server (and the claimed “database”) are “**accessible to [a] web browser,**” *i.e.* Web browser **82** as shown in Figure 3 above. Reynolds explains that an ibook client **60** viewing an ibook uses Web browser **82** (with navigation tool **84** installed via a plugin downloaded from Web server **56**) to retrieve passages from web pages database **76** at the server **56**, and display web pages composed of the retrieved passages. (Reynolds, 5:27-31 (explaining that browser plugins **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**”), 5:35-37 (“When client plug-ins **80** are downloaded onto client **60**, they form tools **84**, **86**, **88**, and **90**.”), 11:24-39 (explaining that navigation tool **84** communicates with server application **64** to retrieve ibook website content from web pages database **76** to generate ibook web pages).)<sup>3</sup> Under

---

<sup>3</sup> Claim 1[f] is not clear as to whether the “**database,**” or the “**server,**” must be “**accessible to [a] web browser.**” But as confirmed by the citations from Reynolds in the text, this distinction is immaterial because both the server and the database are

the proposed combination discussed in claim 1[e], the database in Reynolds would be adapted to include a multi-dimensional array as discussed. In any case, the database and server would be “**accessible to [a] web browser,**” as claimed.

(g) “**wherein the database is produced such that a web browser with access to a runtime engine is configured to generate the web-site from the objects and style data extracted from the provided database.**” (Claim 1[g])

116. This limitation was largely covered in my discussion of the preceding claim limitations. For purposes of this claim, the “**runtime engine**” in Reynolds takes the form of navigation tool **84**. Navigation tool **84** is one of the client plug-ins **80** that “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.) “When client plug-ins **80** are downloaded onto client **60**, they form tools **84**, **86**, **88**, and **90**.” (Reynolds, 5:35-37.)

117. Reynolds confirms that navigation tool **84** qualifies as a “**runtime engine**,” because it is a file executed at runtime (*e.g.* in web browser **82** as web pages are received), that facilitates the retrieval of information from the database and generates commands to display the ibook web pages. Reynolds explains how

---

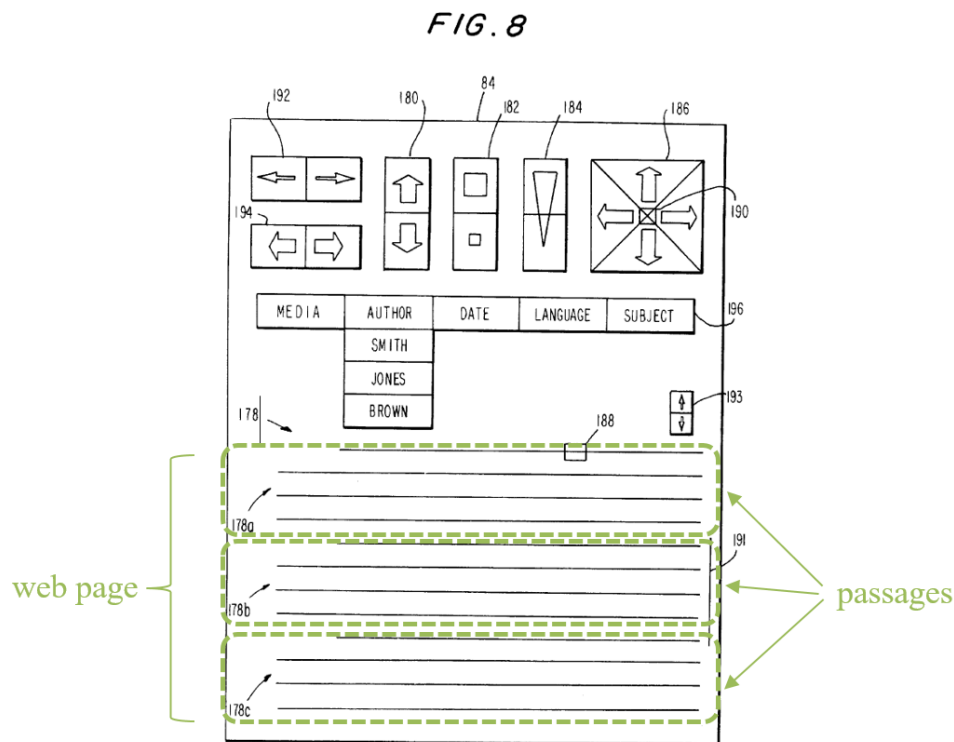
accessible to Web browser **82** in Reynolds, because the browser can access the server over network **62** and retrieve information from the database.

navigation tool **84** builds web pages from passages retrieved from the database:

At step 198, the user at client **60** determines which passage is to be viewed, for example, by using the various buttons provided by navigation tool **84** (FIG. 8). Navigation tool **84** preferably generates a request to view a desired passage and transmits this request to server **56** at step 200. 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, 11:28-40.)

