

UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE PATENT TRIAL AND APPEAL BOARD

AKAMAI TECHNOLOGIES, INC.,

Petitioner,

v.

EQUIL IP HOLDINGS LLC,

Patent Owner.

Case IPR2023-00332

U.S. Patent No. 9,158,745

**DECLARATION OF VIJAY K. MADISETTI IN SUPPORT OF PETITION
FOR *INTER PARTES* REVIEW OF U.S. PATENT NO. 9,158,745**

TABLE OF CONTENTS

	Page
I. INTRODUCTION.....	1
II. QUALIFICATIONS.....	3
III. LEGAL UNDERSTANDING.....	11
A. My Understanding of Claim Construction.....	11
B. My Understanding of Anticipation	12
C. My Understanding of Obviousness.....	15
IV. BACKGROUND OF THE TECHNOLOGY	20
V. THE '745 PATENT AND PROSECUTION HISTORY.....	24
A. '745 Patent Overview	24
B. Prosecution History of the '745 Patent.....	26
VI. THE '745 PATENT PRIORITY DATE	28
VII. LEVEL OF ORDINARY SKILL IN THE ART	29
VIII. CLAIM CONSTRUCTION	31
A. Preambles	31
B. “content generation operations”.....	31
C. “transformation operations”	34
IX. GROUNDS OF UNPATENTABILITY	37
A. Ground 1: Tso in view of Huang Renders Obvious Claims 1-5	39
1. U.S. 6,421,733 (“Tso”) Overview	39
2. U.S. 6,438,576 (“Huang”) Overview and Motivation to Modify Tso with Huang’s Teachings	45
3. Invalidity of Claims 1-5 Over Tso in view of Huang.....	58
(a) Element [1.pre]	58
(b) Element [1.a].....	60
(c) Element [1.b]	62

	(d) Element [1.c].....	65
	(e) Element [1.d]	73
	(f) Element [1.e].....	75
	(g) Element [1.f]	80
	(h) Element [2.pre]	82
	(i) Element [2.a].....	82
	(j) Element [2.b]	83
	(k) Element [2.c].....	84
	(l) Element [2.d]	85
	(m) Element [2.e].....	85
	(n) Element [2.f]	86
	(o) Element [2.g]	86
	(p) Claims 3 and 4	87
	(q) Element [5.a].....	89
	(r) Element [5.b]	91
B.	Ground 2: Tso in view of Huang and Lawler Renders Obvious Claims 6-7	92
	1. U.S. 5,905,522 (“Lawler”) Overview.....	93
	2. Motivations to Modify Tso in view of Huang with Lawler’s Teachings.....	95
	3. Invalidity of Claims 6-7 Over Tso in view of Huang with Lawler’s Teachings.....	100
	(a) Element [6.a].....	100
	(b) Element [6.b]	103
	(c) Element [6.c].....	105
	(d) Element [7.a].....	106
	(e) Element [7.b]	107
C.	Grounds 3 and 4: Samaniego Anticipates and Renders Obvious Claims 1-7	109
	1. Overview of Samaniego	109
	2. Invalidity of Claims 1-7 in View of Samaniego.....	111
	(a) Element [1.pre]	111
	(b) Element [1.a].....	113
	(c) Element [1.b]	113
	(d) Element [1.c].....	115
	(e) Element [1.d]	116
	(f) Element [1.e].....	117

(g)	Element [1.f]	119
(h)	Element [2.pre]	119
(i)	Element [2.a].....	119
(j)	Element [2.b]	120
(k)	Element [2.c].....	121
(l)	Element [2.d]	122
(m)	Element [2.e].....	122
(n)	Element [2.f]	122
(o)	Element [2.g]	123
(p)	Claims 3 and 4	123
(q)	Claim 5.....	124
(r)	Element [6.a].....	126
(s)	Element [6.b]	127
(t)	Element [6.c].....	127
(u)	Claim 7.....	128
D.	Ground 5: Samaniego in View of Tso Renders Obvious Claims 2-4.....	130
E.	Ground 6: Samaniego in View of Lawler Renders Obvious Claims 6-7	134
X.	SECONDARY CONSIDERATIONS.....	139
XI.	CONCLUSION	140

I. INTRODUCTION

1. I have been retained by Akamai Technologies, Inc. (I will refer to them as “Akamai”) to provide my opinions on certain issues related to U.S. Patent No. 9,158,745 (the “’745 patent,” which I understand has been designated as Exhibit 1001) in connection with the above-captioned *inter partes* review (IPR) proceeding. In particular, I have been asked to provide my insights, analysis, and opinions regarding whether claims 1-7 of the ’745 patent (the “Challenged Claims”) are disclosed by and/or obvious in view of the prior art references identified below.

2. I understand the ’745 patent is titled “Optimization of Media Content Using Generated Intermediate Media Content,” identifies as its named inventors Sean Barger, Brian Rice, Matt Butler, and David Pochron, and is currently owned by Equil IP Holdings LLC. I have considered the ’745 patent.

3. I understand that the file history of the ’745 patent has been designated as Exhibit 1002. I have considered this file history, and I will refer to it as the “’745 File History” or by its exhibit number.

4. I understand the ’745 patent was filed as U.S. Patent Application 13/752,110 on January 28, 2013.

5. I have considered the prior art cited in my declaration, including:

- a. **U.S. Patent No. 6,421,733** to Tso (“Tso”), titled “System for Dynamically Transcoding Data Transmitted Between

Computers,” filed September 8, 1997 and issued July 16, 2002. I understand that a copy of this patent has been designated as Exhibit 1004.

- b. **U.S. Patent No. 6,438,576** to Huang (“Huang”), titled “Method and Apparatus of a Collaborative Proxy System for Distributed Deployment of Object Rendering,” filed on March 29, 1999, and issued August 20, 2002. I understand that a copy of this publication has been designated as Exhibit 1005.
- c. **U.S. Patent No. 5,905,522** to Lawler (“Lawler”), titled “Resource Allocation Method for Interactive Televideo System,” filed August 31, 1995, and issued May 18, 1999. I understand that a copy of this publication has been designated as Exhibit 1006.
- d. **U.S. Publication No. 2002/0078093** to Samaniego (“Samaniego”), titled “Automated Media Delivery System,” filed August 14, 2001, and published June 20, 2002. I understand that a copy of this publication has been designated as Exhibit 1007.

6. I am being compensated by Akamai at my standard hourly consulting rate of \$650 for my time on this matter. My compensation is not dependent on the outcome of this proceeding.

7. In forming my opinions, I relied on the documents cited in this declaration and the documents identified in the attached **Appendix B**. These documents comprise patents, file histories, printed publications, and other related documents. As discussed below, each document is a type that experts in my field would have reasonably relied upon when forming their opinions. Further, experts in my field would have had access to each document either through the applicable patent offices and/or well-known libraries, conferences, or publications in the field. My opinions are also based upon my personal and professional experience.

II. QUALIFICATIONS

8. 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.

9. I am a tenured Full Professor in the Colleges of Computing and Engineering at Georgia Tech. I am knowledgeable and familiar with software engineering, web applications, cloud computing, data analytics, wireless communications, microprocessor architecture, hardware, RF, cellular networks, ASIC design, computer engineering, embedded systems, digital signal processing,

and associated software and firmware design. I have created and taught undergraduate and graduate courses in hardware and software design for computer science and engineering applications at Georgia Tech for the past thirty 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 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.

10. 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 **Appendix A**. I graduated more than 20 Ph.D. students that now work as professors or in technical positions around the world.

11. 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).

12. 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.

13. 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.

14. 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 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.

15. 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.

16. My experience is relevant to this case. In addition to authoring textbooks and research papers in the area of cloud computing, virtualization, web-

services, and web-based applications, I have developed web-based document management systems and also web-based fintech applications. I am listed as an inventor on over a dozen patents on web-based applications. I also invented a new protocol that replaces the HTTP protocol for token-based networks, called VTTP, that is a subject of a few issued US patents.

17. I have been extensively involved in the activities of one of the premier standard setting organizations in the world, the IEEE, since the 1980s, and I have participated in the development of standards for hardware design and description languages, such as VHDL, used in design of computer chips – IEEE 1076.6. I was also a member of the IEEE Press Board from 1995-1997. I was the Technical Program Chair of IEEE ICASSP in 1996. I was also the Associate Editor of IEEE Transactions on Circuits & Systems II from 1994 to 1997. I am also the director of the IEEE Atlanta Chapter, and the author of several drafts of IETF proposals.

18. I have 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. 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.

19. 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 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.

20. I have submitted approximately 100 invention disclosures and provisional patents over the past ten years. I am listed as a first inventor on over thirty issued US patents.

21. I have testified as an expert witness before. Over the past two years, I've testified as an expert in more than 20 proceedings, and several of these were in the area of enterprise, cloud and web-based software applications.

22. I have attached a more detailed list of my qualifications as **Appendix A**.

III. LEGAL UNDERSTANDING

A. My Understanding of Claim Construction

23. I have been informed that patent claims are construed from the viewpoint of a person having ordinary skill in the art of the patent at the time of the invention. I have been informed that patent claims generally should be interpreted consistent with their plain and ordinary meaning as understood by a person having ordinary skill in the art in the relevant time period (*i.e.*, at the time of the purported invention, or the so called “effective filing date” of the patent application), after reviewing the patent claim language, the specification, and the prosecution history (*i.e.*, the intrinsic record).

24. I have further been informed that a person having ordinary skill in the art must read the claim terms in the context of the claim itself, as well as in the context of the entire patent specification. I understand that in the specification and prosecution history, the patentee may specifically define a claim term in a way that differs from the plain and ordinary meaning. I understand that the prosecution history of the patent is a record of the proceedings before the U.S. Patent and Trademark Office, and may contain explicit representations or definitions made during prosecution that affect the scope of the patent claims. I understand that an applicant may, during the course of prosecuting the patent application, limit the scope of the claims to overcome prior art or to overcome an examiner’s rejection, by

clearly and unambiguously arguing to overcome or distinguish a prior art reference, or clearly and unambiguously disavowing claim coverage.

25. In interpreting the meaning of the claim language, I understand that a person having ordinary skill in the art may also consider “extrinsic” evidence, including expert testimony, inventor testimony, dictionaries, technical treatises, other patents, and scholarly publications. I understand this evidence is considered to ensure that a claim is construed in a way that is consistent with the understanding of those of ordinary skill in the art at the time of the claimed invention. For example, this can be useful for a technical term whose meaning may differ from its ordinary English meaning. I understand that extrinsic evidence may not be relied on if it contradicts or varies the meaning of claim language provided by the intrinsic evidence, particularly if the applicant has explicitly defined a term in the intrinsic record.

B. My Understanding of Anticipation

26. I understand that a patent claim is invalid as “anticipated” if all of the limitations of the claim are present, either expressly or inherently, in a single pervious device, or described, either expressly or inherently, in a single prior art publication or patent. I understand that, in the relevant time period, 35 U.S.C. § 102 stated that a person shall be entitled to a patent unless:

- The invention was known or used by others in this country, or patented or described in a printed publication in this or a foreign country, before the invention thereof by the applicant for patent, or
- The invention was patented or described in a printed publication in this or a foreign country or in public use or on sale in this country, more than one year prior to the date of the application for patent in the United States, or
- He has abandoned the invention, or
- The invention was first patented or caused to be patented, or was the subject of an inventor's certificate, by the applicant or his legal representatives or assigns in a foreign country prior to the date of the application for patent in this country on an application for patent or inventor's certificate filed more than twelve months before the filing of the application in the United States, or
- The invention was described in—(1) an application for patent, published under section 122(b), by another filed in the United States before the invention by the applicant for patent or (2) a patent granted on an application for patent by another filed in

the United States before the invention by the applicant for patent, except that an international application filed under the treaty defined in section 351(a) shall have the effects for the purposes of this subsection of an application filed in the United States only if the international application designated the United States and was published under Article 21(2) of such treaty in the English language; or

- He did not himself invent the subject matter sought to be patented, or
- (1) during the course of an interference conducted under section 135 or section 291, another inventor involved therein establishes, to the extent permitted in section 104, that before such person's invention thereof the invention was made by such other inventor and not abandoned, suppressed, or concealed, or (2) before such person's invention thereof, the invention was made in this country by another inventor who had not abandoned, suppressed, or concealed it. In determining priority of invention under this subsection, there shall be considered not only the respective dates of conception and reduction to practice of the invention, but also the reasonable

diligence of one who was first to conceive and last to reduce to practice, from a time prior to conception by the other.

27. I also understand that a prior art reference can disclose a claimed feature because the feature is expressly described or because the feature is inherent in the disclosure. I understand that something is inherent in a prior art reference if the missing descriptive matter must necessarily be present, not merely probably or possibly present, and it would be so recognized by a person of ordinary skill in the art as of the priority date of the patent.

28. Claim limitations that are not expressly found in a prior art reference are inherent if the prior art necessarily functions in accordance with, or includes, the claim limitations. I understand that it is acceptable to examine evidence outside of the prior art reference (i.e., extrinsic evidence) in determining whether a feature, while not expressly discussed in the reference, is necessarily present in it.

C. My Understanding of Obviousness

29. I understand that a claim may be invalid if the subject matter described by the claim as a whole would have been obvious to a hypothetical person of ordinary skill in the art in view of a prior art reference or in view of a combination of references at the time the claimed invention was made. Therefore, I understand that obviousness is determined from the perspective of a hypothetical person of ordinary skill in the art and that the asserted claims of the patent should be read from

the point of view of such a person at the time the claimed invention was made. I further understand that a hypothetical person of ordinary skill in the art is assumed to know and to have all relevant prior art in the field of endeavor covered by the patent in suit.

30. I have been informed that there are two criteria for determining whether prior art is analogous and thus can be considered prior art: (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 patentee's endeavor, whether the reference still is reasonably pertinent to the particular problem with which the patentee is involved. I have also been informed that the field of endeavor of a patent is not limited to the specific point of novelty, the narrowest possible conception of the field, or the particular focus within a given field. I have also been informed that a reference is reasonably pertinent if, even though it may be in a different field from that of the patentee's endeavor, it is one which, because of the matter with which it deals, logically would have commended itself to a patentee's attention in considering their problem.

31. I have also been advised that an analysis of whether a claimed invention would have been obvious should be considered in light of the scope and content of the prior art, the differences (if any) between the prior art and the claimed invention,

and the level of ordinary skill in the pertinent art involved. I understand as well that a prior art reference should be viewed as a whole.

32. I have also been advised that in considering whether a claimed invention could be obvious over a combination of prior art references, I may assess whether there are apparent reasons to combine known elements in the prior art in the manner claimed in view of interrelated teachings of multiple prior art references, the effects of demands known to the design community or present in the marketplace, and/or the background knowledge possessed by a person having ordinary skill in the art. I understand that other principles may be relied on in evaluating whether a claimed invention would have been obvious, and that these principles include the following:

- A combination of familiar elements according to known methods is likely to be obvious when it does no more than yield predictable results;
- When a device or technology is available in one field of endeavor, design incentives and other market forces can prompt variations of it, either in the same field or in a different one, so that if a person having ordinary skill can implement a predictable variation, the variation is likely obvious;

- If a technique has been used to improve one device, and a person having ordinary skill in the art would recognize that it would improve similar devices in the same way, using the technique is obvious unless its actual application is beyond their skill;
- An explicit or implicit teaching, suggestion, or motivation to combine two prior art references to form the claimed combination may demonstrate obviousness, but proof of obviousness does not depend on or require showing a teaching, suggestion, or motivation to combine;
- Market demand, rather than scientific literature, can drive design trends and may show obviousness;
- In determining whether the subject matter of a patent claim would have been obvious, neither the particular motivation nor the avowed purpose of the named inventor controls whether the claim is obvious;
- One of the ways in which a patent's subject can be proved obvious is by noting that there existed at the time of invention a known problem for which there was an obvious solution encompassed by the patent's claims;

- Any need or problem known in the field of endeavor at the time of invention and addressed by the patent can provide a reason for combining the elements in the manner claimed;
- “Common sense” teaches that familiar items may have obvious uses beyond their primary purposes, and in many cases a person having ordinary skill will be able to fit the teachings of multiple patents together like pieces of a puzzle;
- A person having ordinary skill in the art is also a person having ordinary creativity, and is not an automaton;
- A patent claim can be proved obvious by showing that the claimed combination of elements was “obvious to try,” particularly when there is a design need or market pressure to solve a problem and there are a finite number of identified, predictable solutions such that a person having ordinary skill in the art would have had good reason to pursue the known options within his or her technical grasp; and
- One should be cautious of using hindsight in evaluating whether a claimed invention would have been obvious.

33. I further understand that, in making a determination as to whether the claimed invention would have been obvious to a person having ordinary skill, the

Board may consider certain objective factors if they are present, such as: commercial success of products practicing the claimed invention; long-felt but unsolved need; teaching away; unexpected results; copying; and praise by others in the field. These factors are generally referred to as “secondary considerations” or “objective indicia” of nonobviousness. I understand, however, that for such objective evidence to be relevant to the obviousness of a claim, there must be a causal relationship (called a “nexus”) between the claim and the evidence and that this nexus must be based on a novel element of the claim rather than something in the prior art. I also understand that even when they are present, secondary considerations may be unable to overcome primary evidence of obviousness (such as motivation to combine with predictable results) that is sufficiently strong.

34. I have been asked to consider the validity of claims 1-7 of the '745 patent, which I refer to as the “Challenged Claims.” I understand that for *inter partes* reviews, invalidity must be shown under a preponderance of the evidence standard. I have concluded that each of the Challenged Claims is anticipated by and rendered obvious by the prior art based on the references described below and as explained herein.

IV. BACKGROUND OF THE TECHNOLOGY

35. The '745 patent's purported invention as recited in challenged claims 1-7 of the patent is a method of processing web media content via a cache for

playback in a network. The claims purport to provide a method for automated graphics delivery in which graphics can be modified from an original file based on specified parameters. To this end, the Claims recite receiving a content request including information specifying “content generation” and “transformation” operations. The system determines whether an “intermediate” content version created using the “content generation” operations is already cached and, if so, generates and applies the “transformation” operations to that version or, if not, applies both operations to an original version of the content.

36. Based on my experience in the field, however, the functionalities that are claimed in the '745 patent were well-known in the art prior to the '745 patent's alleged priority date. One example of such prior art is U.S. Patent No. 6,521,733, which was originally assigned to Intel, and identifies Tso as its first named inventor. I understand that a copy of Tso has been designated as Exhibit 1004. Tso discloses that a system receives a packet from a client requesting media content, and that the packet contains “selection criteria” in, for example, the header, body, or both, of the packet. Tso, 6:64-7:62. The selection criteria specify the properties of the requested content, including properties such as the “data type [and] type of encoding/compression” of the content that should be used to generate the media content as well as criteria specifying operations to further transform the content, including “the degree of alteration” of the content that is required. *Id.* Tso also

describes a server requesting a specific, “particular version of” a media object using a “GetScaledObject” call, and teaches using this call to retrieve a “particular version” of content such as a version that has already been processed and cached. Tso, 6:9-13. Tso further teaches that, if the requested version of the media content already exists, then the existing content item is returned to the requestor. Tso, 4:1-5. If the version of content that was requested does not already exist, then the original content item is retrieved and transcoded to generate the requested content. Tso, 6:9-50.

37. Another relevant prior art reference is U.S. Patent No. 6,438,576, which was originally assigned to IBM, and which identifies Huang as the first named inventor. I understand that a copy of Huang has been designated as Exhibit 1005. Huang teaches retrieving a partially-rendered, intermediate media item from a cache and completing required rendering for that media item using this retrieved cached version of the media item. Huang, 7:23-8:11. As explained in greater detail below in Section IX.A.2, a person having ordinary skill in the art would have been motivated to modify Tso with Huang’s teachings to advantageously reduce the unnecessary re-transcoding and re-retrieval of media content. Applying Huang’s teachings to Tso yields a system that retrieves an already partially-rendered content item from a cache, and then uses that retrieved item to complete any remaining, necessary rendering for the client.