118. Figure 8 of Reynolds, reproduced and annotated below, illustrates how navigation tool **84** on the client generates a web page built from passages **178a**, **178b**, and **178c** retrieved from the server and database:



(Reynolds, Fig. 8 (annotation added).) As explained, “[w]hen passages **152** are executable modules,” as is the case under the combination of Reynolds with Lemay Web and Lemay Java as discussed for claim 1[c], “navigation tool **84** presents the passages by downloading and executing the passages.” (Reynolds, 9:52-54.) Figure 8 also shows navigation tool **84** having various “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.) “As the user navigates through an ibook using navigation tool **84**,” as explained,

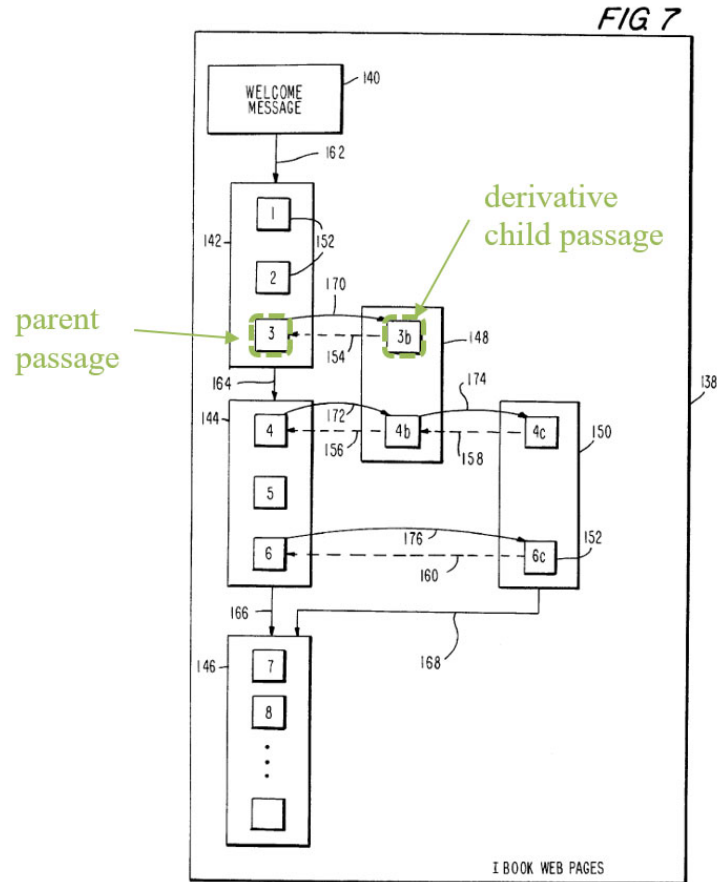
“navigation tool **84** transmits certain access information to server **56**, making requests to view material.” (Reynolds, 11:24-26.) Thus, as the viewing user navigates through the web pages of the ibook website, navigation tool **84** generates the web pages by requesting and displaying the passages of those web pages from the web server, which are retrieved (i.e., extracted) from the database. (Reynolds, 11:24-38.)

119. Reynolds therefore discloses that the database “**is produced such that a web browser with access to a runtime engine,**” *i.e.* navigation tool **84** executing in the web browser **82**, “**is configured to generate the web-site from the objects and style data extracted from the provided database.**” Under the combination of Reynolds with Lemay Web and Lemay Java as explained for claim 1**[b]**-**[c]** above, ibook passages are associated with style data (*e.g.* data defining the appearance of a Java image or button object), which is stored in the database as explained for claim 1**[d]** above. Accordingly, under the proposed combination, navigation tool **84** will generate the website for display in the browser from passages and style data extracted from the database.

2. **Claim 2: “The system of claim 1, wherein one of said plurality of objects is a child, and wherein the build engine is configured to accept user input to associate a style with child button and child image objects.”**

120. Reynolds discloses that passages (“objects”) can have a **child**, which

corresponds in Reynolds to a derivative passage such as passage 3b in Figure 7:



(Reynolds, Fig. 7 (annotation added).) As I explain below, Reynolds and Lemay Web render obvious that server application 64 (the “**build engine**”) “**is configured to accept user input to associate a style with child button and child image objects**” in two ways: (1) by accepting user input to directly associate a style with an object of a derivative passage; or (2) by associating the user-inputted style of the parent passage to the derivative passage.

121. First, as I explained for claim 1[c] above, it would have been obvious to adapt Reynolds' server application **64** based on the teachings of Lemay Web to allow the user to specify for each web page, during the process of creating an ibook passage, an animated image object (e.g., by specifying an instance of the "Animator" Java applet in Lemay), including attributes defining the object's visual appearance (i.e., the object's "**style**"). The proposed adaptation of Reynolds' server application **64** applies equally to all of Reynolds' passages, including derivative passages that are "based on" the parent passage but are otherwise created by the author of the derivative passage. (Reynolds, 1:47-48 ("Works that are created based on previously contributed material are called derivations.")) Thus, the adapted Reynolds ibook creation system would accept user input to associate a style with an animated image object of a derivative passage that is otherwise based on a corresponding animated image object of a parent passage.