38. Another example disclosure that I have considered is U.S. Patent Application Publication No. 2002/0078093, which is a publication of Application No. 09/929,904, and which identifies Samaniego as the first named inventor. I understand that a copy of Samaniego has been designated as Exhibit 1007. As can be seen from comparing the cover pages of Samaniego and the '745 patent, Samaniego does not share any common inventors with the '745 patent. I have been instructed to assume that Samaniego qualifies as prior art to the '745 patent, and have not formulated my own separate opinion as to whether Samaniego qualifies as prior art. Samaniego's specification is substantively identical to the '745 patent's specification. I have compared the text of Samaniego and the text of the '745 patent and generated a redline comparison between the specifications. I understand that a copy of that redline comparison has been designated as Exhibit 1013. I have identified specification support in Samaniego that discloses and renders obvious claims 1-7 of the '745 patent. Additionally, as set forth in the prosecution history for the '745 patent, the Examiner considered whether Samaniego provides support for the Challenged Claims, and stated that it does provide "support for the claimed subject matter." *See* Ex. 1002 ('745 FH), 165.

39. None of the features recited in the Challenged Claims of the '745 patent was novel at the time of the alleged invention, and there was nothing novel as to how the alleged inventors of the '745 patent combined those features. As I discuss below,

the Challenged Claims are anticipated by and obvious in view of the prior art. Further, there were no technological barriers to combining these features to form the claimed invention, and a person having ordinary skill in the art would have been motivated to combine these elements and would have understood that the combination would have yielded predictable results.

V. THE '745 PATENT AND PROSECUTION HISTORY

A. '745 Patent Overview

40. The Challenged Claims of the '745 patent are method claims that generally recite methods in which media is processed into a final version based on intermediate versions of content items that are stored in a cache. '745 patent, 1:22-26, 17:39-41, 23:9-39. Aspects of an embodiment of a system in which the claimed methods in the Challenged Claims are implemented are shown in the annotated version of Figure 21 below.

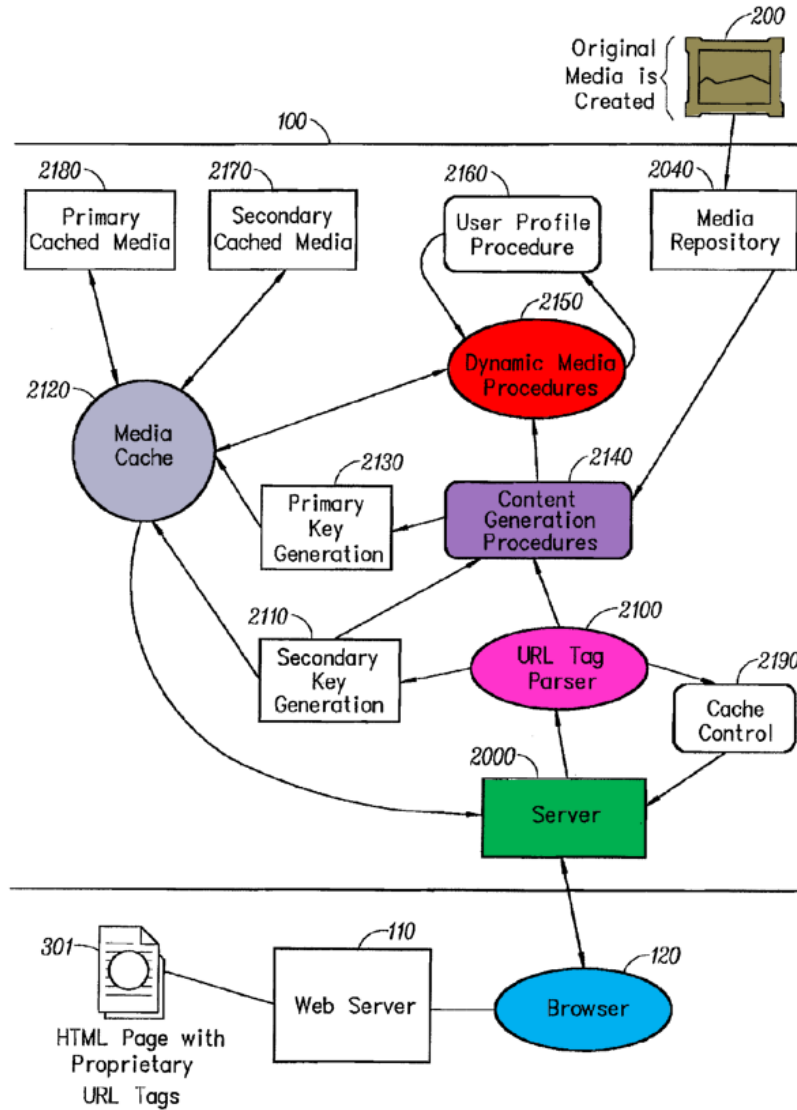


FIG. 21

41. As shown, the **server 2000**, annotated in green operates as a claimed host computer and receives a request from user's **browser 120**. '745 patent, 19:7-15. After receiving this request, the server extracts information from the request, including information that, for example, identifies the **original media content 200** requested by the user, as well as content generation operations, shown as **content generation procedures 2140**, and transformation operations, shown as **dynamic**

media procedures 2150 (which are used to perform “dynamic content processing” operations), for that content. ’745 patent, 19:15-23. The server uses a **URL tag parser 2100** to determine whether a fully processed version of the requested media, which is referred to in the ’745 patent as a “primary” version of the requested media, has been cached in **media cache 120**. ’745 patent, 19:24-27. If the content has been cached, then the cached version of the content is then returned to the requesting user. *Id.* Otherwise, the **URL tag parser 2100** searches the **media cache 120** for a version of the media for which content generation operations have already been applied, which the ’745 patent refers to as “intermediate” media. ’745 patent, 19:27-36, 20:1-2. If it is able to find a cached intermediate media, the **server 2000** applies the transformation operations (**dynamic media procedures 2150**) to generate the requested media, and returns that media to the user’s **browser 120**. *Id.* If the system has not cached any such intermediate media, the requested media is generated by applying the content generation operations (**content generation procedures 2140**) and then the transformation operations (**dynamic media procedures 2150**) to a retrieved copy of the **original media content 200** to generate the final version of the requested media, which is then returned to the user’s **browser 120**. *Id.* 20:1-2.

B. Prosecution History of the ’745 Patent

42. In formulating my opinions, I considered the contents of the prosecution history of the ’745 patent, a copy of which I understand has been

designated as Exhibit 1002. The prosecution history shows that the Challenged Claims of the '745 patent were allowed after the first office action, without any rejections. Ex. 1002, 156-57. However, during prosecution there was an interview and examiner's amendment. *Id.*, 158-169. The "Examiner-Initiated Interview Summary" states that there was no prior art discussed, and instead the Examiner and Applicant discussed corrections to the claims to address objections that the specification would not otherwise disclose certain features in the claims. Ex. 1002, 168-69.

43. Following the interview, the Examiner issued a Notice of Allowance, along with an Examiner's Amendment that addressed the language raised during the interview. Ex. 1002, 156-67. In the Examiner's Amendment accompanying the Notice of Allowance, the Examiner also considered the Applicant's claims to priority to earlier applications. Of note, the Examiner stated:

[T]his application does not receive the prior date of App. 09425326 filed 21 October 1999 since it claims additional disclosure not presented in the prior application. Furthermore, ***support for the claimed subject matter has been found in 12238842, 12173747, 11269916 and 09929904. Thus, 13752110 receives the priority date of App. 09929904.***

Ex. 1002, 165 (emphasis added). The number "09929904" is a reference to the application number for Samaniego, which states on the cover that it a publication of Application No. 09/929,904. Ex. 1007, Samaniego, Cover. The number

“13752110” refers to the ’745 patent, which states on its cover that it issued from Application No. 13/752,110. Ex. 1001, 1.

44. After this initial notice of allowance, the ’745 patent then later issued on October 13, 2015, after the Examiner issued a supplemental Notice of Allowance on September 8, 2015. ’745 FH, 267-78.

VI. THE ’745 PATENT PRIORITY DATE

45. The ’745 patent matured from U.S. Application 13/752,110, filed on January 28, 2013. The ’745 patent claims priority to U.S. Patent Applications 09/425,326 (filed October 21, 1999), 09/929,904 (filed August 14, 2001), 11/269,916 (filed November 7, 2005), 12/173,747 (filed July 15, 2008), and 12/238,842 (filed September 26, 2008).

46. I have not formulated any opinion about the correct priority date for the ’745 patent. For purposes of this declaration, I was asked to assume, with respect to each of the prior art Grounds discussed in this declaration, that the ’745 patent is entitled to the priority date of November 7, 2005.

47. With respect to my opinions regarding both claim construction and prior art Grounds 1 and 2 set forth below, I was also asked to consider whether my opinions would be affected if the ’745 patent was instead entitled to a priority date no later than October 21, 1999. My claim construction opinions and opinions with respect to prior art Grounds 1 and 2 below would not be materially impacted if the

'745 were entitled to this October 21, 1999 date, and these opinions as set forth in this declaration apply equally under either priority date.

VII. LEVEL OF ORDINARY SKILL IN THE ART

48. In rendering the opinions set forth in this declaration, I was asked to consider the patent claims and the prior art through the eyes of a person having ordinary skill in the art at the time of the alleged invention. I understand that the factors considered in determining the ordinary level of skill in a field of art include the level of education and experience of persons working in the field; the types of problems encountered in the field, the teachings of the prior art, and the sophistication of the technology at the time of the alleged invention. I understand that a person having ordinary skill in the art is not a specific real individual, but rather is a hypothetical individual having the qualities reflected by the factors above. I understand that a person having ordinary skill in the art would also have knowledge from the teachings of the prior art, including the art cited below.

49. With respect to all of the Grounds, I have been asked to assume that the '745 patent is entitled to the priority date of November 7, 2005. Upon consideration of these factors, in my opinion, on or before this priority date, a person having ordinary skill in the art relating to the technology of the '745 patent would have had a minimum of either a bachelor's degree in computer systems, computer science, or the equivalent thereof, and at least two years of experience with networked media

delivery or related technologies. Additional graduate education could substitute for professional experience, or significant experience in the field could substitute for formal education.

50. I have also been asked to consider whether my understanding of a person of ordinary skill in the art would be different if the '745 patent were entitled to a priority date no later than October 21, 1999 rather than November 7, 2005. My definition of a person of ordinary skill in the art would remain the same with respect to the October 21, 1999 priority date, and any difference in the level of ordinary skill between these priority dates does not materially impact my opinions set forth in this declaration.

51. Regardless of which priority date applies, I am qualified to opine concerning what a person having ordinary skill in the art would have known and understood during that time. Well before both November 7, 2005 and October 21, 1999, my level of skill in the art was at least that of a person having ordinary skill. I am qualified to opine concerning what a person having ordinary skill in the art would have known and understood at the time of the '745 patent's purported invention. My analysis and conclusions herein are from the perspective of a person having ordinary skill in the art as of November 7, 2005.

VIII. CLAIM CONSTRUCTION

52. I have been instructed that the Challenged Claims 1-7 of the '745 patent are to be given their ordinary and customary meaning as understood by one of ordinary skill in the art at the time of the alleged invention in light of the specification and the prosecution history pertaining to the patent. I have followed these principles in my analysis set forth in this document.

53. I understand that there is a pending district court litigation involving the '745 patent, and the parties have not engaged in claim construction in that litigation. I reserve the right to supplement my opinions in this declaration to the extent that the parties later propose claim constructions for any claim terms in the '745 patent.

A. Preambles

54. I understand the preamble of a claim may or may not be limiting. For purposes of this declaration, I have not been asked to formulate any opinion as to whether any claim preamble is limiting, and have not formulated any such opinion. Instead, I have been instructed to consider whether the prior art discloses the preambles, to the extent they are limiting. To the extent they are limiting, they are disclosed by the prior art as discussed in Section IX.

B. “content generation operations”

55. The term “content generation operations” appears in the following elements of the Challenged Claims:

- a. [1.b] (“... the first request contains information, the information indicating a first original media content, first ***content generation operations***, and first transformation operations”)
- b. [1.c] (“... the previously-generated first intermediate media content having been created using the first original media content and the first set of ***content generation operations***”)
- c. [1.e] (“...creating a first optimized media content for the first playback device by creating a first intermediate content using the first original media content and the first set of ***content generation operations***”)
- d. [2.a] (“... the second request contains information, the information indicating a second original media content, second set of ***content generation operations***, and second set of transformation operations”)
- e. [2.d] (“... determining whether a second intermediate media content is available having been previously created using the second original media content and the second set of ***content generation operations***”)
- f. [2.f] (“... creating a second optimized media content for the first playback device by creating the second intermediate content using the second original media content and the second set of ***content generation operations***, and performing the second set of transformation operations on the second intermediate media content”)

- g. [3] (“... the first set of *content generation operations* and the second set of *content generation operations* are the same”)
- h. [4] (“... the first set of *content generation operations* and the second set of *content generation operations* are different”)
- i. [5.a] (“... the previously-generated first optimized media content was created using the first original media content, the first *content generation operations*, and the first transformation operations”)
- j. [6.b] (“... determining an alternate first set of *content generation operations* or an alternate first set of transformation operations”);
- k. [6.c] (“... creating an alternate first optimized media content using the alternate first set of *content generation operations* or the alternate first set of transformation operations”)

56. The exact term “content generation operations” does not appear in the specification, but the specification does use similar phrases including “content generation procedures” (*e.g.*, ’745 patent, 20:48-21:14, 22:43-67; *see also id.* at Figs. 21-23, 5:65-6:1, 7:6-29, 17:16-39, 18:14-33, 18:41-43, 19:5-43, 20:3-39) and “content creation commands” (*e.g.*, ’745 patent, 21:15-22:39). Based on the context around the discussion of “content creation commands,” both this term and the term “content generation procedures” appear to be referring to the same types of commands, both of which are claimed “content generation operations.” For

example, when discussing the “tags” used for “content generation,” the ’745 patent uses the term “content generation procedure” to describe the processing of the image. *E.g.*, ’745 patent, 20:27-21:14. When discussing the types of commands implemented using these tags, the ’745 refers to the commands as “Content Creation Commands.” ’745 patent, 21:15-22:38. The specification does not use the term “content creation commands” outside of this discussion, further confirming that “content generation procedures” and “content creation commands” refer to the same processing steps. *See id.* The specification provides several examples of “content creation commands” or “content generation procedures” including, for example, converting a media content item to a specified type/bit-depth, scaling a media content item to a specified size, and saving the media content item to a specified file. ’745 patent, 21:16-22:39. Thus, regardless of the full scope of the term “content generation operations,” a person having ordinary skill in the art would have understood it to encompass at least the “content creation commands” that are described in the ’745 patent’s specification, including the exemplary commands that I have identified above. As discussed in Sections IX.A and IX.C, the prior art discloses these same content generation operations.

C. “transformation operations”

57. The term “transformation operations” appears in the following elements of the Challenged Claims:

- a. [1.b] (“... the first request contains information, the information indicating a first original media content, first content generation operations, and first *transformation operations*”)
- b. [1.d] (“... creating a first optimized media content for the first playback device by performing the first set of *transformation operations* on the previously-generated first intermediate media content”)
- c. [1.e] (“... creating a first optimized media content for the first playback device by creating a first intermediate content using the first original media content and the first set of content generation operations, and performing the first set of *transformation operations* on the first intermediate media content”)
- d. [2.a] (“... the second request contains information, the information indicating a second original media content, second set of content generation operations, and second set of *transformation operations*”);
- e. [2.e] (“... creating a second optimized media content for the first playback device by performing the second set of *transformation operations* on the second intermediate media content”)
- f. [2.f] (“... creating a second optimized media content for the first playback device by creating the second intermediate content using the second original media content and the second set of content generation

operations, and performing the second set of *transformation operations* on the second intermediate media content”)

- g. [3] (“... the first set of *transformation operations* and the second set of *transformation operations* are different”)
- h. [4] (“... the first set of *transformation operations* and the second set of *transformation operations* are different”)
- i. [5.a] (“... the previously-generated first optimized media content was created using the first original media content, the first content generation operations, and the first *transformation operations*”)
- j. [6.b] (“... determining an alternate first set of content generation operations or an alternate first set of *transformation operations*”)
- k. [6.c] (“... creating an alternate first optimized media content using the alternate first set of content generation operations or the alternate first set of *transformation operations*”)

58. The ’745 patent’s specification does not provide an explicit definition of the term “transformation operations.” Indeed, outside of the patent’s claims, the ’745 patent uses a form of the word “transform” only once, in the context of describing a prior art product. *See* ’745 patent, 3:56. However in addition to the “content creation commands” that are disclosed in the specification, the specification also discloses “proprietary tags” that are used to implement a large set of different

example “media processing script commands.” ’745 patent, 10:27-15:20. These media processing script commands describe a variety of different types of transformations to be applied to content, including commands such as a “Set Resolution” command (two thirds of the way down column 13 in Table B), as well as several other transformations such as commands to “Colorize,” and so on. *Id.*

59. Thus, regardless of the full scope of the term “transformation operations,” a person having ordinary skill in the art would have understood it to at least encompass the “media processing script commands” identified in the ’745 patent’s specification, including the exemplary commands that I have identified above. As discussed in Sections IX.A and IX.C, the prior art discloses these same transformation operations.

IX. GROUNDS OF UNPATENTABILITY

60. As I explained above in Section IV, the methods for developing transformation processing operations methods recited in Challenged Claims 1-7 were well-known in the art.

61. In this section, I will explain why Samaniego anticipates and renders obvious, and combinations of teachings from the following references render obvious, each of the Challenged Claims:

Name	US Patent/ Publication	Exhibit	Filed	Published/ Issued
Tso	6,421,733	1004	9/8/1997	7/16/2002
Huang	6,438,576	1005	4/29/1999	8/20/2002
Lawler	5,905,522	1006	8/31/1995	5/18/1999
Samaniego	2002/0078093	1007	8/14/2001	6/20/2002

I have been instructed to assume that each of these references qualifies as prior art.

62. Specifically, as I explain below, the Challenged Claims are invalid based upon the following prior art grounds:

- a. **Ground 1:** Challenged Claims 1-5 of the '745 patent are rendered obvious by Tso in view of Huang;
- b. **Ground 2:** Challenged Claims 6-7 are rendered obvious by Tso in view of Huang and Lawler;
- c. **Ground 3:** Challenged Claims 1-7 are anticipated by Samaniego;
- d. **Ground 4:** Challenged Claims 1-7 are rendered obvious by Samaniego;
- e. **Ground 5:** Challenged Claims 2-4 are rendered obvious by Samaniego in view of Tso; and
- f. **Ground 6:** Challenged Claims 6-7 are rendered obvious by the teachings of Samaniego in view of Lawler.

63. It is my opinion that Challenged Claims 1-7 of the '745 patent are unpatentable as explained below. I note that in this discussion, I may focus on the

particular language in specific claims, but it should be understood that my analysis applies to the corresponding elements of all Challenged Claims containing identical or substantially identical language.

A. Ground 1: Tso in view of Huang Renders Obvious Claims 1-5

64. Claims 1-5 of the '745 patent are obvious over Tso in view of Huang. Tso teaches a system that dynamically transcodes and then caches media to be transmitted from a server to a client in response to a request. Huang discloses a method for object rendering (including by transcoding) in which partially-rendered objects are further rendered to complete rendering of an object for a client without re-retrieval or re-rendering of content. Thus, as further set forth below, Tso in view of Huang renders obvious claims 1-5 of the '745 patent.