122. Alternatively, Reynolds explains that the content of the parent passage may be copied to, and thus associated with, a derivative passage. (Reynolds, 1:47-61 (explaining that "[w]orks that are created based on previously contributed material are called derivations" that can be created "us[ing] certain authoring functions in the ibook system, such as by performing a cut and paste operation from existing material"), 8:28-35 ("Attribution information is preferably automatically

generated when a contributor copies existing ibook material. For example, contributor Jones may have used a cut and paste feature in derivation tool 132 to cut and paste text or multimedia material from passage 3 when creating passage 3b. During the process of cutting and pasting from passage 3 to passage 3b, authoring tool 86 automatically generates attribution information including attribution link 154.”.) When an animated image object is copied from a parent passage to a derivative passage, the copied material would naturally include the style that defines the visual appearance of the animated image object.

123. As explained in **Part V**, two definitions for “child”/“child object”/“child button” have been provided in previous litigations: “*an element with one or more properties that depends on another element*” and “*element/object/button that is spawned by another element/object/button (i.e., the parent element/object/button).*” Reynolds’ derivative passages meet either of these definitions because the content of a derivative passage “*depends on*” and is likewise “*spawned by*” its parent passage.

124. In addition to having content derived from a parent passage, a derivative passage has attribution properties that link the derivative passage to the parent passage, further indicating that at least part of the derivative passage *depends on* and is *spawned* from a particular parent passage:

Attribution information is preferably automatically generated when a contributor copies existing ibook material. For example, contributor Jones may have used a cut and paste feature in derivation tool **132** to cut and paste text or multimedia material from passage 3 when creating passage 3b. During the process of cutting and pasting from passage 3 to passage 3b, authoring tool **86** automatically generates attribution information including attribution link **154**.

(Reynolds, 8:28-35; *see also id.*, 8:19-21 (“Attribution link **154** indicates that passage 3b, which was contributed by Jones, was based on passage 3, contributed by Smith.”), 8:61-9:7 (“Derivation links **170, 172, 174, and 176** are another type of link that is formed in connection with ibook web pages **142, 144, 146, 148, and 150**. Derivation links **170, 172, 174, and 176** preferably point in the opposite direction of attribution links **154, 156, 158, and 160**.”).) The attribution properties of the child passage, in the form of derivation link **170** and attribution link **154**, are stored in passage database 74 to be “used by navigation tool 84 when a user navigates through the material in an ibook.” (Reynolds, 9:8-19.) Specifically, when navigating through the ibook, a user is “alerted” when a “derivation exists” and is “given an opportunity to view the derivation,” *i.e.*, child, passage. (Reynolds, 9:12-15.) Likewise, the user is alerted when a passage is a derivation and thus has a “corresponding original,” *i.e.*, parent, passage. (Reynolds, 9:16-17.) Thus, both the content and the attribution properties of the child passage depend on and are spawned

from the parent passage.

125. Thus, it would have been obvious to adapt Reynolds' ibook server application to accept user input to associate a style with animated image objects (which are also button objects) of derivative child passages, either through the user's specification of new animation styles for the derivative child passages or through copying the user-specified styles of the parent passages.

**3. Claim 3: "The system of claim 2, wherein at least one of said styles includes values defining time lines for child button and child image objects."**

126. As I previously described for claim 2, Reynolds in view of Lemay Web renders obvious the adaption of Reynolds' ibook server application to accept user input to associate an animation style for animated image objects (which are also button objects) of derivation (*i.e.*, **child**) passages. Moreover, as I explained for claim 1[c], it would have been obvious in view of Lemay Web and Lemay Java to adapt Reynolds' ibook creation system such that the animation style includes "**values defining time lines for the . . . objects**" in the form of <PARAM> tags such as "STARTIMAGE," "ENDIMAGE," "PAUSE," and "REPEAT." (*See, e.g.*, Lemay Web, p.256.) Thus, the combination of Reynolds, Lemay Web, and Lemay Java renders obvious that the animation **style** for animated image objects of a derivation passage ("**child button and child image objects**") includes "**values**

**defining time lines for the child button and child image objects.”**

**4. Claim 4: “The system of claim 1, wherein at least one of said styles includes settings for multiple object states.”**

127. As I previously explained for claim 1[c], the combination of Reynolds and Lemay Web discloses that the user-defined animation style for Java’s Animator applet includes the STARTIMAGE and ENDIMAGE parameters that set which images (and how many) are used for the animation style. (*See, e.g.,* Lemay Web, p.256 (explaining how a user sets the parameters for the Animator Java applet).) Thus, the combination of Reynolds and Lemay Web discloses styles including **“settings for multiple object states,”** because each image represents a different object state, and the animation transitions between successive states each displaying a different image as set by the STARTIMAGE and ENDIMAGE parameters.