1. U.S. 6,421,733 (“Tso”) Overview

65. Tso discloses a “system for dynamically transcoding data transmitted between two computers over a communications link.” Tso, 1:13-14. An embodiment of this system is shown in Fig. 3, which is annotated below:

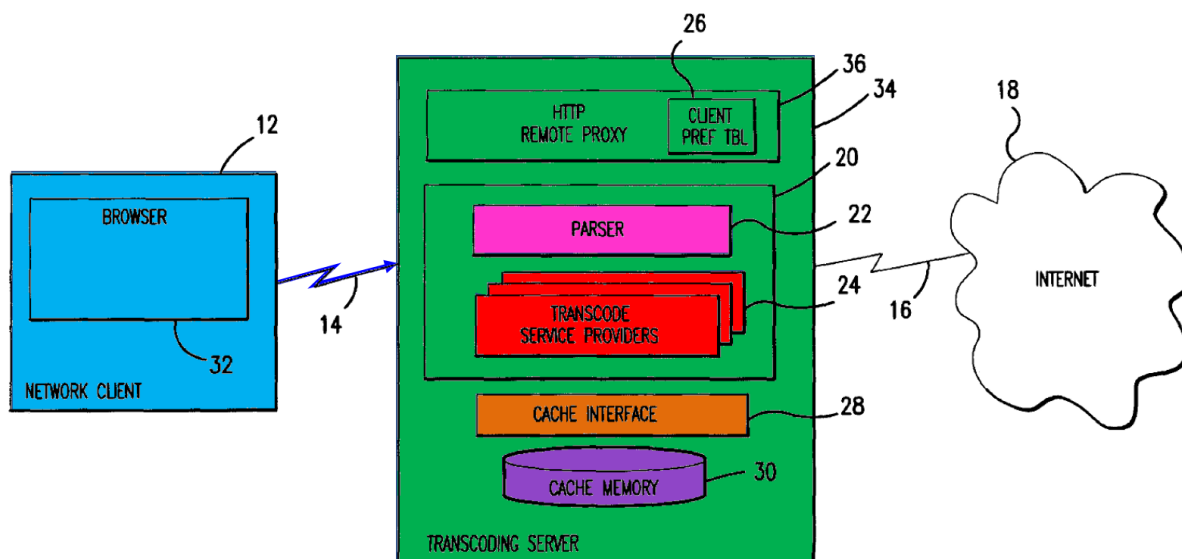


FIG. 3

66. As is shown and described, a **network client** includes a browser and sends a **request for media content, indicated as arrow 14**, such as a web page including “image, video, or HTML” content, to a **transcoding server**. Tso, 3:10-14, 3:45-53, 4:6-10, 9:54-65, 14:1-7. The “request packet” sent by the client carries “information” in both its “header” and “data portion[s]” that includes “selection criterion,” which is used to specify what transcoding should be applied to the content. Tso, 6:64-8:9.¹

¹ Tso sometimes inconsistently uses the terms “selection criterion” and “selection criteria” to collectively refer to either singular selection criterion and combinations of different selection criteria. See, e.g., Tso, 7:20-21 (using the singular “criterion” to refer to a list of criteria: “The predetermined selection criterion may comprise: [examples of selection criteria]”), 8:10-13 (using the plural “criteria” to refer to the same list of criteria: “Applying the above-listed selection criteria, or combinations thereof, embodiments of the present invention may be used to provide a virtually limitless range of dynamic transcoding services.”), 15:36-38 (“location-specific characteristics are to be used as predetermined selection criteria”). I have used

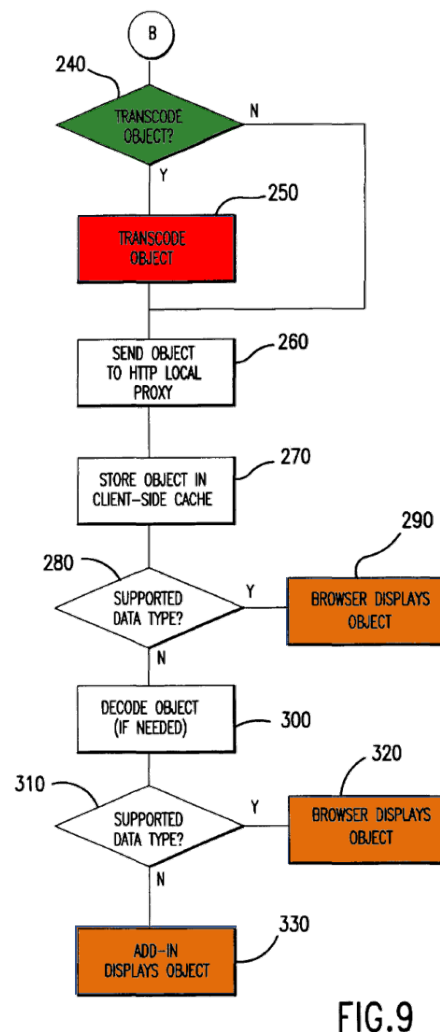
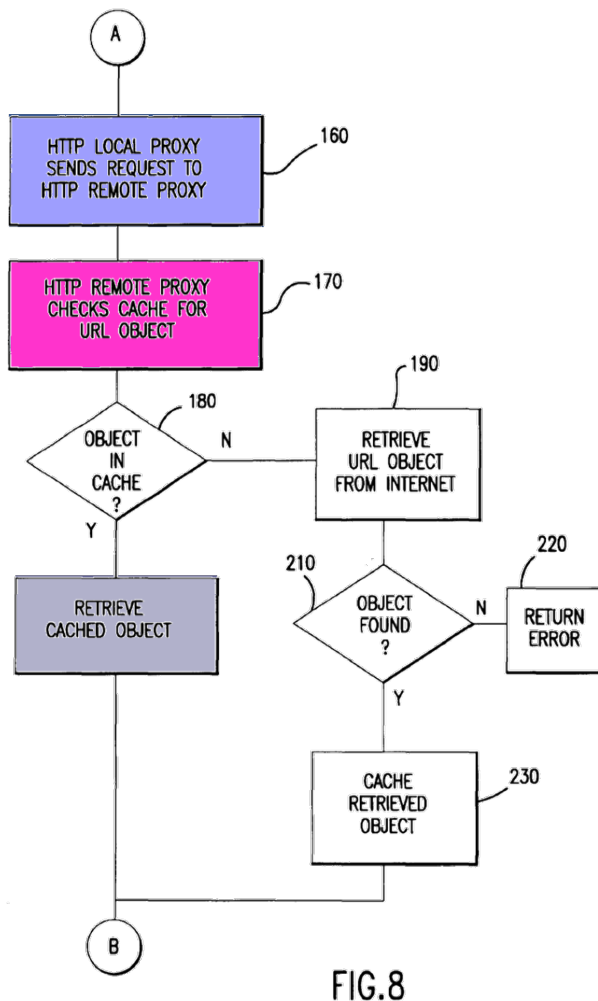
67. Tso further discloses that the request packet stores different selection criteria. For example, selection criteria may specify the “particular [media] content” being requested, as well as “network client” information “such as a display dimension, resolution, number of colors, processor type, memory/disk configuration, modem or network interface type”; these types of selection criteria describe what are referred to as formats of a client presentation system as claimed in the ’745 patent. Tso, 7:4-7, 7:20-33; ’745 patent, 21:33-34, 22:22-38. The selection criteria may also specify “the degree of alteration desired for its content” and other “rules or programs to run for customizing content ... based on user or client characteristics” such as “resolution”; these selection criteria describe parameters that are referred to as media processing script commands or transformation operations in the ’745 patent. Tso, 7:20-29, 7:60-8:9; *see also* ’745 patent, 10:47-15:21 (describing media transformation script commands, which are transformation operations, including “SetResolution” and “Colorize”). The criteria may also indicate the “current processing load” of the device on which the transcoding server is installed, which is “dynamically update[d].” Tso, 6:64-7:14. The system uses these criteria and “combinations thereof” to provide a “virtually limitless range” of “dynamic transcoding services,” including by selecting which of the “**transcode**

“criteria” and “criterion” consistent with Tso’s teachings of these terms, and generally use “criterion” to refer to the singular and “criteria” to refer to the plural.

service providers 24 are invoked” to generate the version of content to be provided in response to the client’s request. Tso, 3:48-65, 7:15-17, 8:10-14.

68. After the request is received by the **transcoding server**, it is handed to a transcoder 20, which includes a **“parser,” “transcode service providers,”** and a **“cache interface”** connected to a **“cache memory.”** Tso, 3:8-17, 3:66-4:5, 4:14-19.

69. Below Figures 8 and 9 are annotated to summarize the relevant portions of the process disclosed by Tso:



70. At step 160, the client **sends a request** to the server for a media content item. Tso 3:11-16, 14:21-25. After receiving this **request**, at **step 170**, the HTTP remote proxy invokes the parser to issue a “GetScaledObject” call to **determine whether a particular version of the requested content item is cached**. Tso, 14:25-30. For example, this request can be used to request a version with a “stage” property specifying the progress of “staged refinements” where the transcoding process is staged into incremental transcoding steps, which not only renders obvious, but discloses using a “stage” parameter to retrieve partially-transcoded content items that are at a partially complete transcoding stage. Tso, 5:36-42, 6:9-23 (“Unlike traditional caching proxies, transcode service providers 24 can use server-side cache memory 30 to store several different versions of an object to support clients with different communications and/or presentation capabilities. Thus, an additional “Stage” parameter may be used to indicate which version of the cached object is to be returned to network client 12.”). If the requested content item is cached, then it is **retrieved from cache**, and **returned to the client**. Tso 14:30-32. If the requested item is not in cache then, when a content item is retrieved, at **step 240 the parser determines whether additional transcoding is required** by using “predetermined selection criteri[a].” Tso, 14:47-50, 7:15-8:9. The system performs any required transcoding by using one or more **transcode service providers**. Tso, 14:50-55, 10:37-49. Each of the different transcode service providers is used to perform

different transcoding operations such that, if there is a need for additional transcoding techniques to be implemented by the system, additional providers are “dynamically added to the system to provide new features and/or better transcoding algorithms.” *Id.*; Tso, 4:23-37.

71. Tso further teaches that as part of the transcoding process, both the “original” version, as well as “several different” “transcoded” “versions of content are cached. Tso, 4:1-5, 6:13-23. Once the requested version of the content is generated and cached, the server uses additional GetScaledObject() requests to retrieve that desired version of the content and, at **step 260**, the server **returns the media content item to be displayed on the client’s browser**. Tso, 6:42-50, 14:52-57.

72. Tso further discloses that the system uses “multiple-thread processing” to “process requests for the same hypertext object from multiple network clients.” Tso, 5:43-50, 6:51-63. Multiple-thread processing allows a device to perform multiple tasks, such as processing requests for content, in parallel, as opposed to single thread processing which requires incoming tasks to be handled serially. For example, Tso teaches that if a server receives requests from two different clients, both of which are requesting different versions of the same media content item, the server can retrieve the content item once from the origin of the content (e.g., the site that originally generated the content), process “appropriate versions” of the item for

the different “client characteristics” that each requester will have (for example, for different types of computers with screens that have different resolutions), and then transmit those differently transcoded versions “to such multiple network clients 12 concurrently.” Tso, 6:51-63, 7:60-8:9.

73. In addition to teaching that transcoding service providers within a server are used for transcoding, Tso also teaches that different devices can distribute the transcoding process such that, for example, a client computer enabled to perform transcoding of partially-transcoded content that is received from a server (Tso, 14:47-15:6), and proxy servers communicating with one another are also able to “collaboratively” complete transcoding of content by having one proxy server provide a partially-completed transcoding of content to another proxy server, which completes transcoding of the partially-transcoded content item (Tso, 15:66-16:14). Tso thus discloses that servers are configured to both send and receive partially-transcoded content, and are also configured to cache content that has been transcoded.

2. U.S. 6,438,576 (“Huang”) Overview and Motivation to Modify Tso with Huang’s Teachings

74. Huang teaches a system and “object rendering method for a collaborative data network” such as the Internet that caches both “partially” and “completely” “rendered objects” in order to avoid repeating rendering operations on a content object. Huang, Abstract, 1:17-22, 6:63-67. Huang uses the term

“rendering” to refer to image manipulation steps that would have been understood by a person having ordinary skill in the art to encompass steps implemented in transcoding. Huang, 2:37-39. For example, Huang describes rendering processes in which content is rendered into “different forms or resolutions,” which describes processes such as scaling or changing the encoding format for content, as well as converting content from color to “black-and-white,” which describes processes in which color resolution content is converted into black and white content. Huang, 1:17-22, 1:57-58, 6:59-63. Tso describes similar processes as “transcoding” processes. For example, Tso teaches using selection criteria to transcode content based on a client device’s “resolution” and “number of colors,” as well as installed hardware and software for “decompression” and the “data type” for content—both of which are used to determine whether the client system can handle transcoded content of different forms. Tso, 7:15-33. Thus, Huang uses the term “rendering” to refer to processing steps including steps commonly referred to as “transcoding” steps in the art. *See id.* Importantly, this stands in contrast to the use of “rendering” as it is sometimes used in the art to describe the display or other performance of previously-generated or transcoded content. Huang separately uses the term “consum[ing]” to refer to “consuming” the content at a client (*e.g.*, preparing and displaying the content to a user) to refer to this latter process. Huang, 1:59-64. In

this declaration, I use the term “rendering” and variations thereof to refer to the “transcoding” taught in Huang as part of the “rendering.”

75. Below, Figure 4 of Huang is annotated to summarize relevant portions of Huang’s teachings of a rendering (including transcoding) process:

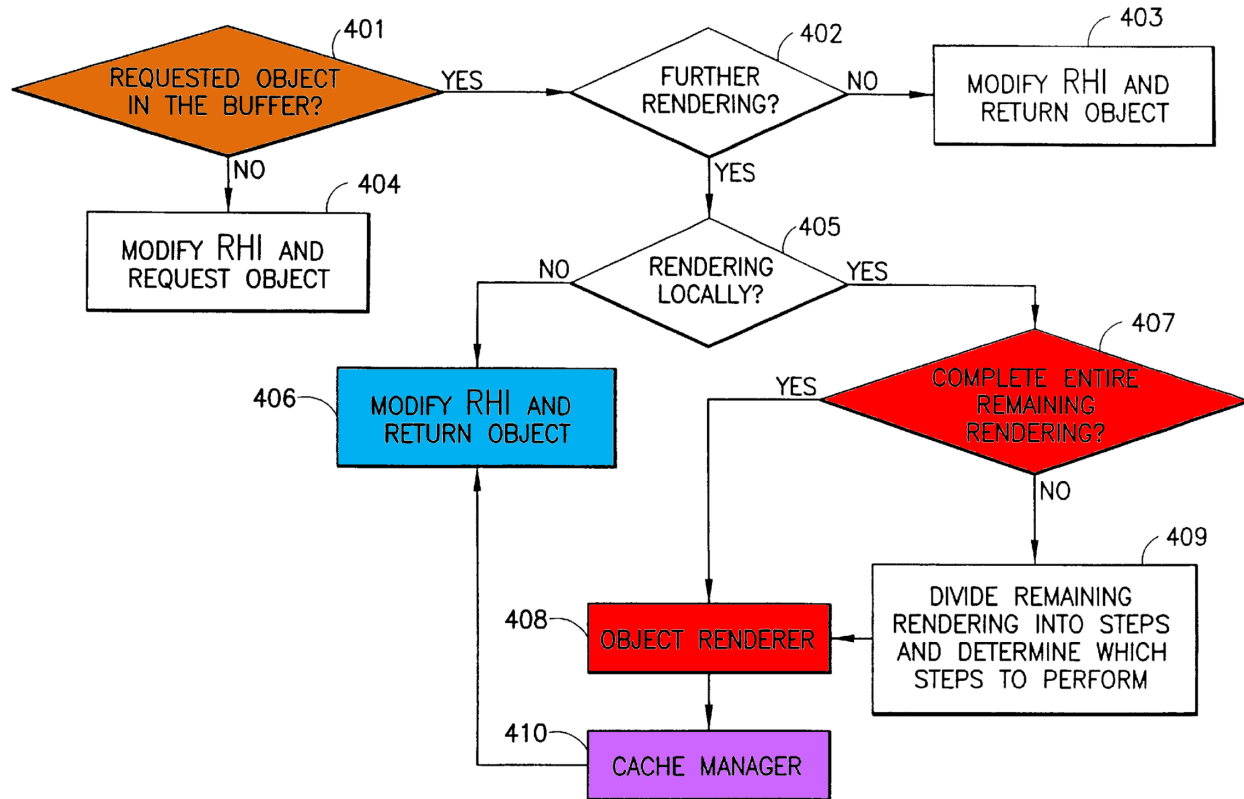


FIG.4

76. At **step 401**, the proxy server searches the cache to determine whether there is already a stored version of the requested content that is sufficiently “detailed” to handle the request. Huang, 7:23-36. A version is “less detailed” if the version “does not satisfy the requirement” for the requested item, while a “more detailed version” may be “further rendered” to “satisfy the request.” Huang, 7:30-36. If further rendering (including, for example, transcoding) of the requested object

is necessary, then at **steps 407-408** the proxy server completes the remaining rendering steps locally. Huang, 7:53-8:4. The system then caches a copy of the requested object (**step 410**), and returns the requested content to the user (**step 406**). Huang, 8:4-8:11.

77. Huang further teaches how to track different versions of content in order to allow the system to retrieve partially-transcoded versions of content in order to further transcode such content. Huang teaches that each proxy server maintains “Receiver Hint Information” (RHI). Huang, 3:50-67. The RHI information provides “meta-information describing the capabilities of the object requesting node” as well as information “associated with a particular object.” Huang, 3:50-67, 6:56-58. For example, the RHI provides information that allows the server (and other servers in communication with the server) to determine “what portion ... of the remaining required rendering is to be performed by” the server, as well as to determine how to divide “a remaining rendering operation into steps” to be performed by a server. Huang, 4:24-37. For example, a server performs “complete or partial rendering based on the associated RHI” by “partition[ing]” the rendering process into “two or more steps” and modifying the RHI to “reflect the processing that [the server] performed.” Huang, 6:9-23. When a new request for previously-cached content is received, the server uses a cached object’s RHI to determine if “further rendering is necessary” and whether to “complete the entire rendering process” locally. Huang,

7:43-64. The steps for rendering can occur in different orders. For example, Huang teaches ordering steps to perform the “steps with the most bandwidth reduction” first, or to perform the steps in an order designed to “reduce load unbalancing.” Huang, 4:33-37.

78. A person having ordinary skill in the art would have been motivated to modify Tso with relevant teachings from Huang. Tso and Huang are analogous prior art and are in the same field as the '745 patent which, as I discussed above, with respect to the level of ordinary skill in the art, is networked media delivery. *See* '745 patent, 1:23-26 (“[T]he invention relates to an Internet server-based software system that provides delivery of automated graphics and other media to Web sites for access by an end user or consumer”). Both Tso and Huang are also reasonably pertinent to the alleged problems identified in the '745 patent, namely, automating media delivery and reducing the required overhead and time required to generate Web pages. '745 patent, 1:23-26 (“an Internet server-based software system that provides delivery of automated graphics and other media”), 4:52-58, 5:1-10, 6:6-10; Tso, 1:10-31 (“a system for dynamically transcoding data transmitted between two computers over a communications link”), 1:59-2:5, 4:1-5; Huang, 1:20-26 (“proxy servers perform distributed object rendering so that object contents can be displayed on or consumed by various kinds of client devices”), 2:1-10, 2:37-44.

79. A person having ordinary skill in the art would have understood Tso to not only render obvious, but actually disclose, retrieving partially-transcoded content from cache, including as part of a staged transcoding as I discussed in Section IX.A.1, and Huang provides explicit teachings of retrieving a partially transcoded version of a content item from cache, and then further processing that partially transcoded version of content to generate the requested version of content. *E.g.* Tso, 5:39-41 (disclosing retrieving stored information about cached objects such as “current compression progress for scaled data access and staged refinements”), 14:64-15:6; Huang, 6:63-67 (“Proxy server logic 203 may also include a cache manager 207 which maintains a local copy of the partially rendered or completely rendered objects in order to avoid repeating some object rendering operations with the same proxy server.”), 6:56-58 (“according to the RHI associated with a particular object”). Huang also explicitly discloses tracking the different versions of the generated and cached content using the RHI information that is included with the content. Huang, 6:9-23. Huang further states that retrieving such content advantageously avoids unnecessary re-rendering of content. Huang, 7:23-57. A person having ordinary skill in the art thus would have been motivated to modify Tso’s teachings of retrieving a specific version of content and performing “staged” transcoding by implementing these detailed teachings from Huang of retrieving and tracking partially-transcoded content items from a cache to achieve

the beneficial and predictable result of tracking the transcoding status of content on a server and reducing the processing involved in re-processing previously-processed data for at least the following independent reasons.

80. **First**, Tso’s and Huang’s teachings both include mechanisms for improving processing efficiency that are complementary with one another. As I described in greater detail in the prior section, Tso provides multiple teachings of using partially-transcoded content in order to transcode content in response to client requests. For example, Tso teaches that “both original and transcoded versions of content” are cached for transmission to a client in order to avoid “the need to re-retrieve the content from Internet 18 or to re-transcode the content,” teaching that servers use previously-transcoded versions of content to satisfy later client requests and avoid re-performing transcoding that was previously performed. Tso, 4:1-5. Tso also teaches that a client requests a partially-transcoded version of content on which “additional transcoding” is required—teaching that transcode service providers will request partially-transcoded content to satisfy a content request. Tso, 14:57-15:6. Tso also teaches that, when content is retrieved from cache, a “stage” property is used so that the requestor can determine how much the content has already been transcoded, and using this information during the process of determining how to transcode content. Tso, 5:35-42, 6:9-23. Tso thus teaches that a proxy server completes transcoding of partially-transcoded content from another

proxy server, and that proxy servers send, receive, and further transcode partially-transcoded content, for example, as part of a “staged” or “collaborative” transcoding process. *E.g.*, Tso, 4:39-61, 5:35-42, 14:57-15:6, 15:66-16:14. For example, Tso teaches that generated content may need to be converted to greyscale for certain users; when implementing this teaching, partially or fully-transcoded content may be further transcoded “to greyscale for users lacking a color display.” Tso, 17:9-15. These teachings not only render obvious, but actually disclose, retrieving and further transcoding partially-transcoded content from cache. Huang extends these teachings from Tso by providing an explicit teaching of performing such retrieval and further rendering of content that was locally cached. Huang, 6:63-67, 7:42-8:11. Further, Tso teaches that it is advantageous to eliminate “the need to re-retrieve” or “re-transcode” “content.” Tso, 4:1-5. Complementing this disclosure, Huang teaches that by utilizing previously-cached versions of content, including such versions that were “partially rendered,” a server avoids “repeating some object rendering operations.” Huang, 6:63-67, 2:9-13. Thus, whether retrieving a fully transcoded file from the Tso cache and treating as an intermediate version of requested content for further transcoding as suggested by Huang, or retrieving a partially transcoded version (for example, as discussed at Tso, 14:57-15:6) for further transcoding, both paths would provide an obvious improvement to Tso in processing speed and efficiency as taught by Huang. A person having ordinary skill in the art would have

been motivated to modify Tso with this teaching to advantageously allow a server with its own locally-cached and partially-rendered or transcoded copy of content to also perform the rendering (which, as previously explained, encompasses transcoding) process locally on that server. Huang, 7:23-50, 2:8-13; Tso, 4:1-5.

81. **Second**, Huang provides additional detailed teachings of how to track transcoding status of content that allow Huang to effectuate its teachings of using partially-rendered content to respond to user requests for content objects in a manner that complements Tso's teachings of how to retrieve specific versions of content. Tso broadly teaches using the "GetProperties()" and "SetProperties()" calls to track transcoding status by retrieving and setting the properties of content, and teaches that this includes tracking properties describing the status of a "staged" transcoding process. *E.g.*, Tso, 5:36-42, 14:30-36. However, Tso provides only broad teachings of how such information about the transcoding status of content is tracked. Huang provides additional detailed teachings of tracking the rendering (including transcoding) status by using "RHI" information that includes and "object-specific descriptor information" to track transcoding status. Huang, 5:60-6:23, 6:56-58. Huang teaches that when a particular version of content is requested and not available, the server can use a "more detailed version" in cache instead of re-retrieving the original object to "avoid repeating some object rendering operations." Huang, 7:23-42, 2:8-13, 6:63-67. The server uses the "RHI" value to track the

content available on the server, as well as RHI information associated with a particular content object, including the different transcoding statuses of cached versions of the content. *Id.*; Huang, 6:9-23, 6:56-58, 7:43-64. For example, Huang teaches “stag[ing]” a rendering process by breaking the process into steps, determining which steps to be performed first, and tracking the progress of performing the transcoding steps using RHI information. Huang, 3:29-49, 4:24-52, 6:9-23, 6:56-58, 8:50-9:3. A person having ordinary skill in the art therefore would have found it obvious to modify Tso with Huang’s teachings by using Tso’s GetScaledObject() and GetProperties() calls together with Huang’s teachings of RHI information to find and retrieve content with properties specifying a partially-rendered, “more detailed version” to be further refined into the requested content item, *e.g.*, as part of staged transcoding, and using Tso’s SetProperties() calls to update RHI values, which allows the system to use the RHI to track the transcoding profiles of different content items in a manner that allows for the location and retrieval of partially-transcoded content, thereby advantageously reducing unnecessary repeated retrieval and rendering of content.

82. Applying these RHI teachings to Tso advantageously allows splitting the transcoding and rendering processes performed on servers into separate “steps,” after each step is performed, caching “partially” transcoded or rendered versions, and tracking the transcoding or rendering status of each cached version. Huang,

4:24-39, 6:9-23, 6:63-67. As a result, the server is also capable of configuring processing to arrange the rendering process into a specified order of steps as taught by Huang, in order to for example maximize “bandwidth reduction” or reduce “load unbalancing” during the process of transcoding, without repeating any such rendering or transcoding steps. *Id.* Because of Huang’s teachings of minimizing “bandwidth,” a person having ordinary skill in the art would have found it obvious to first perform transcoding to reduce the size of content because performing this step first will streamline all remaining steps by reducing the amount of data to be processed. Indeed, performing such scaling as the first step during rendering or transcoding was well-understood in the art to be advantageous. For example, Luca Delgrossi et al., *Media Scaling in a Multimedia Communication System*, 2 MULTIMEDIA SYS. 172 (1994), a copy of which I understand has been designated as Exhibit 1036, teaches that because “scaling is intended to reflect bandwidth constraints in the underlying resources, it is useful to scale a data stream before it enters a system bottleneck; otherwise, it is likely to contribute to the overload of the bottleneck resource.” Delgrossi, 173. As another example, Armando Fox and Eric A. Brewer, *Reducing WWW latency and bandwidth requirements by real-time distillation*, COMPUT. NETWORKS & ISDN SYS., PROCS. 5TH INT’L WORLD WIDE WEB CONF. (May 1996), which I understand has been designated as Exhibit 1037, teaches that “[a] large color graphic can be scaled down to one-half or one-quarter

length along each dimension, reducing the total area and thereby reducing the size of the representation” before undergoing “[f]urther compression ... by reducing the color depth or colormap size.” Fox and Brewer, 1446. As a third example, Delp (Ex. 1008) teaches an image generation process where an image must “first be decompressed [and] scaled to the correct size” before it is “corrected to the original color, matched to the color of the display, converted to the resolution of the display, and rotated. Delp, 3:61-4:2.

83. In light of the above teachings, a person having ordinary skill in the art would have had a reasonable expectation of success when modifying Tso with Huang’s teachings of reusing partially-processed content. Indeed, Tso already teaches many of the features described in Huang, including partially processing content to generate requested content, and also teaches mechanisms that are able to take advantage of Huang’s teachings, including performing such processing at both servers and clients, and processing content using a “staged” process or using discrete “transcode service providers.” Tso, 5:36-42; 12:53-13:11. Further, such use of intermediate content items to minimize the amount of further processing that is required was well understood in the art and known be advantageous when serving content to clients. For example, U.S. Patent No. 5,420,967, which is assigned to IBM and names Delp as its first named inventor, a copy of which I understand has been designated as Exhibit 1008, similarly teaches that “intermediate images generated

for one requested image can be used by other requested images.” Delp, Abstract. Delp teaches a system in which “[e]ach intermediate image corresponds to a change of an image state between an existing image and a requested image,” including, for example, intermediate images in which “the scaling of the existing image is a change of image state and will result in the addition of the representation of an intermediate image to the image table.” *Id.* As a result of maintaining such intermediate images, the system can develop an “optimal image path” that minimizes resource consumption to generate a requested image. Delp, 2:6-42. As another example, U.S. Patent No. 5,969,716, which on its face states it was assigned to Interval Research Corporation, and names Davis as its first named inventor, a copy of which I understand has been designated as Exhibit 1009, teaches a system that “permits pre-existing media content to be reused in a meaningful way.” Davis, 4:2-4. The system uses “descriptive data that represent[s] the content of media signals” to allow the system to compose “new content representations” in audio and video contexts as well by “reus[ing]” preexisting content. Davis, 2:20-67. These examples confirm that it was well understood to reuse previously-transcoded content to generate additional content items, further demonstrating that a person having ordinary skill in the art would have had a reasonable expectation of success when modifying Tso with Huang’s teachings of caching intermediate content. A person having ordinary skill in the art would have understood that modifying Tso to include Huang’s disclosures

would require little more than updating the code implementing Tso to carry out the additional functionalities taught by Huang. Modifying Tso with Huang's teachings would have been a straightforward application of those teachings in a manner that advantageously improves the efficiency of the system.

3. Invalidity of Claims 1-5 Over Tso in view of Huang

(a) Element [1.pre]

84. **Element [1.pre] recites “A method in a host computer for developing transformation processing operations to optimize media content playback to a plurality of playback devices connected with the host computer in a network, the method comprising.”** This element is met by Tso.

85. **Tso discloses a method in a host computer (e.g., “server”) for developing transformation processing operations (e.g., using “selection criterion” to “provide a virtually limitless range of dynamic transcoding services”) to optimize media content playback (e.g., “compress and/or scale” media including “image, video, or HTML” to optimize or “reduce[] user-visible latencies”) to a plurality of playback devices (e.g., “multiple users (i.e., clients),” “network client computer[s]”) connected with the host computer in a network (e.g., “an ISP’s network,” “Internet 18”).** *See, e.g.,* Tso:

- Figure 3

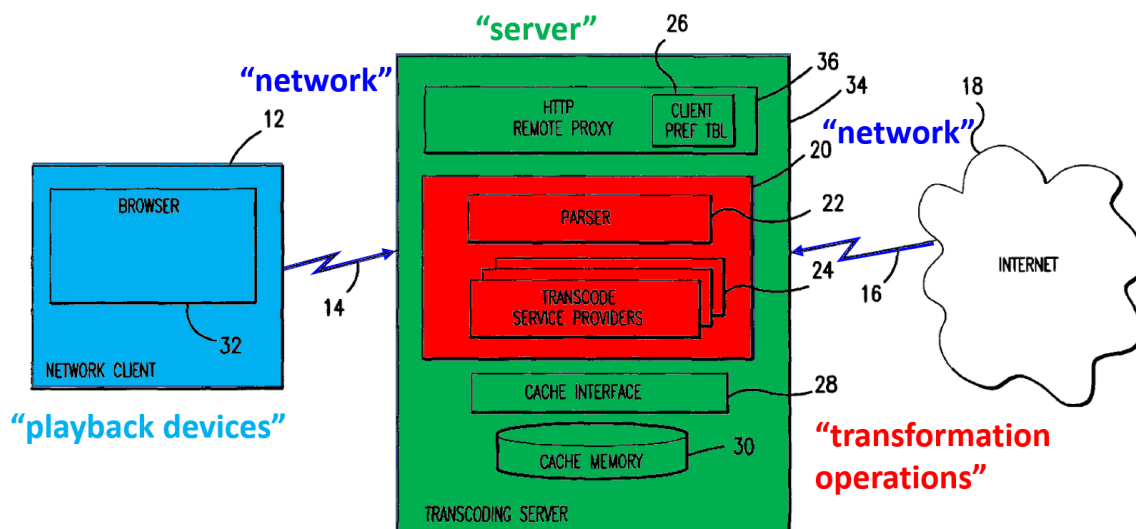


FIG.3

- 2:8-12 (“Embodiments of the present invention relate to *devices*, systems and methods for transcoding information transmitted between computers, such as a network server computer and a network client computer.”)
- 2:44-47 (“Embodiments of the present invention provide the ability to dynamically transcode information transmitted between, for example, a network server computer and a network client computer.”)
- 3:17-30 (“Transcoder 20 may be implemented, for example, as a software module installed in a network proxy, in a client device, in a network server device, or in a content server device. In one particular implementation, illustrated in FIG. 3, transcoder 20 is installed in a remote transcoding server 34 arranged between network client 12 and Internet 18. Transcoding server 34 may comprise, or be a part of, a network server, a stand-alone computer in communication 25 with a network server, or a distributed system of computers. *Remote transcoding server 34 may be coupled, for example, to an ISP's network, a corporate network, or anywhere on Internet 18, and may provide multiple users (i.e., clients) with a means to obtain content on Internet 18.*”)
- 3:45-60 (“[T]ranscoder 20 is coupled to HTTP remote proxy 36. Parser 22 manages the transcoding of data to be transmitted from transcoding server 34 to network client 12. To this end, parser 22 controls *transcode*

service providers 24 to selectively transcode content based on a predetermined selection criterion. For example, one or more transcode service providers 24 may provide the capability to compress and/or scale different types of data content, such as image, video, or HTML (HyperText Markup Language).”)

- 8:10-13 (“*Applying the above-listed selection criteria, or combinations thereof, embodiments of the present invention may be used to provide a virtually limitless range of dynamic transcoding services.*”)
- 16:63-17:15 (“The unique features of such embodiments *enhance the ability of these computers to* access *data* on the network in a timely fashion *with reduced user-visible latencies*, thereby enabling content authors to produce rich content without fear that only users with highly-sophisticated data communications and display capabilities are able to enjoy it.”)
- *See also* 2:50-3:6, 6:55-63, 7:15-8:9, 15:7-28, 16:63-17:15, 22:13-18, Fig. 5.

86. Tso discloses that “a network server computer” develops a process for media “transcoding” operations by using combinations of “selection criteria,” both individually or in combinations thereof, to provide a “virtually limitless range of dynamic transcoding services” for “image, video, or HTML” content. Tso, 2:8-12, 2:44-47, 3:45-60, 8:10-13, Fig. 3. Tso further discloses that this content is sent to “multiple users” across a network, such as the Internet, and is optimized to reduce “user-visible latencies.” Tso, 3:17-30, 16:63-17:15.

(b) Element [1.a]

87. Element [1.a] recites “receiving a first request from a first playback device for media content.” This element is met by Tso.

88. Tso discloses receiving a first request (e.g., “HTTP request”) from a first playback device (e.g., “user,” “network client”; see [1.pre]) for media content (e.g., “image” or “video” “data content”). See, e.g., Tso:

- **Figure 3**

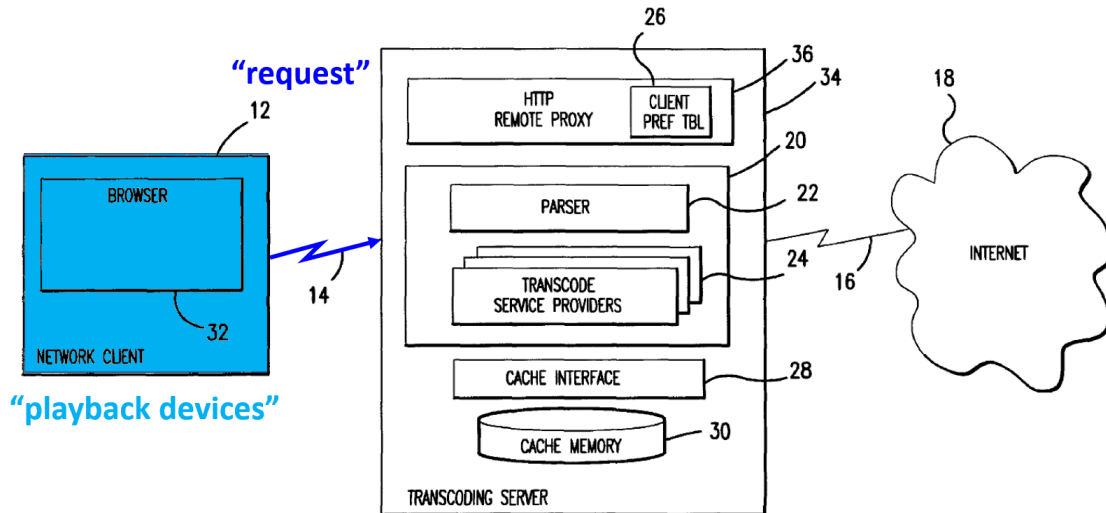


FIG.3

- 3:45-60 (“[T]ranscoder 20 is coupled to HTTP remote proxy 36. Parser 22 manages the transcoding of data to be transmitted from transcoding server 34 to network client 12. To this end, parser 22 controls *transcode service providers 24 to selectively transcode content based on a predetermined selection criterion*. For example, one or more *transcode service providers 24 may provide the capability to compress and/or scale different types of data content, such as image, video, or HTML (HyperText Markup Language)*.”)
- 9:49-65 (“*Network client 12, via browser 32, transmits an HTTP request for the hypertext object to transcoding server 34 over client/server communications link 14.*”)
- See also 3:7-16, 3:39-60, Fig. 5.

89. See my discussion of element [1.pre] for further discussion on Tso's disclosures of this element. Tso also discloses that a "transcoding server" receives, from a "network client," an "HTTP request" for image or video data.

(c) Element [1.b]

90. **Element [1.b] recites "wherein the first request contains information, the information indicating a first original media content, first content generation operations, and first transformation operations."** This element is met by Tso.

91. **Tso discloses wherein the first request contains information (e.g., request "packet" contains "selection criteri[a]"), the information indicating a first original media content (e.g., indicating "particular content," "requested hypertext object," which can be saved as the "original version"), first content generation operations (e.g., indicating "content characteristics" specifying "data type, type of encoding/compression" or "size"), and first transformation operations (e.g., indicating "network client" properties, "content characteristics" specifying a "resolution," or "content provider preferences" specifying how to "customize content").** See, e.g., Tso:

- 6:64-7:7 ("As noted above, *parser 22 may selectively invoke one of transcode service providers 24 based upon satisfaction of a predetermined selection criterion. Such selection criterion may comprise, for example, information contained in a header portion* of a data packet received by transcoding server 34, such as a MIME type, a URL (Uniform Resource Locator), a last modified time indicator and so

on. *Alternatively, the predetermined selection criterion may comprise information contained in a data portion of such a data packet, such as particular content, key words, structures (for example, heading levels), and so on.*”)

- 7:15-8:9 (“The following discussion provides still more *examples of the types of information which may be used to dictate which of transcode service providers 24 are invoked*. It should be noted, however, that these examples are provided by way of illustration only, and are not intended to limit in any way the scope of the invention claimed herein. The predetermined *selection criterion* may comprise: (1) *network client 12, such as a display dimension, resolution, number of colors, ...* (3) *content characteristics, such as its data type, type of encoding/compression, size, and dimension; ...* (8) *content provider preferences, including the degree of alteration desired for its content, ... rules or programs to run for customizing content* (for example, news or advertising, custom language translation software) *based on user or client characteristics....*”)
- 14:37-46 (“*If parser 22 determines that the requested hypertext object does not exist* in the server-side cache memory 30, HTTP remote proxy 36 issues an HTTP request to retrieve the hypertext object from Internet 18 (Step 190). ... [I]f the object is found, HTTP remote proxy 36 passes the handle for the incoming data stream to parser 22, which in turn initiates *caching of an original version of the retrieved hypertext object* (Step 230).”)
- 17:9-15 (“Embodiments of the present invention may also be advantageously used for ... *converting color images to greyscale images for users lacking a color display....*”)
- *See also 9:26-41, 9:49-65* (“requests corresponding to ... hypertext objects that may be embedded in the web page”), *10:16-49* (“requested hypertext object”).