**5. Claim 6: “The system of claim 1, where said data is stored as one or more of a Boolean an integer, a string, a floating point variables, or a URL.”**

128. Claim 6 depends from claim 1, which recites “produce a database with a multidimensional array comprising the objects that comprise the web site including **data defining**, for each object, the object style, an object number, and an indication of the web page that each object is part of[.]” I have therefore interpreted **“said data”** in claim 6 to refer to the specific data (*i.e.* object style, object number, and indication of the web page of which each object is part of) recited in claim 1. Claim

Declaration of Vijay K. Madiseti, Ph.D. in Support of  
Petition for *Inter Partes* Review of  
U.S. Patent No. 7,594,168 B2

6 adds nothing significant and is disclosed by or obvious in view of the prior art cited for claim 1.

129. As I explained for claim 1, the “**object style**” under the combination of Reynolds, Lemay Web and Lemay Java is expressed through a series of HTML statements (*e.g.* <APPLET> and <PARAM> tags) that identify the Java animator applet and specify characteristics of how it will be presented on the screen.

```
<APPLET CODE="Animator.class" WIDTH=100 HEIGHT=100>  
<PARAM NAME="STARTIMAGE" VALUE="1">  
<PARAM NAME="ENDIMAGE" VALUE="12">  
<PARAM NAME="PAUSE" VALUE="1000">  
<PARAM NAME="REPEAT" VALUE="TRUE">  
</APPLET>
```

(Lemay Web, p.256.) These HTML statements are simply a collection of textual characters that, at a minimum, would be stored as at least a “**string**,” as claimed. (*See Microsoft Press Computer Dictionary* (3d ed. 1997), Ex. 1009, p.451 (defining “string” as “[a] data structure composed of a sequence of characters usually representing human-readable text”).) The <APPLET> tag as shown also stores a textual string containing the name of the Animator applet (“Animator.class”), and the <PARAM> tags store “**integer**” variables (*i.e.* values for the STARTIMAGE, ENDIMAGE, PAUSE parameters), and “**Boolean**” variables (*i.e.* the TRUE or FALSE value for REPEAT).

130. With respect to the “**object number**,” as explained for claim 1, it would

have been obvious to assign a numeric value to each passage to facilitate identification. It would have been obvious to an ordinarily skilled artisan that an “object number” would be represented at least as an **“integer.”** (*See* Ex. 1009 p.254 (defining “integer” as “[a] data type representing whole numbers,” which is “widely used in programming for counting and numbering purposes”).) An integer is, as noted, “widely used for counting and numbering purposes,” (*id.*), so it would have been obvious to store the object number as an integer as claimed.

131. Finally, with respect to the **“indication of the web page that each object is part of,”** it would have been obvious to store this data as a **“URL,”** as recited in claim 6. As explained for claim 1, Reynolds explains that “during the creation of passages **152** authoring tool **86** and ibook server application **64** cause information to be stored in passage database **74** that identifies which Web page **138** contains each passage **152.**” (Reynolds, 9:27-30.) Although Reynolds does not state the format of this information, it would have been obvious to store it in the form of a Uniform Resource Locator (URL), the standard and widely known way of representing and uniquely identifying documents or resources on the web. This is confirmed by Reynolds itself, which provides several examples of storing URLs to identify ibook web pages and passages in ibook Web pages database **76**. (Reynolds, *e.g.*, 15:24-26 (“For example, server **278** may contain the URLs of passages **276** and

290 in ibook database such as passage database 74 and ibook Web pages database 76.”), 11:53-55 (“The passage identifier is preferably a universal resource locator (URL) for the passage requested from the ibook Web pages database 76.”), 15:48-51 (“If desired, ibook Web pages on a given server or client may be represented by a reference (e.g., a URL) to material that is actually stored elsewhere in the data network.”).) Because Reynolds expressly discloses the use of URLs to identify passages and web pages, and it was widely known to ordinarily skilled artisans (and even casual users) to use URLs to identify web pages, it would have been obvious to use URLs to keep track of which objects are associated with which web pages, which would have provided the claimed “indication” of the web page of which each object is a part.