92. See my discussion of elements [1.pre] and [1.a] (§§ IX.A.3(a), IX.A.3(b)) for further discussion on Tso’s disclosures of this element. Tso also discloses that HTTP request packets contain “information” in either the “header” or

“data” “portions” of the packet, and that this “information” includes information specifying the “particular content” requested, as well as the “selection criterion” which specifies how to transcode original media content into the requested media content. Tso, 6:64-7:14. For example, these “specification criteria” include parameters specifying the content “data type, type of encoding/compression, size, and dimension,” as well as “the degree of alteration desired for” the content and “rules or programs to run for customizing content.” Tso, 7:15-8:9. Using the “selection criterion,” the server’s “parser” will “selectively invoke” one or more “transcode service providers” to perform the necessary transcoding. *Id.*

93. Tso discloses that “selection criteria” contained in the request specify “content characteristics” including, for example, “data type, type of encoding/compression,” and “size.” Tso, 7:15-8:9. A person having ordinary skill in the art would have understood that these types of characteristics specify content generation operations that are consistent with, for example, the “save” and “scale” content generation operations described in Table E of the ’745 patent, which allow saving media to a “specified file” and scaling media to a “specified size.” 745 patent, 21:20-22:38. For example, the “Scale” content generation operation “[s]cales the media to the specified size” just as the “size” content characteristic taught by Tso specifies the same parameter. *Id.*; Tso, 7:15-8:9.

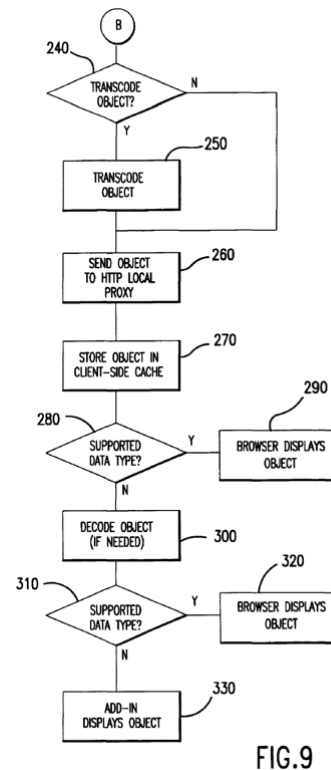
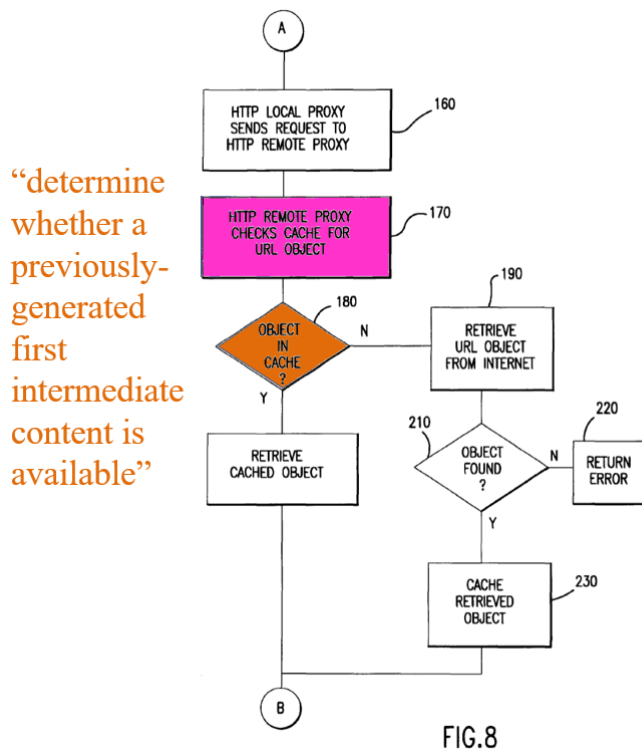
94. Tso also discloses “selection criteria” that further specify, for example, “client” characteristics such as a “number of colors,” additional “content characteristics” such as “encoding/compression” or “resolution,” and “content provider preferences” including, for example, “the degree of alteration desired for its content.” Tso, 7:20-8:4. A person having ordinary skill in the art would have understood these to disclose transformation operations that are consistent with the ’745’s “media processing script command” transformation operations, including, as an example, the “Set Resolution” operation. ’745 patent, 10:50-15:20. The “resolution” selection criterion, for example, specifies the same parameters for transcoding as the “Set Resolution” operation in the ’745 patent, while the “number of colors” selection criterion specifies the same parameters for transcoding as the “Colorize” operation. *See id.*; Tso, 7:60-8:4. Indeed, Tso teaches using the “number of colors” parameter to, for example, “convert[] color images to greyscale images for users lacking a color display.” Tso, 17:9-15.

(d) Element [1.c]

95. **Element [1.c] recites “determining whether a previously-generated first intermediate media content is available for reuse, the previously-generated first intermediate media content having been created using the first original media content and the first set of content generation operations.”** This element is met by the combined teachings of Tso and Huang.

96. Tso discloses determining whether a previously-generated media content is available for reuse (e.g., use “GetScaledObject()” to “request[] a particular version of that object, such as a high-quality rendition” from “cache” “determine whether a ... version of the requested hypertext object already exists” in “cache” and, if “not found,” “create ... a placeholder”), and media content created using the first original media and the first set of content generation operations (“transcoded version[] of content” created using original media (see [1.b] (§ IX.A.3(c))) and content generation operations (see [1.b] (§ IX.A.3(a))))). See, e.g., Tso:

- Figures 8 and 9



- 4:1-5 (“*Server-side cache memory 30 may be used to store both original and transcoded versions of content for later transmission to network*”)

client 12 *without the need to re-retrieve the content from Internet 18 or to re-transcode the content.*")

- 4:23-29 ("The WOSA architecture ... enables *additional transcode service providers 24 to be dynamically added to the system* to provide new features and/or better transcoding algorithms, while at the same time *not requiring changing or retesting other software components in the system.*")
- 5:36-41 ("The GetProperties() and SetProperties() calls retrieve and store *information about cached objects*, including information maintained by transcode service provider 24 *used to determine transcoding properties and transcoding status of a cached object. Transcode service provider 24 may use such information*, for example, *to determine current compression progress for scaled data access and staged refinements.*")
- 6:9-28 ("The *GetScaledObject()* call is similar to GetObject(), and is also used to request an object from server-side cache memory 30; however, it *adds support for requesting a particular version of that object, such as a high-quality rendition. ... [W]hen network client 12 requests a hypertext object, HTTP remote proxy 36 uses either the GetObject() or GetScaledObject() call* (depending on if network client 12 is capable of receiving scaled/transcoded datatypes) *to retrieve the hypertext object* from parser 22. *If the hypertext object is not found*, parser 22 uses the CreateEntry() call to *create* an entry (in effect, *a placeholder*) in server-side cache memory 30 for the new object.")
- 14:47-15:6 ("Upon receipt, HTTP local proxy 48 initiates caching of the transcoded hypertext object (Step 270). In addition, *client-side parser 50 determines whether any further processing is required* before the hypertext object is rendered *If additional transcoding is required, client-side parser 50 passes the handle to an appropriate transcode service provider 52* (Step 300).")
- 15:66-16:14 ("[A]n embodiment of the present invention may use *a special proxy-to-proxy protocol that extends the existing request/response structures to indicate whether and what sort of transcoding has already been performed on the content*. Such a specialized protocol ... *enables multiple proxies to work collaboratively*, yet still transparently to users, client software, existing "standard" proxies and content servers.")

- See also 4:38-47, 4:62-5:7, 5:36-42, 13:39-44, 14:21-15:6.

97. **Huang discloses determining whether a previously-generated first intermediate media content is available for reuse** (e.g., determining that “a copy of the requested object can be found in the local cache” for which “further rendering is necessary”), **the previously-generated first intermediate media content having been created using the first original media content and the first set of operations** (e.g., the copy found in local cache is “partially rendered” using “only one” of “two or more steps” tracked with “RHI” information). See .e.g., Huang:

- 4:24-39 (“A selection method is provided for *each intermediate node to decide dynamically and independently of other nodes what portion of the required rendering to perform locally using the RHI information*. The selection method can include steps of (a) *dividing a remaining rendering operation into steps*; (b) *selecting one or more rendering steps to be performed locally* in order to optimize a given objective function, using the RHI information as an input parameter; and (c) *performing the one or more rendering steps selected for the current node*. The objective function can be to *perform the rendering steps with the most bandwidth reduction first, and/or to perform the rendering steps so as to reduce load unbalancing* among the remaining nodes on the path to the requesting client device node.”)
- 6:9-23 (“When a requested object passes through the proxy network, *any proxy server 110, 111, 112 can perform a complete or partial rendering based on the associated RHI*. For example, *if the entire rendering process can be partitioned into two or more steps, a given one of the proxy servers (e.g., 110) may decide to perform only one of the rendering steps*, and to then forward the partially rendered object to another proxy server (e.g., 111). *The proxy server 110 also modifies the RHI to reflect the processing that it performed*, and forwards the modified RHI as well to the proxy server 111.”)

- **6:56-58** (“An object renderer 206, which performs object rendering according to the *RHI associated with a particular object*, may also be included in the proxy server logic 203.”)
- 6:63-67 (“Proxy server logic 203 may also include a *cache manager 207* which *maintains a local copy of the partially rendered* or completely rendered *objects in order to avoid repeating some object rendering operations with the same proxy server.*”)
- 7:42-8:11 (“*If a copy of the requested object can be found in the local cache, at step 402, the proxy server checks the cached object against the RHI to see if any further rendering is necessary. ... By checking the RHI, the proxy server 110, 111, 112 can determine if any further rendering is necessary.* If no further rendering is necessary, the proxy server modifies the RHI to indicate its local condition for providing rendering services and returns the object at step 403. *If further rendering is found to be necessary*, based on the RHI or the requesting device, then *the proxy server* executes at step 405 a selection function to *determine whether* or not it wishes *to perform the rendering locally.* ... If the proxy server instead decides to perform the rendering locally, it checks the RHI at step 407 to *determine if it wishes to complete the entire rendering process*, or just some part of the required rendering process. ... *After* the rendering process is complete (*either a complete or partial rendering*), the cache manager 207 is called at step 410 to determine whether or not to *cache a copy of the rendered object locally.* *The proxy server* 110, 111, 112 *then* modifies the RHI at step 406 to reflect its local condition and *returns the* rendered (*completely rendered* or partially rendered) *object* along with the modified RHI.”)
- *See also* 3:32-47, 8:50-59.

98. See my discussion of elements [1.pre] and [1.b] (§§ IX.A.3(a) and IX.A.3(c)) for further discussion on Tso’s disclosures of this element. Further, Tso also discloses using a “GetScaledObject()” call to retrieve a “particular version of [a media] object,” for example, a version of the media object for which the claimed content generation operations were already applied, from a local cache to avoid the

“need to... re-transcode the content.” Tso, 6:9-28, 4:1-5. The system determines whether the item is in a cache because, if it is “not found,” the system will create “a placeholder” for the item so that it will be stored in cache. Tso, 6:9-28. This “placeholder” is thus generated as the outcome of the response to the determination whether a previously generated media content is available for reuse that occurs when the result of the determination is that the content is not available. *Id.*

99. Tso further discloses that proxy servers create and receives partially-transcoded content, and cache transcoded versions of content, for example, in the context of servers “collaboratively” working to transcode content, and by sending partially-transcoded content to clients, as I discussed in greater detail above in Section IX.A.1. Tso, 4:1-5, 5:36-41, 14:47-15:6, 15:66-16:14; §IX.A.1. As a result, because the server caches received and transcoded content, a person having ordinary skill in the art would have understood that a server responsible for performing only a partial transcoding caches that partially-transcoded version of content. *Id.*

100. Tso not only least renders obvious, but actually discloses performing content generation steps first. Tso is agnostic to the order in which transcoding steps are performed, and indeed permits transcoding steps to be performed in any order: for example, when a server adds new transcoding “features” using new “transcode service providers,” the new processes are implemented without changing the operation of existing providers. Tso, 4:23-29. Because this is true regardless of the

type of transcoding service provided by the service provider, this teaches that any transcoding service, including services that are described in the '745 patent as “content generation” and “transformation” procedures, may be performed first. Because providers can perform transcoding in any order, Tso not only renders obvious but actually discloses first converting content to a particular “encoding/compression” or “size”—in other words, according to a claimed content generation operation—before performing additional transcoding operations.

101. In addition to Tso’s teachings of servers sending, receiving, and caching partially transcoded content, Huang further teaches that a server determines if a partially-rendered version of a content item that is of “higher quality”—for example, an intermediate version of the content. Huang, 4:24-39, 6:9-23, 6:63-67, 7:42-8:11; §IX.A.2. If such a content item is found, the server will then perform “further rendering” the “partially rendered” content, for example by performing the remainder of the transcoding on a version of content for which “only one” of “two or more steps” of rendering operations was performed. Huang, 4:24-39, 6:9-23, 6:63-67, 7:42-8:11; §IX.A.2. Huang further discloses that the rendering status for content on a server is tracked using “RHI” information that specifies information about particular content objects. Huang, 6:9-23, 6:56-58, 7:42-8:11. The server selects an order to perform rendering steps based on an order such as load balancing or bandwidth reduction—which as I explained in Section IX.A.2 a person having

ordinary skill in the art would have understood to describe, for example, processing the content to reduce its size first before performing other content operations. Huang, 4:24-39.

102. As I discussed above in Section IX.A.2, a person having ordinary skill in the art would have been motivated to modify Tso with Huang's teachings to advantageously avoid re-retrieval and re-transcoding of transcoding steps already performed on partially-transcoded content, and to perform transcoding steps in an order that was known and well-understood to improve the efficiency of the system, thereby improving the efficiency of content generation. Tso, 14:47-15:6, 15:66-16:14; Huang, 6:63-67, 7:42-8:11. Applying Huang's teachings, a person having ordinary skill in the art would have found it obvious for a server to use Tso's `GetScaledObject()` and `GetProperties()` requests to retrieve and check the transcoding status of a cached copy of content, including determining the transcoding steps that have already been performed on the content before then performing additional transcoding operations on the content, rather than re-retrieving and re-transcoding the content from scratch. *Id.*

103. A person having ordinary skill in the art also would have found it obvious to apply Huang's teaching of performing "bandwidth reduction" transcoding steps during the transcoding process by first performing Tso's "size"-based transcoding operation, which the '745 patent describes as a content generation

operation, prior to performing other types of transcoding—which as I previously discussed in Section IX.A.2 would have improved the “bandwidth” consumption of the transcoded content. Huang, 4:24-39. Content transcoded for size is then stored as “partially-rendered” content in cache and is used to “avoid repeating some object rendering operations.” Huang, 6:63-67. As a result, subsequent searches for partially-transcoded content would locate such partially-transcoded and cached versions of content on which content generation operations, but not transformation operations, had been performed, rendering obvious this element. *Id.*

(e) Element [1.d]

104. **Element [1.d] recites “responsive to determining that a previously-generated first intermediate media content is available, creating a first optimized media content for the first playback device by performing the first set of transformation operations on the previously-generated first intermediate media content.”** This element is met by the combined teachings of Tso and Huang.

105. **Tso discloses performing the first set of transformation operations.** See my discussion of “transformation operations” above for element [1.b] in Section IX.A.3(c) above, as well as my discussion of the “determination” referenced in this element. Additionally, see my discussion of elements [1.pre] and [1.c] (§§ IX.A.3(a), IX.A.3(d)) for further discussion of Tso’s disclosures of this element.

106. **Huang discloses responsive to determining that a previously-generated first intermediate media content is available** (e.g., “if a copy of the requested object can be found in the local cache” for which “further rendering is necessary”), **creating a first optimized media content for the first playback device** (e.g., creating a “completely rendered ... object”) **by performing the first set of operations on the previously-generated first intermediate media content** (e.g., “proxy server” “decides to perform the rendering locally” and “complete the entire rendering process”). *See, e.g., Huang:*

- 7:42-8:11 (“***If a copy of the requested object can be found*** in the local cache, at step 402, the proxy server checks the cached object against the RHI to ***see if any further rendering is necessary. ... By checking the RHI, the proxy server 110, 111, 112 can determine if any further rendering is necessary.*** If no further rendering is necessary, the proxy server modifies the RHI to indicate its local condition for providing rendering services and returns the object at step 403. ***If further rendering is found to be necessary,*** based on the RHI or the requesting device, then ***the proxy server*** executes at step 405 a selection function to ***determine whether*** or not it wishes ***to perform the rendering locally.*** ... If the proxy server instead decides to perform the rendering locally, it checks the RHI at step 407 to ***determine if it wishes to complete the entire rendering process,*** or just some part of the required rendering process. ... ***After*** the rendering process is complete (***either a complete or partial rendering***), the cache manager 207 is called at step 410 to determine whether or not to ***cache a copy of the rendered object locally.*** ***The proxy server 110, 111, 112 then*** modifies the RHI at step 406 to reflect its local condition and ***returns the*** rendered (***completely rendered*** or partially rendered) ***object*** along with the modified RHI.”)
- ***See also Abstract, 3:17-48, 4:2-15, 5:44-55, 6:63-67, 10:46-56.***

107. See my discussion of elements [1.b] and [1.c] (§§ IX.A.3(c)-(d)) for further discussion on Huang’s disclosures of this element. As I discussed with respect to element [1.c], Tso in view of Huang renders obvious the recited determination; Huang further discloses that if a “partially rendered” object is found in cache, the node creates an optimized media content by “complet[ing] the entire rendering process” based on “RHI” information stored with the content. Huang, 7:42-8:11, 6:9-23; *see* [1.c] (§IX.A.3(d)). Applying this teaching to Tso’s teaching of performing transformation operations (*see* my discussion of element [1.b] (§ IX.A.3(c))) renders obvious this element. And as I discussed above in Section IX.A.2, a person having ordinary skill in the art would have been motivated to modify Tso with Huang’s teachings to advantageously track content in a manner that avoids re-retrieving and re-transcoding of content by reusing cached and partially-transcoded versions of content.

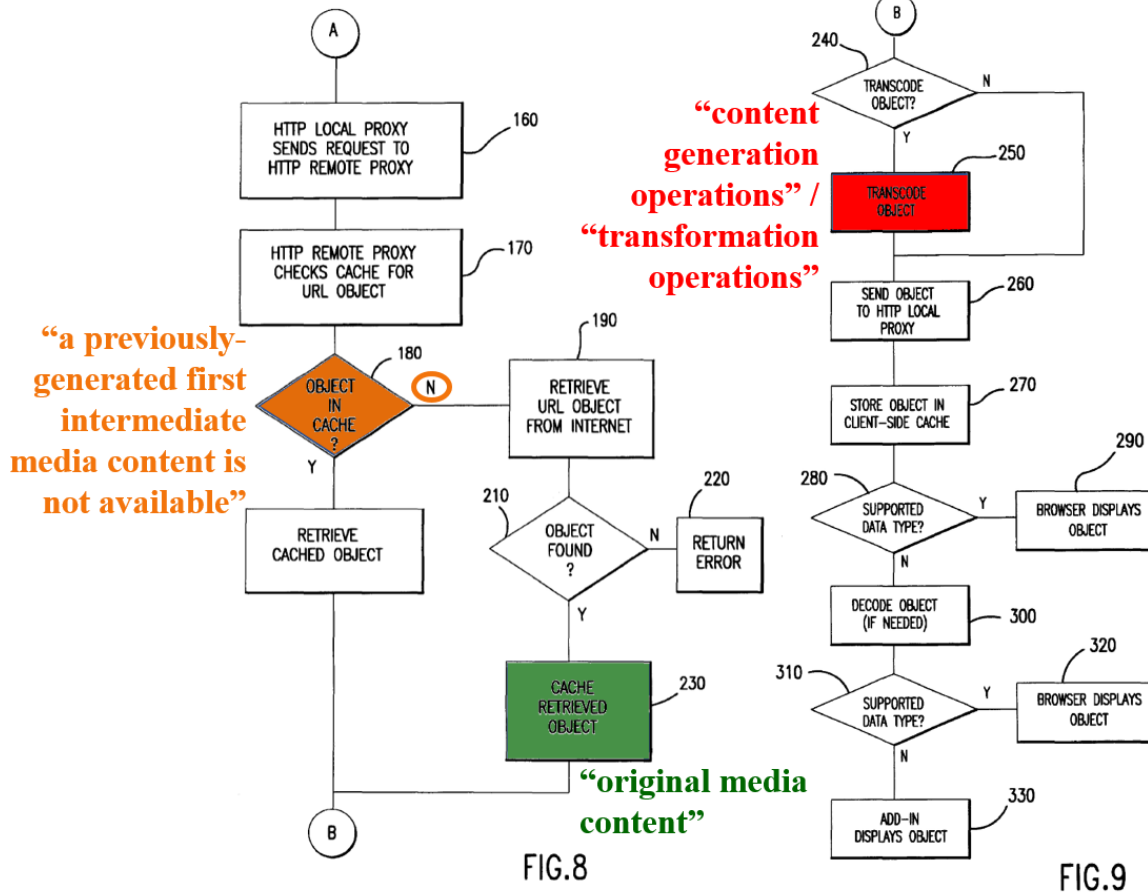
(f) Element [1.e]

108. **Element [1.e] recites “responsive to determining that a previously-generated first intermediate media content is not available, creating a first optimized media content for the first playback device by creating a first intermediate content using the first original media content and the first set of content generation operations, and performing the first set of transformation**

operations on the first intermediate media content.” This element is met by the combined teachings of Tso and Huang.

109. **Tso discloses responsive to determining that a previously-generated first media content is not available** (*e.g.*, responsive to “determin[ing]” that “a copy of the required version” does not “resid[e] in server-side cache memory”), **creating a first optimized media content for the first playback device** (*e.g.*, “retriev[ing] the hyper text object” and “transcode[] the data stream appropriately” for “network client 12”), **by using the first original media content and the first set of content generation operations, and performing the first set of transformation operations** (*see* [1.b]-[1.d] (§§ IX.A.3(c)-(e))). *See, e.g.*, Tso:

- Figures 8-9



- 10:24-49 (“Upon being invoked, *parser 22* first calls cache interface 28 with the requested hypertext object to *determine whether a copy of the required version already resides in server-side cache memory 30*. For purposes of illustration, assume no entry exists in server-side cache memory 30 for the requested hypertext object. *HTTP remote proxy 36 then invokes a call to retrieve the hypertext object from Internet 18* over server/network communications link 16. ... *Transcode service provider 24 then transcodes the data stream* appropriately, *and HTTP remote proxy 26 transmits the transcoded data stream to network client 12.*”)

- See also 3:43-60, 4:62-66, 5:36-42, 6:9-50, 6:64-66, 14:21-63, 14:37-63.

110. Huang further discloses the first intermediate media content (see [1.c] (§IX.A.3(d))) and creating a first optimized media content for the first playback device (e.g., creating a “completely rendered ... object”) by creating a

first intermediate content using the first original media content (e.g., creating and maintaining a “partially rendered ... object” in “cache” “based on objects retrieved from the content servers”) **and the first set of operations** (e.g., operations required for “partially rendered ... object”), **and performing additional operations on the first intermediate media content** (performing the “further rendering” required to generate a “completely rendered” object). *See, e.g., Huang:*

- 5:42-44 (“***Object renderings are performed*** by the proxies 110, 111, 112 ***based on objects retrieved from the content servers*** 120, 121.”)
- 6:63-67 (“Proxy server logic 203 may also include a ***cache manager*** 207 which ***maintains a local copy of the partially rendered or completely rendered objects in order to avoid repeating some object rendering operations*** with the same proxy server.”)
- 7:36-41 (“***If the requested object cannot be found*** in the cache, at step 404, ***the proxy server*** 110, 111, 112 modifies the associated RHI to indicate its ability for providing rendering services and then ***sends the request*** and the modified RHI to another proxy server or ***to the content server*** 120, 121, depending on the position of the proxy server in the entire proxy chain.”)
- 7:42-8:11 (“***If a copy of the requested object can be found*** in the local cache, at step 402, the proxy server checks the cached object against the RHI to ***see if any further rendering is necessary***. ... ***By checking the RHI, the proxy server 110, 111, 112 can determine if any further rendering is necessary***. If no further rendering is necessary, the proxy server modifies the RHI to indicate its local condition for providing rendering services and returns the object at step 403. ***If further rendering is found to be necessary***, based on the RHI or the requesting device, then ***the proxy server*** executes at step 405 a selection function to ***determine whether*** or not it wishes ***to perform the rendering locally***. ... If the proxy server instead decides to perform the rendering locally, it checks the RHI at step 407 to ***determine if it wishes to complete the entire rendering process***, or just some part of the required rendering process. ...

After the rendering process is complete (*either a complete or partial rendering*), the cache manager 207 is called at step 410 to determine whether or not to *cache a copy of the rendered object locally*. *The proxy server* 110, 111, 112 *then* modifies the RHI at step 406 to reflect its local condition and *returns the* rendered (*completely rendered* or partially rendered) *object* along with the modified RHI.”)

- See also Abstract, 2:32-44, 3:32-37.

111. See my discussion of elements [1.pre] and [1.b]-[1.d] (§§ IX.A.3(a), IX.A.3(c)-(e)) for further discussion on Tso’s disclosures of this element. Tso also discloses that if the system determines that the required version of the media content is not in cache, then the original content will be retrieved from the Internet and transcoded for the client using selection criteria, which specify the claimed content generation and transformation operations for that content. This is done by invoking the transcode service providers selected using the “selection criterion.” I discuss this process specifically in my analysis of elements [1.b], [1.c], and [1.d]. See §IX.A.3(c)-(e).

112. In addition to Tso’s teaching of transcoding content that has not been cached by using multiple transcode service providers that were selected using selection criterion, Huang further teaches generating optimized media content by using objects retrieved from a content server by “stag[ing],” rendering, and storing “partially rendered” content items. Huang, 6:63-67, 5:42-44, 7:36-41.

113. As I discussed in Section IX.A.3, a person having ordinary skill in the art would have been motivated to modify Tso with this teaching from Huang in order

to advantageously cache the “partial” transcoding output from each of the transcode service providers in order to advantageously reduce processing and networking time incurred from re-retrieval and re-processing. For example, in addition to transcoding the final version of content, modifying Tso with Huang’s teachings, the partially-transcoded content items that are output from the transcode service providers before transcoding is completed using the final transcode service provider are intermediate content items on which some but not all transcoding operations have been performed. This intermediate content is cached such that those intermediate content items are re-retrieved in response to further requests for content items that require the same transcoding operations, such as content generation operations, to be performed. Huang, 6:63-67. (“Proxy server logic 203 may also include a cache manager 207 which maintains a local copy of the partially rendered or completely rendered objects in order to avoid repeating some object rendering operations with the same proxy server.”). As a result, this cached intermediate media content output from service providers is retrieved from cache, instead of repeating the transcoding steps already performed by the transcode service provider. *Id.*

(g) Element [1.f]

114. **Element [1.f]** recites “**sending the first optimized media content to the first playback device.**” This element is met by Tso.

115. Tso discloses sending the first optimized media content to the first playback device (e.g., “transmit[]” the “transcoded data stream” to “network client[] 12”). See, e.g., Tso:

- 6:51-63 (“*Multiple-thread processing improves the efficiency of the present embodiment by not waiting for a hypertext object to be received in its entirety by HTTP remote proxy 36, or added in its entirety to server-side cache memory 30, before beginning to send the object to network client 12. Another benefit of multiple-thread processing is that parser 22 may efficiently process requests for the same hypertext object from multiple network clients 12. The hypertext object need only be retrieved from Internet 18 once, and appropriate versions may be transmitted to such multiple network clients 12 concurrently.*”)
- 10:24-49 (“Upon being invoked, *parser 22* first calls cache interface 28 with the requested hypertext object to *determine whether a copy of the required version already resides in server-side cache memory 30.* For purposes of illustration, assume no entry exists in server-side cache memory 30 for the requested hypertext object. *HTTP remote proxy 36 then invokes a call to retrieve the hypertext object from Internet 18* over server/network communications link 16. ... *Transcode service provider 24 then transcodes the data stream appropriately, and HTTP remote proxy 26 transmits the transcoded data stream to network client 12.*”)
- See also 6:24-50, 14:47-15:6.

116. See my discussion of elements [1.pre] and [1.c] (§§ IX.A.3(a) and (d)) for further discussion on Tso’s disclosures of this element. Tso also discloses that, after generating the “transcoded version” of the content item, the server “transmits” a “data stream for the transcoded hypertext object to network client 12.” E.g., Tso, 6:51-63, 10:24-49.

(h) Element [2.pre]

117. **Element [2.pre]** recites **“The method of claim 1, further comprising:”** See my discussion of claim 1 above in Sections IX.A.3(a)-(f) (the same process is followed). This element is met by Tso in view of Huang.

(i) Element [2.a]

118. **Element [2.a]** recites **“receiving a second request from a second playback device for media content; wherein the second request contains information, the information indicating a second original media content, second set of content generation operations, and second set of transformation operations.”** This element is met by Tso.

119. **Tso discloses receiving a second request from a second playback device for media content** (*e.g.*, second “network client” sends “HTTP request[]” for “image” or “video” “data content”). *See, e.g.*, Tso:

- 6:51-63 (“Multiple-thread processing improves the efficiency of the present embodiment by not waiting for a hypertext object to be received in its entirety by HTTP remote proxy 36, or added in its entirety to server-side cache memory 30, before beginning to send the object to network client 12. Another benefit of multiple-thread processing is that *parser 22 may efficiently process requests for the same hypertext object from multiple network clients 12. The hypertext object need only be retrieved from Internet 18 once, and appropriate versions may be transmitted to such multiple network clients 12 concurrently.* It should be noted, however, that embodiments of the present invention may be implemented without multiple-thread processing.”)
- 3:45-60 (“[T]ranscoder 20 is coupled to HTTP remote proxy 36. Parser 22 manages the transcoding of data to be transmitted from transcoding server 34 to network client 12. To this end, parser 22 controls *transcode*

service providers 24 to selectively transcode content based on a predetermined selection criterion. For example, one or more transcode service providers 24 may provide the capability to compress and/or scale different types of data content, such as image, video, or HTML (HyperText Markup Language).”)

120. Also see my discussion of elements [1.a], [1.b] (§§ IX.A.3(b)-(c)). Tso also discloses that “multiple network clients 12” send “requests” for media content, as I discussed within element [1.a]. *See* Section IX.A.4(b). The remainder of this element simply recites that the second request contains the same types of information contained in the first request discussed for element [1.b]. *See* Section IX.A.3(c).

(j) Element [2.b]

121. **Element [2.b]** recites “wherein the second request is received substantially concurrently with receiving the first request.” This element is met by Tso.

122. Tso discloses wherein the second request is received substantially concurrently with receiving the first request (*e.g.*, “process requests” from “multiple network clients” and “concurrently” deliver “appropriate versions” of content to each of those “clients”). *See, e.g.*, Tso:

- 6:51-63 (“Multiple-thread processing improves the efficiency of the present embodiment by not waiting for a hypertext object to be received in its entirety by HTTP remote proxy 36, or added in its entirety to server-side cache memory 30, before beginning to send the object to network client 12. Another benefit of multiple-thread processing is that *parser 22 may efficiently process requests for the same hypertext object from multiple network clients 12. The hypertext object need only be retrieved from Internet 18 once, and appropriate versions may be*

transmitted to such multiple network clients 12 concurrently. It should be noted, however, that embodiments of the present invention may be implemented without multiple-thread processing.”)

- See also 4:62-5:7.

123. See my discussion of element [1.a] (§ IX.A.3(b)) for further discussion on Tso’s disclosures of this element. Tso also discloses using “multiple-thread processing” to “concurrently” handle and transmit appropriate versions of content to multiple clients in response to client requests for content. When multiple clients issue requests that are “concurrently” handled by processing and serving content for those requests, Tso discloses the second request is received substantially concurrently with the first.

(k) Element [2.c]

124. **Element [2.c] recites “wherein the first original media content and the second original media content are the same media content.”** This element is met by Tso.

125. **Tso discloses wherein the first original media content and the second original media content are the same media content** (*e.g.*, two network clients request “the same hypertext object”). *See, e.g.*, Tso:

- 6:51-63 (“Multiple-thread processing improves the efficiency of the present embodiment by not waiting for a hypertext object to be received in its entirety by HTTP remote proxy 36, or added in its entirety to server-side cache memory 30, before beginning to send the object to network client 12. Another benefit of multiple-thread processing is that *parser 22 may efficiently process requests for the same hypertext object from multiple network clients 12. The hypertext object need only be*

retrieved from Internet 18 once, and appropriate versions may be transmitted to such multiple network clients 12 concurrently. It should be noted, however, that embodiments of the present invention may be implemented without multiple-thread processing.”)

126. Tso discloses that two different clients request are concurrently served with appropriate versions of the same original media content. Tso, 6:51-63.

(l) Element [2.d]

127. **Element [2.d] recites “determining whether a second intermediate media content is available having been previously created using the second original media content and the second set of content generation operations.”**

This element is met by the combined teachings of Tso in view of Huang.

128. Tso in view of Huang renders obvious this element for the reasons that I explained for element [1.c] in section IX.A.3(d). Element [1.c] and this element recite making similar determinations with respect to the “first” and “second” intermediate content, respectively.

(m) Element [2.e]

129. **Element [2.e] recites “responsive to determining that the second intermediate media content is available, creating a second optimized media content for the first playback device by performing the second set of transformation operations on the second intermediate media content.”** This element is met by the combined teachings of Tso in view of Huang.

130. Tso in view of Huang renders obvious this element for the reasons that I explained for element [1.d] in section IX.A.3(e). Element [1.d] and this element recite performing similar steps with respect to the first and second intermediate content and optimized media content, respectively.

(n) Element [2.f]

131. **Element [2.f] recites “responsive to determining that the second intermediate media content is not available, creating a second optimized media content for the first playback device by creating the second intermediate content using the second original media content and the second set of content generation operations, and performing the second set of transformation operations on the second intermediate media content.”** This element is met by the combined teachings of Tso in view of Huang.

132. Tso in view of Huang renders obvious this element for the reasons that I explained for element [1.e] in section IX.A.3(f). This element recites that the “second” intermediate media content is generated the same way that the “first” intermediate content is generated as recited in element [1.e].

(o) Element [2.g]

133. **Element [2.g] recites “sending the second optimized media content to the second playback device.”** This element is met by Tso.

134. Tso discloses this element for the reasons that I explained for element [1.f] in section IX.A.3(g) above. The same process for sending the second optimized media content is followed in this element.

(p) Claims 3 and 4

135. Claims 3 and 4 differ in that in claim 3, the first and second content generation operations are “the same,” and in claim 4, the first and second content generation operations are “different,” and recite:

a. Claim 3: “The method of claim 2, wherein the first set of content generation operations and the second set of content generation operations are the same, and the first set of transformation operations and the second set of transformation operations are different.”

b. Claim 4: “The method of claim 2, wherein the first set of content generation operations and the second set of content generation operations are different, and the first set of transformation operations and the second set of transformation operations are different.”

136. Tso discloses that the first set of content generation operations and the second set of content generation operations are [the same/different] (*see* [1.b] (§ IX.A.3(c)), [2.a] (§ IX.A.3(i)), the same or different content generation

operations), **and the first set of transformation operations and the second set of transformation operations are different** (*see* [1.b] (§ IX.A.3(c)), [2.a] (§ IX.A.3(i)), different transformation operations). *See, e.g., Tso:*

- 6:50-63 (“Multiple-thread processing improves the efficiency of the present embodiment by not waiting for a hypertext object to be received in its entirety by HTTP remote proxy 36, or added in its entirety to server-side cache memory 30, before beginning to send the object to network client 12. Another benefit of multiple-thread processing is that *parser 22 may efficiently process requests for the same hypertext object from multiple network clients 12. The hypertext object need only be retrieved from Internet 18 once, and appropriate versions may be transmitted to such multiple network clients 12 concurrently.* It should be noted, however, that embodiments of the present invention may be implemented without multiple-thread processing.”)
- 8:10-13 (“*Applying the above-listed selection criteria, or combinations thereof, embodiments of the present invention may be used to provide a virtually limitless range of dynamic transcoding services.*”)

137. Claims 3 and 4 recite the same claim elements, with the exception that in claim 3 the first and second content generation operations are “the same,” while in claim 4 the first and second content generation operations are “different.” As I explained for elements [1.b], [1.d], and [1.e] (§ IX.A.3(c), (e)-(f)), Tso also discloses specifying different “content generation” and “transformation” operations using different “selection criteria.” Tso, 6:64-7:14, 7:15-9. Because combinations of such criteria are disclosed as being used to generate “virtually limitless” transformations, a person having ordinary skill in the art would have understood that Tso discloses a second user request for content that specifies either the same or different content

generation operations and different transformation operations from the first set of operations—for example, different requests for the same content item may require generating a different data type, number of colors, or resolution for the requested content. For example, when two devices running different operating systems request the same content, different image formats (i.e., data types generated using content generation operations—see my discussion of element [1.b] in section IX.A.3(c)) are served to each of the devices, and the images are scaled differently depending on screen size (i.e., different resolutions generated using different transformation operations—see my discussion of element [1.b] in section IX.A.3(c)) and color resolution (e.g., a black and white image or lower color bit depth image is generated for a device with a black and white or lower color resolution screen). Similarly, if the two devices are running the same operating system but have different screen resolutions, the content generation operations are the same because the data types are the same, while the transformation operations may be different due to different screen resolutions.

(q) Element [5.a]

138. **Element [5.a] recites “The method of claim 1, further comprising: determining whether a previously-generated first optimized media content is available for reuse, wherein the previously-generated first optimized media content was created using the first original media content, the first content**

generation operations, and the first transformation operations.” This element is met by Tso.

139. **Tso discloses determining whether a previously-generated first optimized media content is available for reuse** (*e.g.*, “determine whether a copy of the required version already resides in server-side cache memory 30”; *see* my discussions of elements [1.c] (Section IX.A.3(d)) and [1.e] (Section IX.A.3(f))), **wherein the previously-generated first optimized media content was created using the first original media content, the first content generation operations, and the first transformation operations** (*e.g.*, previously-generated “appropriate version[]” generated using the same “network client” “selection criterion” and “content characteristics”; *see* [1.b]-[1.e] (Sections IX.A.3(c)-IX.A.3(f))). *See, e.g.*, Tso:

- 10:24-49 (“Upon being invoked, *parser 22* first calls cache interface 28 with the requested hypertext object to *determine whether a copy of the required version already resides in server-side cache memory 30*. For purposes of illustration, assume no entry exists in server-side cache memory 30 for the requested hypertext object. *HTTP remote proxy 36 then invokes a call to retrieve the hypertext object from Internet 18* over server/network communications link 16. ... *Transcode service provider 24 then transcodes the data* stream appropriately, *and* HTTP remote proxy 26 *transmits the transcoded data stream to network client 12.*”)
- *See also* 3:43-60, 4:62-66, 5:36-42, 6:9-50, 6:64-66, 14:21-63, 14:37-63.