132. But at a more fundamental level, an ordinarily skilled artisan would not have perceived claim 6 as adding nothing of patentable significance. Regardless of the information that is being stored in the database, it would have been obvious to store it as “**Boolean,**” “**integer,**” “**string,**” or “**floating point variables,**” because these represent basic and fundamental data types used to represent information in a computer system. As the *Microsoft Press Computer Dictionary* explains under the definition of “data type,” “[d]efining the data type allows a computer to manipulate the data appropriately,” and “[d]ata types are most often supported in high-level

languages and often include types such as real, integer, floating point, character, Boolean, and pointer.” (Ex. 1009, p.133.) These data types would have been familiar to any ordinarily skilled artisan and are often taught in introductory undergraduate computer programming courses. An ordinarily skilled artisan could not even build a functioning computer program without understanding, at a minimum, how to use different data types including at least integer and string variables. In any case, it would have been plainly obvious to use the fundamental data types recited in claim 6 to represent data in the database, including the data recited in claim 1, and a skilled artisan would have been motivated to use widely-used data types that are already “supported in high-level languages.” (*Id.*)

133. For example, claim 6 allows the data to be stored as a “**string**,” which is simply “a sequence of characters,” and can be used to store any type of computer-usable information. (*See Microsoft Press Computer Dictionary*, Ex. 1009, p.451.) In fact, even if “string” is interpreted as a sequence of human-readable textual (ASCII) characters, it was well-known to use strings to represent data through encoding techniques such as Base64 (widely used for email attachments using Multipurpose Internet Mail Extensions (MIME)), uuencode/uudecode, and BinHex, all of which could encode any arbitrary data as a sequence of readable textual characters, a technique widely used to transmit data over Internet using email or

Declaration of Vijay K. Madiseti, Ph.D. in Support of  
Petition for *Inter Partes* Review of  
U.S. Patent No. 7,594,168 B2

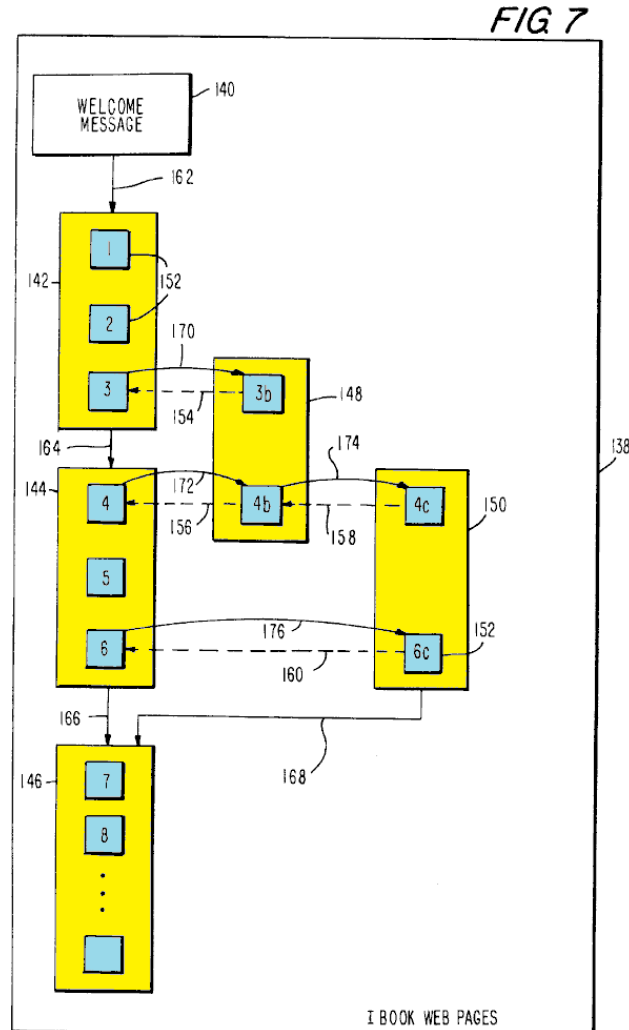
newsgroups. (See Ex. 1009, p.489 (definition of uuencode), p.51 (definition of BinHex), p.309 (definition of MIME), p.475 (“**translated file**. *n.* A file containing data that has been changed from binary (8-bit) format to ASCII (7-bit) format. BinHex and uuencode both translate binary files into ASCII. Such translation is necessary to transmit data through systems (such as e-mail) that may not preserve the eighth bit of each byte.”).) Thus, it would have been obvious that all of the data in Reynolds’ database, including data used to generate the “plurality of web pages,” could be stored as strings. A skilled artisan would have been motivated to do so because strings are not only a fundamental and well-known data type implemented using any high level programming language, but also ideal for transmission between servers and clients using known communications protocols for transferring data such as HTTP. (See, e.g., Ex. 1009, p.451; Coombs, p.351 (defining the “Hypertext Transfer Protocol (HTTP)” as “[a] system for transferring files across the World Wide Web”).) Under the proposed combination, for example, strings would have been used to store the data in the database in a form that could be readily transmitted to navigation tool **84** in Reynolds at the client using conventional Internet protocols. Navigation tool **84**, in turn, would use that information in generating the ibook web pages, as fully explained in my analysis of claim 1[g].

**C. Ground 2: Claims 1-4 and 6 are obvious over Reynolds in view of Lemay Web, Lemay Java, Perry, and Witkowski, in Further View of Coombs**

**1. Independent Claim 1**

134. As I explained above for Ground 1, Reynolds discloses a method for users to assemble 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”).).

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



136. (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, 7:57-67.) “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.

137. But Reynolds does not expressly use the word “**website**” to describe the collection of ibook web pages. Because the ’168 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.

138. 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.)

139. ***Rationale and Motivation to Combine:*** It would have been obvious to implement the ibook system according to the combined teachings of Reynolds, Lemay Web, Lemay Java, Perry 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”), 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.

140. 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.

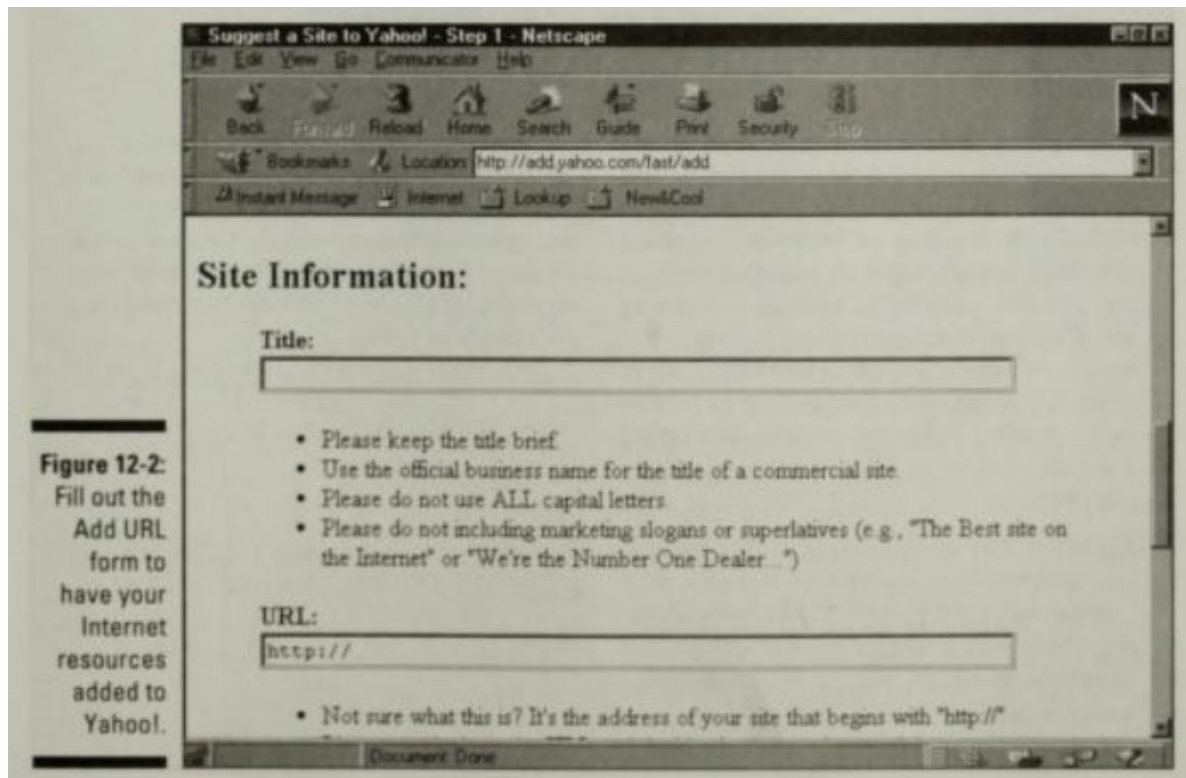
141. Reynolds and Coombs both provide express motivations to combine. First, as any user of web pages would appreciate, a domain name provides a simple and easy way for visitors to reach a website. Coombs explains that obtaining a domain name enables Internet users to access a website without “worry[ing] 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

turn would have made it easier to access the website.

142. 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.)

143. 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.

**2. Claim 2: "The system of claim 1, wherein one of said plurality of objects is a child, and wherein the build engine is configured to accept user input to associate a style with child button and child image objects."**

144. As explained in Section VI.B.2, Reynolds in combination with Lemay Web discloses this limitation.

**3. Claim 3: “The system of claim 2, wherein at least one of said styles includes values defining time lines for child button and child image objects.”**

145. As explained in Section VI.B.3, Reynolds in combination with Lemay Web and Lemay Java discloses this limitation.

**4. Claim 4: “The system of claim 1, wherein at least one of said styles includes settings for multiple object states.”**

146. As explained in Section VI.B.4, Reynolds in combination with Lemay Web discloses this limitation.

**5. Claim 6: “The system of claim 1, where said data is stored as one or more of a Boolean an integer, a string, a floating point variables, or a URL.”**

147. As explained in Section VI.B.5, this limitation would have been obvious in view of Reynolds.

**VII. NO SECONDARY CONSIDERATIONS OF NON-OBVIOUSNESS**

148. 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.