140. *See* my discussion of claim 1 in sections IX.A.3(a)–(g) for further discussion on Tso’s disclosures of this element. As I discussed for elements [1.b] to

[1.e], Tso also discloses that the parser on a server is used to search for the required version of the content, which includes searching for a version of the content that was generated using the same content generation and transformation operations, in server-side cache memory.

(r) Element [5.b]

141. **Element [5.b] recites “responsive to determining that the previously-generated first optimized media content is available, sending the previously-generated first optimized media content to the first playback device.”** This element is met by Tso.

142. **Tso discloses responsive to determining that the previously-generated first optimized media content is available** (*e.g.*, if “the required version already resides in” cache), **sending the previously-generated first optimized media content to the first playback device** (*e.g.*, sending “transcoded versions ... without the need to ... re-transcode the content”). *See, e.g.*, Tso:

- 4:1-5 (“Server-side cache memory 30 may be used to *store* both original and *transcoded versions of content for later transmission to network client 12 without the need to re-retrieve the content from Internet 18 or to re-transcode the content.*”)
- 10:24-49 (“Upon being invoked, *parser 22* first calls cache interface 28 with the requested hypertext object to *determine whether a copy of the required version already resides in server-side cache memory 30.* For purposes of illustration, assume no entry exists in server-side cache memory 30 for the requested hypertext object. *HTTP remote proxy 36 then invokes a call to retrieve the hypertext object from Internet 18* over server/network communications link 16. ...

Transcode service provider 24 then transcodes the data stream appropriately, and HTTP remote proxy 26 transmits the transcoded data stream to network client 12.”).

143. See my discussion of elements [1.d], [1.f], and [5.a] in sections IX.A.3(e), (g), and (q) for further discussion of Tso’s disclosures of this element. Tso also discloses transmitting an already-existing cached version to the client when the “required version” is already cached in order to avoid re-retrieving and re-transcoding of the content. *E.g.*, Tso, 4:1-5.

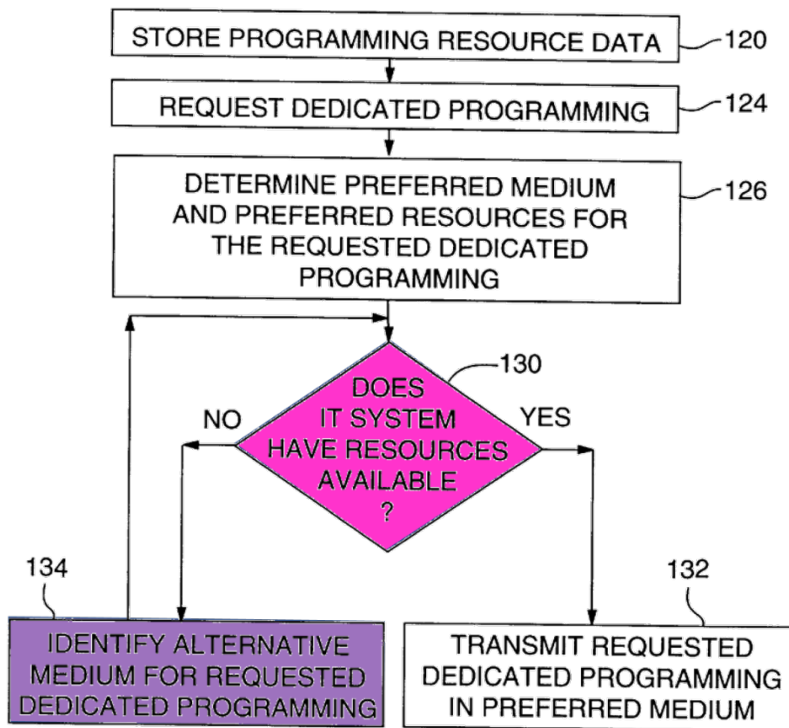
B. Ground 2: Tso in view of Huang and Lawler Renders Obvious Claims 6-7

144. Claims 6-7 of the ’745 patent are obvious over Tso in view of Huang and Lawler. Tso in view of Huang discloses a system for dynamically and incrementally transcoding new and cached data transmitted between two computers over a communications link. Lawler discloses an interactive television or televideo (IT) system that accounts for the occurrence of overloading the capacity of a CPU. Just like the image and video content handled by Tso and Huang (*e.g.*, Tso, 3:51-54; Huang, 1:48-52, 10:20-28), this IT system handles video as well as “image” content ranging from “individual image frames” to other “graphics images” such as “bitmap images.” Lawler, 4:59-5:6. Thus, as I have further set forth below, Tso in view of Huang and Lawler renders obvious claims 6-7 of the ’745 patent.

1. U.S. 5,905,522 (“Lawler”) Overview

145. Lawler teaches an interactive television system, which Lawler also describes as a televideo (IT) system providing image and video content, that uses an alternative format of content to avoid the situation in which “[t]he capacity of CPU 66” is exceeded when “too many processor cycles are required to run existing or requested processing threads.” Lawler, 4:59-5:6, 5:57-62; 10:3-8. Figure 4 below is annotated to summarize Lawler’s teachings of how to address this CPU capacity problem:

FIG. 4



146. Lawler discloses that, when a system determines that it “lacks the resources required to provide the service in the preferred format,” the system will use “an alternative format that requires fewer IT system resources.” Lawler, 6:14-

16, 2:53-55. “Preferred” and “alternative” formats in Lawler refer to different encoding formats that affect, for example, the quality of the media content. Table 1 from Lawler, annotated below, provides examples of alternative formats including, for example, lower, “VHS” quality streaming video instead of higher, “S-VHS” quality streaming video or, as another example, an 8-bit “final image” instead of a 24-bit “graphic code” for an “animation sequence” (Lawler, 6:2-35):

TABLE 1	
Preferred and Alternative Modes	
Preferred Format	Alternative Format
S-VHS quality MPEG stream	VHS quality MPEG stream
24-bit graphic code resource for graphics animation sequence	8-bit graphic code resource for final image of graphics animation sequence
CD-quality sound	telephone-quality sound
8-bit graphic	1-bit black and white dithered graphic

147. Lawler also discloses varying the quality of the content by changing the resolution of graphics for the alternative format for media content. Lawler, 6:7-12 (“The preferred format of dedicated programming services typically provides higher resolution or quality at the expense of greater IT system resources requirements. The alternative format provides the dedicated programming services at a lower resolution or quality that requires fewer IT system resources.”).

2. Motivations to Modify Tso in view of Huang with Lawler's Teachings

148. Like Tso and Huang and as discussed above, Lawler is analogous art in the same field as the '745 patent, networked media delivery, and is reasonably pertinent to automating media delivery and reducing the overhead and time required to generate Web pages. Lawler, 1:20-26 (“allocating resources of an interactive television or televideo (IT) system for delivering programming to viewers”), 2:31-33; 3:1-4, 4:59-5:6; *see* §§IX.A.1-2.

149. Both Tso and Huang teach the importance of considering the amount of processing resources that are available when determining how to transcode content. *E.g.*, Tso, 7:15-31; Huang, 6:63-67, 2:9-13. Lawler's teachings of generating “alternative” formats builds upon Tso's teaching of determining whether a host computer has available processing resources to perform the requested transcoding. Tso, 7:15-31. Specifically, Lawler teaches switching to generating and using an “alternative” format that consumes fewer resources when restrictions relating to CPU capacity are met in order to avoid “interrupting acceptable delivery of the programming.” Lawler, 5:40-6:18, 2:47-61, 9:36-41 (“Alternatively, CPU 66 in interactive station controller 20 determines whether it and memory system 68 have sufficient available resources for the requested programming and, if necessary, modifies the request sent to central control node 12 accordingly.”). For example, the alternative format for a media content item may vary from the preferred format

based on the bit rate and resolution of the content. Lawler, 6:7-12, 6:37-50. Lawler's teachings of using alternative formats also further builds upon Huang's teachings of improving the speed and efficiency of the system because the alternative formats are used depending on what processing resources are available in order to better use those resources. Lawler, 6:14-16, 2:53-55; Huang, 6:63-67, 2:9-13. A person having ordinary skill in the art would have been motivated to modify Tso (including with Huang's teachings applied to Tso) with Lawler's teachings of using an "alternative" format that consumes fewer resources to avoid "interrupting acceptable delivery of the programming" for at least the following reasons. Lawler, 2:47-61, 5:40-6:18, 9:25-35.

150. **First**, Tso already teaches using "characteristics of [a] transcoding server" including "system load" as a "selection criterion" used in order to determine how to transcode media content. Tso, 7:15-31. Lawler provides a complementary teaching of using "system load" including "CPU" capacity to determine whether to instead use an "alternative format" of the media content that consumes fewer resources. Lawler, 5:57-6:18. A person having ordinary skill in the art would have found it obvious based on these complementary teachings to select an "alternative format" that takes less CPU computing time to perform when CPU capacity use is too high, in order to reduce the load on CPU capacity caused by processing.

151. **Second**, Tso and Lawler both teach that it is advantageous to efficiently and reliably serve media to clients, something that was well-known in the art. Tso, 6:55-58 (“efficiently process[ing] requests ... from multiple network clients” is a “benefit”); Lawler, 11:42-50 (describing the beneficial ability to “allocate IT system resources” while “delivering a substantial number of dedicated programming services simultaneously”), 2:1-9 (“disruption of programming is highly undesirable”). Indeed, it was well-known to a person having ordinary skill in the art that it is advantageous to monitor processing resources as part of a process to ensure minimum levels of service. For example, U.S. Patent No. 6,144,996, which names Starnes as its first named inventor, a copy of which I understand has been designated as Exhibit 1038, teaches that it is advantageous to “guarantee a minimum level of service (for content delivery) to the requestors” in order to allow the system to “scale[] to provide additional processing resources” by monitoring the system’s “utilization of processing resources.” Starnes, Abstract, 2:66-3:5, 4:55-5:4, 6:14-18. Lawler teaches that switching to an “alternative medium” when CPU capacity is “met” advantageously avoids “interrupting acceptable delivery of the programming.” Lawler, 5:57-6:50. A person having ordinary skill in the art would therefore be motivated to modify Tso with Lawler’s teachings in order to better allocate resources through using alternative formats of transcoding that better

guarantees that all recipients of media content receive a better minimum level quality of service.

152. **Third**, Lawler provides complementary teachings that using an alternative, lower-quality content format advantageously uses “fewer IT system resources,” including as little as “one-half” of the requirements for some resources. Lawler, 6:7-50, 2:53-61. A person having ordinary skill in the art would have found it obvious to modify Tso to apply Lawler’s teachings of such alternative content forms (as an example, content with reduced quality based on resolution or lower bitrates) to advantageously reduce “system load” and increase the system’s capacity to process content. *Id.*; Tso, 7:15-31. It was well known in the art to deliver lower quality content to fulfill requests during times of processor capacity constraints. For example, Starnes teaches that users will perform a “content negotiation” to ask a server to send, for example, a “less robust” or “lower quality” content item in order to “facilitate[] improved bandwidth utilization and responsiveness of the Internet.” Ex. 1038, Starnes, 2:24-41. It was thus already well-understood in the art to selectively transcode content based on criteria such as process capacity and bandwidth to better allow for the efficient generation and provision of web pages to users over the Internet.

153. **Fourth**, Lawler provides complementary teachings that the alternative content format may vary from the preferred format by being, for example, a lower-

bit graphical or lower resolution format of the content. Lawler, 6:7-12, 6:37-50. A person having ordinary skill in the art would have been motivated to modify Tso with these teachings because these are additional detailed teachings of “content characteristics” and “content provider preferences” for a content provider to change the quality of content. Tso, 7:15-8:9.

154. In light of the above teachings, a person having ordinary skill in the art would have had a reasonable expectation of success in applying Lawler’s teachings of using an alternative content format depending on the capacity of the system. Indeed, Tso discloses that the selection criteria used to select transcode service providers include a “current processing load” of the system (Tso, 7:7-14), and also teaches that higher quality transcoding services require the use of additional resources. Tso, 16:50-54 (“more resource-intensive compression algorithms” are used to “provide better quality content”). Modifying Tso with Lawler’s teachings is therefore merely a straightforward application of using Tso’s teachings of “current processing load” as a selection criterion when determining how to transcode content by selecting, based on that processing load, an alternative format of content that requires fewer resources. This advantageously improves the efficiency of the system in servicing many user requests.

3. **Invalidity of Claims 6-7 Over Tso in view of Huang with Lawler's Teachings**

(a) **Element [6.a]**

155. Element [6.a] recites “The method of claim 1, further comprising: **determining whether the host computer has sufficient processing resources to create the first optimized media content for the first playback device.**” This element is met by the combined teachings of Tso and Lawler.

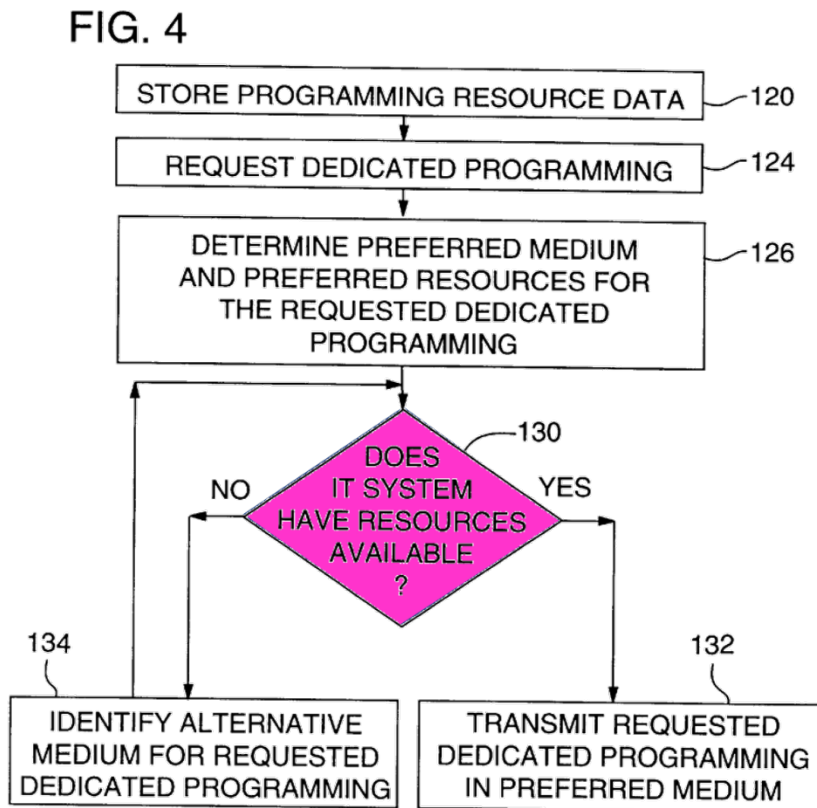
156. Tso discloses **determining whether the host computer has sufficient processing resources** (e.g., “dynamically updat[ing]” the “current processing load” to use as a “predetermined selection criterion”) **and creating the first optimized media content for the first playback device** (*see* [1.d]-[1.e], §IX.A.3(e)-(f)). *See e.g.,* Tso:

- 7:7-31 (“Still further, *the predetermined selection criterion may comprise* a condition of the device on which transcoding server 34 is installed (for example, *a current processing load*), a condition of a device to which transcoding server 34 is coupled, or a condition of a communications link. Transcoding server 34 may provide the ability to *dynamically update such predetermined selection criteria*. The following discussion provides still more examples of the types of information which may be used to dictate which of transcode service providers 24 are invoked. It should be noted, however, that these examples are provided by way of illustration only, and are not intended to limit in any way the scope of the invention claimed herein. The predetermined selection criterion may comprise: (1) network client 12, such as a display dimension, resolution, number of colors, processor type, memory/disk configuration, modem or network interface type, installed add-in boards (for example, hardware compression/decompression), software configuration (for example, availability of pre-installed software decompression modules), physical location/proximity (for example, as determined by a telephone area code), and user identity; (2)

characteristics of transcoding server 34 or some other network server, including system load and identification information (for example, the owner of the server) ...”)

157. Lawler discloses determining whether the host computer has sufficient processing resources for the first optimized media content for the first playback device (e.g., making “an inquiry as to whether” “CPU 66” has “sufficient capacity” to meet the “ProcessorRequirement”). See e.g., Lawler:

- Fig. 4



- 9:25-35 (“Decision block 130 represents an *inquiry as to whether* IT system 10 has available the resources to meet the resource requirements of the preferred medium. ... [C]entral control node ... inquires of **CPU 66** whether it and memory system 68 at station controller *have sufficient capacity to meet the* RAMRequirement and *ProcessorRequirement*.”)

- *See also* Abstract, 10:3-8, 2:53-3:3, 2:1-4, 5:40-53.

158. See my discussion of claim 1 and elements [1.d] and [1.e] in Sections IX.A.3(e)-(f) for further discussion on Tso's disclosures of this element. Tso also discloses that one of the "selection criteria" used by the parser to determine how to transcode content is the "current processing load" at the server. This processing load is "dynamically update[d]" in order to reflect the current processing load when selecting a transcode service provider. Tso, 7:7-31.

159. Lawler further discloses that when a server processes content to be provided, the server determines whether there is sufficient CPU capacity to use the preferred content format. Lawler, Fig. 4, 9:24-27. As I discussed above in Section IX.B.2, a person having ordinary skill in the art would have been motivated to modify Tso with this teaching based on Tso's complementary teaching that "CPU capacity" is used as a selection criterion. Modifying Tso with Lawler's teachings as I discussed above in Section IX.B.2, it would have been obvious to use Tso's CPU capacity selection criterion to confirm whether the system has sufficient "CPU capacity" when determining whether the system has sufficient available "processing load" to perform the transcoding required to generate the preferred (i.e., optimized) format of content, rendering obvious this claim element.

(b) Element [6.b]

160. Element [6.b] recites “responsive to determining that the host computer does not have sufficient processing resources to create the first optimized media content, determining an alternate first set of content generation operations or an alternate first set of transformation operations.”

This element is met by the combined teachings of Tso in view of Lawler.

161. Tso discloses creating the first optimized content (*see* [6.a] (§IX.B.3(a))) and a first set of alternate content generation operations or first set of alternate transformation operations (*see* [1.b]-[1.d], claims [3], [4] (§IX.A.3(c)-(e), (p))).

162. Lawler discloses responsive to determining that the host computer does not have sufficient processing resources (*see* [6.a] (§IX.B.3(a))), determining an alternate media content (*e.g.*, “identif[ying]” “an alternative format that requires fewer IT system resources”). *See e.g.*, Lawler:

- 2:53-61 (“The IT system also *identifies alternative resources for delivering the programming in an alternative format that requires fewer IT system resources*. For example, *an alternative format to the preferred high-resolution digital video signal format could be a conventional-resolution digital video signal that corresponds to a standard VHS quality video signal*. Such an alternative format could require only one-half the level of some resources required by the preferred format.”)
- 9:25-35 (“Decision block 130 represents an *inquiry as to whether* IT system 10 has available the resources to meet the resource requirements of the preferred medium. ... [C]entral control node ... inquires of *CPU*

66 whether it and memory system 68 at station controller *have sufficient capacity to meet the* RAMRequirement and *ProcessorRequirement.*”)

- *See also 6:13-18.*

163. As I discussed for claim elements [1.b], [1.c], and [1.d] and claims 3 and 4 in sections IX.A.3(c)-(e) and (p), Tso discloses a system that uses combinations of different selection criteria to perform a “virtually limitless range of dynamic transcoding services.” Tso, 3:48-65, 7:15-17, 8:10-14. Selection criteria used by the system include “current processing load,” “file size,” and “resolution”. *Id.* When different selection criteria are used to perform different transcoding operations, this results in transcoding performing different content generation operations, transformation operations, or both. *See id.* A person having ordinary skill in the art would have understood that using “current processing load,” content generation operations (for example, “file size”), and transformation operations (for example, “resolution” or “number of colors”) as selection criteria discloses using, for example, a smaller file size and/or a lower resolution based on “processing load” as alternative content generation and/or transformation operations, respectively.