149. 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

Declaration of Vijay K. Madiseti, Ph.D. in Support of  
Petition for *Inter Partes* Review of  
U.S. Patent No. 7,594,168 B2

Patent Owner may assert) was a result of *the particular features recited in the '168 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.

150. 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 '168 patent, as follows:

151. *No commercial success of the claimed invention*. Patent Owner has not cited any evidence of particular commercial success of products embodying the '168 patent as opposed to products that do not embody the '168 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 '168 patent's claims and not due to any other facts.

152. *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 '168 patent. To the contrary, as discussed above, the prior art solved the problems that the '168 patent purported to address.

Declaration of Vijay K. Madiseti, Ph.D. in Support of  
Petition for *Inter Partes* Review of  
U.S. Patent No. 7,594,168 B2

153. *No failure of others.* Patent Owner has not cited any evidence of anyone who tried, but failed, to solve the problems addressed by the '168 patent. As shown by my analysis above, there existed prior art references that successfully disclosed and rendered obvious the subject matter claimed by the '168 patent.

154. *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 '168 patent and its claimed invention.

155. *No unexpected results of the claimed invention.* Patent Owner has not cited any evidence of unexpected results achieved by the '168 patent's claimed invention. To the contrary, the prior art disclosed the predictable, expected results that show why the '168 patent's claims are obvious as discussed in my Declaration.

156. *No praise for the claimed invention.* Patent Owner has not cited any evidence of praise for the claimed invention recited in the '168 patent.

157. *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 invention recited in the '168 patent.

158. *No departure from the wisdom of the prior art.* Patent Owner has not cited any evidence that the claimed inventions of the '168 patent departed from the wisdom of the prior art. The '168 patent claims subject matter that was already

Declaration of Vijay K. Madiseti, Ph.D. in Support of  
Petition for *Inter Partes* Review of  
U.S. Patent No. 7,594,168 B2

present in the prior art, including in the references discussed in my Declaration.

159. 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 '168 patent is directed to techniques known in the prior art and does not provide any inventive technology.

160. 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

161. 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 '168 patent claims at issue and/or offer testimony in support of this Declaration.

162. 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

Petitioner's Ex. 1002  
Page 0111

Dr. Vijay K. Madiseti  
Fellow, IEEE  
[vkm@madiseti.com](mailto: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.

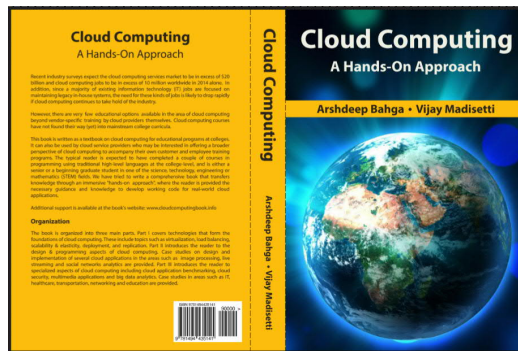
**Earned Degrees**

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

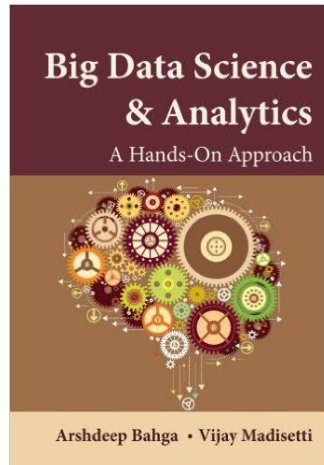
2. Ph.D., Electrical Engineering & Computer Sciences (EECS)  
University of California (UC), Berkeley. CA  
1989.

## Books

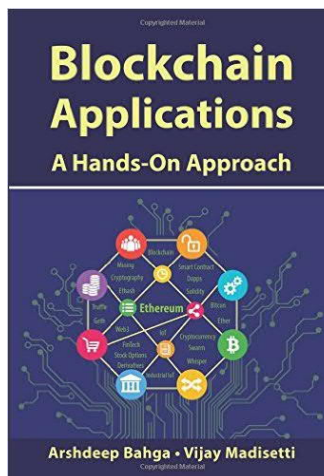
1. VLSI Digital Signal Processors  
Madiseti, V.K.;  
Boston: MA, IEEE Press: Butterworth Heinemann, 1995, 525 pp.
2. Quick-Turnaround ASIC Design in VHDL  
Romdhane, M., Madiseti, V.K., Hines, J.  
Boston: MA, Kluwer Academic Press, 1996, 190 pp.
3. The Digital Signal Processing Handbook (First Edition)  
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.; Madisetti, V.K.;  
Chapter in VLSI Signal Processing, VII, 1994.,  
Pages: 33 – 42
11. Fixed-point co-design in DSP  
Egolf, T.W.; Famorzadeh, S.; Madisetti, V.K.;  
Chapter in VLSI Signal Processing, VII, 1994.,  
Pages: 113 - 126
12. A fast spotlight-mode synthetic aperture radar imaging system  
Madisetti, 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.; Madisetti, 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.; Madisetti, 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  
Madisetti, 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.; Madisetti, V.K.;  
International Journal in Computer Simulation, Vol. 4, No. 4, Nov 1994.
17. System partitioning of MCMs for low power  
Khan, S.A.; Madisetti, 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.; Famorzadeh, 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.; Madiseti, 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.; Madiseti, 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  
Madiseti, 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.; Madiseti, 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.; Madiseti, 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.; Madiseti, 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.; Madiseti, 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.; Madiseti, 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.; Madiseti, 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; Madiseti, 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.; Madiseti, 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.; Madiseti, 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.; Madiseti, 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.; Madisetti, 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.; Madisetti, 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.; Madisetti, 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.; Madisetti, 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.; Madisetti, 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.; Madisetti, V.;  
Parallel and Distributed Processing Symposium, 2003. Proceedings. International , 22-26 April 2003  
Pages:8 pp.
33. Bandwidth aggregation with SCTP  
Argyriou, A.; Madisetti, 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-madiseti-argyriou-qos-sctp-00.txt>. Madiseti, V., Argyriou, A. IETF Internet-Draft, Jul 25, 2002.
2. Voice & Video over Mobile IP Networks <draft-madiseti-argyriou-voice-video-mip-00.txt> Madiseti, V., Argyriou, A. IETF Internet-Draft, Nov 20, 2002.
3. Enhancements to ECRTTP with Applications to Robust Header Compression for Wireless Applications. <draft-madiseti-rao-suresh-rohc-00.txt> Madiseti, 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.

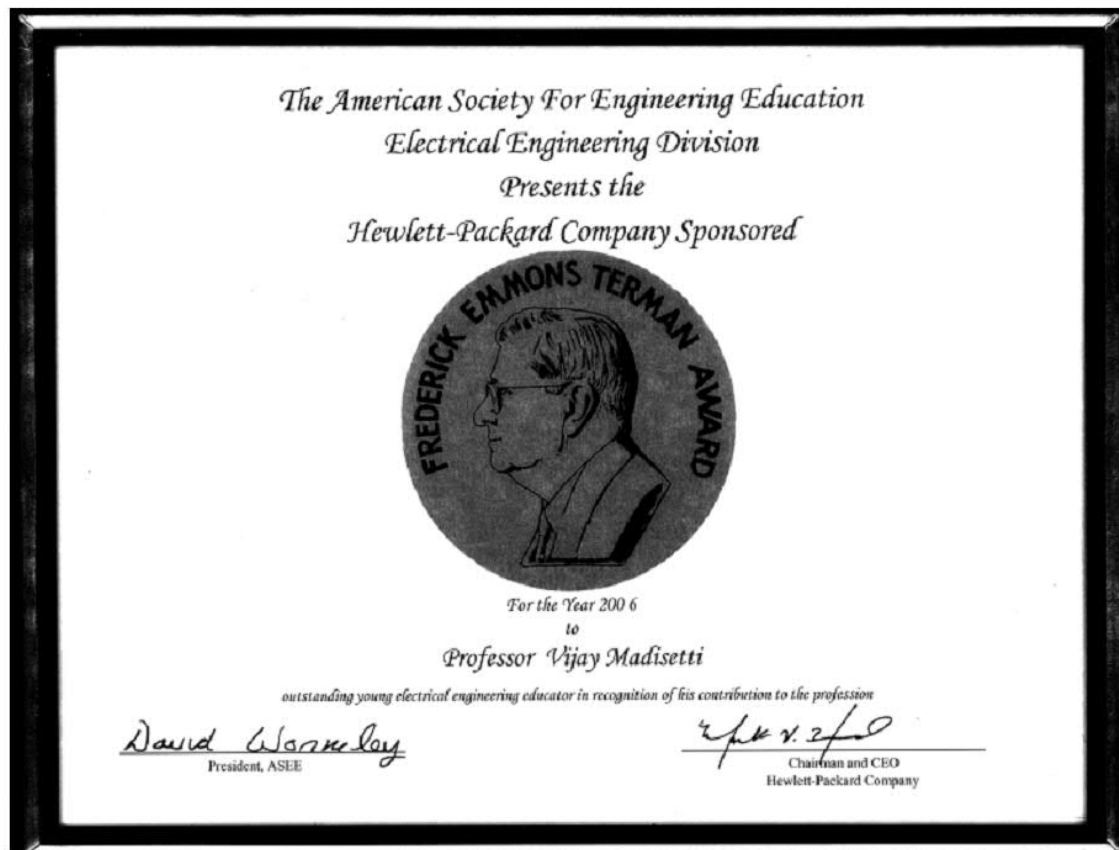
22. Pilho Kim, 2009, Scientist, VP Technologies, Inc.

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