164. In addition to Tso’s and Lawler’s teachings of determining that a server does not have sufficient processing resources, which I discussed with respect to element [6.a], and Tso’s teachings of “virtually limitless” transformations using different content generation and transformation operations, Lawler further discloses that when CPU capacity is consumed such that the system cannot generate a

preferred format of content, the system instead uses an alternative format for content that requires fewer server resources. Lawler, 2:53-61. By using the alternative format, the system is able to still generate and provide users with content when limited processing resources would otherwise prevent the system from doing so. *See id.* As I discussed in Section IX.B.2, a person having ordinary skill in the art would have been motivated to modify Tso with this teaching to advantageously allow the system to reliably provide higher quality content to several customers, yielding a system in which the system uses an alternative format for media content when the system's selection criterion based on CPU capacity indicates that the server does not have sufficient resources to generate the preferred format for the requested content. *See* Lawler, 2:53-3:3; Tso, 7:7-31.

(c) Element [6.c]

165. **Element [6.c] recites “creating an alternate first optimized media content using the alternate first set of content generation operations or the alternate first set of transformation operations.”** This element is met by Tso.

166. **Tso discloses creating an alternate first optimized media content using the alternate first set of content generation operations (see [6.b], [1.b]-[1.d]) or the alternate first set of transformation operations (see [6.b], [1.b]-[1.c]).**

167. Tso discloses generating content using content generation and transformation operations specified by the selection criteria. Accordingly, when an alternative content format is selected, the content is generated using the same transcoding process previously described, but using different content generation and transformation operations. See my discussion of elements [1.b], [1.c], and [1.d] above in Sections IX.A.3(c)-(e). This includes the alternate operations specified as discussed in [6.b], as discussed in Section IX.B.3(b).

(d) Element [7.a]

168. Element [7.a] recites “**The method of claim 6, wherein the first optimized media content is at a first level of quality and the alternate first optimized media content is at a second level of quality, wherein the first level of quality is higher than the second level of quality.**” This element is met by Lawler.

169. Lawler further discloses **the first optimized media content is at a first level of quality (e.g., “preferred format” is “higher resolution or quality”), and an alternate first optimized media content is at a second level of quality, wherein the first level of quality is higher than the second level of quality (e.g., “[t]he alternative format” is “lower resolution or quality”).** See e.g., Lawler:

- 6:7-12 (“The *preferred format* of dedicated programming services typically *provides higher resolution or quality* at the expense of greater IT system resources requirements. *The alternative format provides the dedicated programming services at a lower resolution or quality that requires fewer IT system resources.*”)

- *See also* 2:53-61.

170. Lawler discloses that when the first, “preferred” format is generated, this format is “higher” quality at the expense of taking more system resources, while the “alternative” format is of “lower” quality with the benefit of consuming fewer system resources. Lawler, 6:7-12. As I discussed above in Section IX.B.2, a person having ordinary skill in the art would have been motivated to modify Tso with Lawler’s teachings in order to generate an “optimized” first media content at a second, lower level of quality when there are insufficient processing resources for the first level of quality to, for example, advantageously improve the processing speed and efficiency of the system. This results in the system being able to reliably serve multiple requests that would otherwise not be fully supported because processing resources are being consumed. Lawler, 6:14-16, 2:53-55

(e) Element [7.b]

171. **Element [7.b] recites “wherein a level of quality of a media content is measured based on a compression format, a bit rate, and an image resolution of a media content sent to a playback device.”** This element is met by the combined teachings of Tso in view of Lawler.

172. **Tso discloses wherein a level of quality of a media content is measured based on a compression format** (*e.g.*, use “more resource-intensive

compression algorithms” for “better quality content”) **of a media content sent back to a playback device** (e.g., “content” sent back to system “users”). *See e.g.*, Tso:

- 16:50-54 (“Still further, transcoding server 34 may *use more resource-intensive compression algorithms*, for example, *to provide better quality content* for the same latency at the expense of slowing down access for non VIP users.”)
- *See also* 7:31-33, 8:25-28, 11:19-24, 12:26-32, 12:61-13:5, 16:2-5.

173. Lawler discloses wherein a level of quality of a media content is measured based on a bit rate (e.g., quality is measured based on “24-bit” vs. “8-bit”), and an image resolution of a media content sent to a playback device (e.g., “high-resolution” vs. “conventional-resolution” of “programming services” sent back to a user’s device). *See e.g.*, Lawler:

- 2:53-61 (“*[A]n alternative format to the preferred high-resolution digital video signal format could be a conventional-resolution digital video signal that corresponds to a standard VHS quality video signal.* Such an alternative format could require only one-half the level of some resources required by the preferred format.”)
- 6:7-12 (“The *preferred format* of dedicated programming services typically *provides higher resolution or quality* at the expense of greater IT system resources requirements. *The alternative format provides the dedicated programming services at a lower resolution or quality* that requires fewer IT system resources.”)
- 6:37-50 (“For example, a preferred format of S-VHS quality digital video for video on-demand previews could have a first alternative format of VHS quality digital video, and subsequent *alternative modes in the form of 24-bit graphics and 8-bit graphics.*”)
- *See also* 7:44-58, 4:59-65, Tables 1, 3A-3F.

174. Tso discloses selecting more “resource-intensive compression algorithms” to generate “better” quality media content, which describes a quality of media measured based upon a compression format. Tso, 16:50-54.

175. Lawler further discloses specifying a “higher” quality of a “preferred” formats based on differences in bitrate or resolution. For example, a lower-quality alternative content to be sent back to a user’s device may specify a lower bitrate, such as an 8-bit rather than 24-bit, or a lower resolution. Lawler, 6:7-12, 2:53-61.

176. As discussed above in Section IX.B, a person having ordinary skill in the art would have found it obvious to measure quality using Lawler’s teachings of bitrate and resolution, in addition to Tso’s teaching of compression in order to advantageously allow the system the flexibility to select better alternative quality content for delivery. This yields a system in which the quality of media content is measured based on compression format as taught by Tso, as well as both bit rate and image resolution as taught by Lawler.

C. Grounds 3 and 4: Samaniego Anticipates and Renders Obvious Claims 1-7

1. Overview of Samaniego

177. As previously stated in Section I Samaniego is a publication of U.S. Patent Application No. 09/929,904. Although the face of the ’745 patent identifies the ’745 patent as a “continuation-in-part” of the Samaniego application, none of the inventors named on the ’745 patent is named as inventors on Samaniego. As also

previously stated, I have therefore been asked to assume that Samaniego qualifies as prior art to the '745 patent and, based on that assumption, consider whether Samaniego alone and alternatively in view of other prior art discloses and renders obvious the Challenged Claims. I have not formulated any separate opinion as to whether Samaniego qualifies as prior art.

178. I have reviewed Samaniego and compared the language in its specification to the language in the '745 patent. Based on my review, the text of the specifications of the '745 patent and Samaniego do not have any substantive differences, as the only changes to the text are syntactical edits and a paragraph setting forth priority claims for the '745 patent. I have prepared a redline document showing the language that was changed in the '745 patent's specification, which I understand has been designated as Exhibit 1013. With respect to figures, a comparison of Samaniego's and the '745 patent's figures shows that '745 patent's figures are formalized versions of the figures that are in Samaniego. *Compare* '745 and Samaniego Figs. 1-23. I have not identified any material in the '745 patent's figures that disclose material that is not already disclosed in Samaniego. As a result, if the '745 patent's specification supports the Challenged Claims, then Samaniego anticipates the Claims because it has those same disclosures. Indeed, during prosecution, as I stated above in Section V.B, the Examiner reviewed the disclosures

from Samaniego and stated in the Notice of Allowance that the application published as Samaniego supports each of the Challenged Claims:

[T]his application does not receive the prior date of App. 09425326 filed 21 October 1999 since it claims additional disclosure not presented in the prior application. Furthermore, *support for the claimed subject matter has been found in 12238842, 12173747, 11269916 and 09929904. Thus, 13752110 receives the priority date of App. 09929904.*

Ex. 1002, 165. However, for certain elements, I have also considered Samaniego's teachings in combination with other prior art teachings in formulating my opinions as set forth in Ground 4 below.

2. Invalidity of Claims 1-7 in View of Samaniego

179. In light of the fact that Samaniego's specification is substantially identical to the '745 patent's specification, in my analysis below, I have color coded the pertinent sections of Samaniego's specification to show the specification sections corresponding to the claim language of Challenged Claims 1-7, and have provided additional explanation as helpful to explain a person having ordinary skill in the art's understanding of the disclosures in Samaniego.

(a) Element [1.pre]

180. As set forth in the table below, Samaniego discloses this element:

[1.pre] A method in a host computer for developing transformation processing operations to optimize media content playback to a plurality of playback devices connected with the host computer in a network, the method comprising:

• Fig. 1:

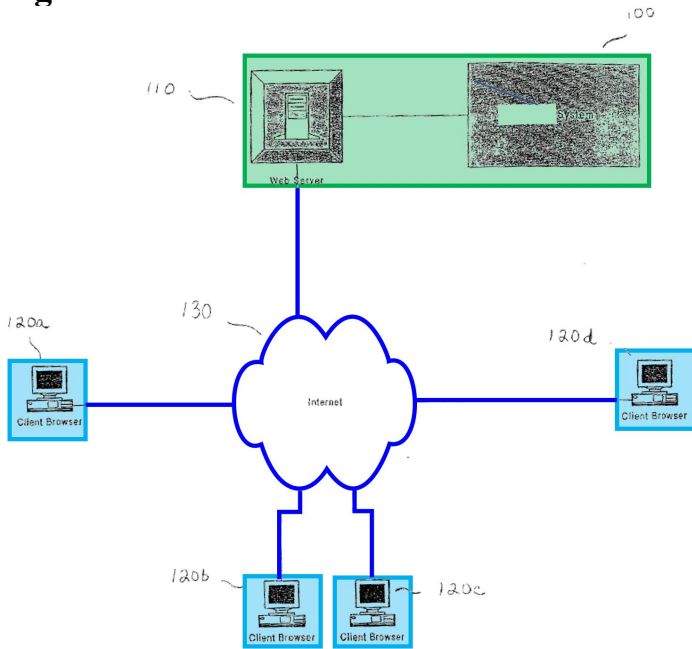


Fig. 1

- [0002] (“[T]he invention relates to an **Internet server-based software system that provides delivery of automated graphics and other media** to Web sites for access by an end user or consumer.”)
- [0094] (“The **system 100 is attached to a Web server 110, which is connected to multiple client browsers 120(a-d) via the Internet 130.**”)
- [0148] (“FIG. 21 is a schematic diagram showing the process flow of a proprietary enabled **page delivered to a Web browser** according to a preferred embodiment of the invention. Original media 200 is created and placed into the system 100 in a media repository 2040. **A content generation procedure 2140 is created with instructions on how the media is to be transformed to create the desired Web page content.**”)

	• [0065] (“In addition, generated media is cached such that further requests for the same media require little overhead.”) • <i>See also</i> [0002], [0049], [0091], [0096]-[0097], [0137], [0140], [0148], Figs. 3, 7, 9, 21.
--	--

181. As annotated in Figure 1 in the above table, the method disclosed in Samaniego includes multiple “**Client Browsers**,” each of which is connected over the “**Internet**” to a “**Web Server**.” *See* Fig. 1. The Web Server has an “Internet-based software system” that is “attached to” (i.e., installed on) the Web Server. Samaniego, [0002], [0094]. The system is used for “**delivery of automated graphics and other media**” that has been “transformed” to “create the desired Web page content” as requested by clients.

(b) Element [1.a]

182. As set forth in the table below, Samaniego discloses this element:

[1.a] receiving a first request from a first playback device for media content;	• [0148] (“The browser 120 requests the Web page 301 from the Web site 110. Upon receipt of the page 301, the browser contacts the system 100 requesting media specified in the URL.”) • <i>See also</i> [0108]-[0110], [0138], [0143], Figs. 7, 9.
--	---

183. Samaniego discloses that a browser issues a request to the server for media content specified in a URL.

(c) Element [1.b]

184. As set forth in the table below, Samaniego discloses this element:

[1.b] wherein the first request contains information, the information indicating a first original media content, first content generation operations, and first transformation operations;	• [0066] (“[T]he invention takes the original media, content generation procedures, and proprietary URL tags as inputs for generating the Web media.”) • [0097] (“Web images 420 are Web versions of the original images. Web images 420 are ready for retrieval by the Web server 110 to be delivered to a Web browser 120. Any image to be modified or updated must pass through the herein above three components before it can be delivered to the Web browser 120. HTML pages 460 have references to Web images 420.”) • [0148] (“The browser 120 requests the Web page 301 from the Web site 110. Upon receipt of the page 301, the browser contacts the system 100 requesting media specified in the URL. The system parses the URL 2100 to determine the content generation procedure 2140 to execute, any corresponding input parameters to be used by such procedure, any dynamic content processing 2150 to be performed by dynamic media procedures, any user profile information 2160 to be used to modify the resulting image, and any cache control HTTP headers 2190 the site instructs to accompany the resulting image.”) • <i>See also</i> [0064], [0090]-[0092], [0097], [0102], [0108], [0111], [0134], [0135], [0143], Figs. 7, 9, 10.
--	--

185. See my discussion for element [1.a] regarding the claimed “first request.” Samaniego also discloses that the claimed first request provides as inputs the “original media,” “content generation procedures,” and “proprietary URL tags” that specify “dynamic content processing” procedures—which as I discussed in Section VIII.C describes the claimed transformation operations.

(d) Element [1.c]

186. As set forth in the table below, Samaniego discloses this element:

[1.c] determining whether a previously-generated first intermediate media content is available for reuse, the previously-generated first intermediate media content having been created using the first original media content and the first set of content generation operations; and	• [0148]-[0149] (“Upon receipt of the page 301, the browser contacts the system 100 requesting media specified in the URL. The system parses the URL 2100 to determine the content generation procedure 2140 to execute, any corresponding input parameters to be used by such procedure, any dynamic content processing 2150 to be performed by dynamic media procedures, any user profile information 2160 to be used to modify the resulting image, and any cache control HTTP headers 2190 the site instructs to accompany the resulting image. The parser generates a unique primary lookup key 2110 for the specified resulting media. If the key corresponds to an existing generated media 2180, such media is returned immediately to the browser 120 through a media cache 2120, and the transaction is complete. Otherwise, a media generation occurs. In the case of media generation requiring dynamic content processing, a unique secondary lookup key corresponding to intermediate media is generated 2130. If intermediate media 2170 corresponding to this key is found, such media is passed directly to the dynamic media content system 2150 having dynamic media procedures, wherein appropriate action is taken to generate the required derivative from the intermediate media data. A unique key is generated for the derivative 2130 and passed to the media caching system 2120. If the media caching system finds no such intermediate image, such intermediate image is generated according to instructions specified by the content generation procedure, cached by the media cache system 2120 as a secondary cached media 2170, and passed to the dynamic media system 2150.”) • <i>See also</i> [0064], [0065], [0104], [0108], [0111], [0140], [0143]-[0146], [0152].
--	--

187. See my discussion for element [1.b] regarding the “first original media content,” and “first content generation operations” specified in the request. Samaniego discloses that upon receiving the request, the server parses the URL identifying requested content for the required content generation procedures and transformation procedures required to generate the requested content, and searches for cached versions of the requested content. Samaniego, [0148]-[0149]. If the version of the content requested by the user is not already cached in the “primary cache,” the server then **uses a “secondary lookup key” to determine** whether there is a version of the **original media content** available in a “secondary cache” that was **created using the first set of content generation operations**. *Id.* If that version does not exist, the secondary lookup causes the system to generate and then store in secondary cache a copy of this intermediate version for use in future requests. *Id.* This check in the secondary cache with the secondary lookup key performs the claimed determination of whether a previously-generated first intermedia media content is available for reuse as claimed.

(e) Element [1.d]

188. As set forth in the table below, Samaniego discloses this element:

[1.d] responsive to determining that a previously-generated first intermediate	[0149] (“If intermediate media 2170 corresponding to this key is found, such media is passed directly to the dynamic media content system 2150 having dynamic media procedures, wherein appropriate
---	--

media content is available, creating a first optimized media content for the first playback device by performing the first set of transformation operations on the previously-generated first intermediate media content; and	action is taken to generate the required derivative from the intermediate media data.”) • <i>See also</i> [0065], [0103], [0105], [0108], [0111], [0112]-[0114], [0125], [0136], [0138], [0143], [0145], [0147], [0148], [0150]-[0152], Figs. 6, 7, 9, 20.
---	--

189. *See* my discussions with respect to element [1.c] regarding the “determining” step, element [1.a] regarding the “first playback device,” and element [1.b] regarding the “first transformation operations.” Samaniego discloses that when an intermedia content associated with the secondary key discussed with respect to element [1.c] is found in the secondary cache, that cached media is passed to the “dynamic media content system 2150” to perform the identified first set of transformation operations, as I discussed with respect to element [1.b]. Samaniego, [0149].

(f) Element [1.e]

190. As set forth in the table below, Samaniego discloses this element:

[1.e] responsive to determining that a previously-	• [0097] (“Web images 420 are Web versions of the original images. Web images 420 are ready for retrieval by the Web server 110 to be delivered to a
--	---

generated first intermediate media content is not available, creating a first optimized media content for the first playback device by creating a first intermediate content using the first original media content and the first set of content generation operations, and performing the first set of transformation operations on the first intermediate media content; and	Web browser 120. Any image to be modified or updated must pass through the herein above three components before it can be delivered to the Web browser 120. HTML pages 460 have references to Web images 420.”) • [0149] (“If the media caching system finds no such intermediate image, such intermediate image is generated according to instructions specified by the content generation procedure, cached by the media cache system 2120 as a secondary cached media 2170, and passed to the dynamic media system 2150. Again, appropriate action is taken to generate the required derivative from the intermediate image data.”) • <i>See also</i> [0064]-[0065], [0105], [0108], [0111], [0112]-[0114], [0125], [0136], [0138], [0140], [0143], [0145], [0147], [0148], [0150]-[0151], Figs. 6, 7, 9, 20.
--	---

191. See my discussion of element [1.c] regarding the “determining” step, element [1.a] regarding the “first playback device,” and [1.b] regarding the “first” “original media content,” “content generation operations,” and “transformation operations.” When the secondary cache is checked as discussed in element [1.c], if there is no intermedia media content available in the secondary cache, then the intermedia media is generated according to the content generation operations and cached so that it is available in response to later requests, and then further processed

in the dynamic media system 2150 to generate the requested version of content.

Samaniego, [0149].

(g) Element [1.f]

192. As set forth in the table below, Samaniego discloses this element:

[1.f] sending the first optimized media content to the first playback device.	• [0151] (“The resulting image is handed to the media cache system 2120 for caching and returned to the browser 120.”) • <i>See also</i> [0108], [0111], [0138], [0147], [0149], Figs. 7, 10, 18.
--	--

193. *See* my discussion of elements [1.d] and [1.e] regarding the “first optimized media content” and element [1.a] regarding the “first playback device.” Samaniego discloses that once processing is complete, the “resulting image” is “returned to the browser 120” of the first playback device. Samaniego, [0151].

(h) Element [2.pre]

194. Element [2.pre] recites: “The method of claim 1, further comprising.”

195. Samaniego discloses the method of claim 1, as I discussed above in Sections IX.C.2(a)-(g).

(i) Element [2.a]

196. As set forth in the table below, Samaniego discloses this element:

[2.a] receiving a second request from a second playback device	• [0137] (“Multiple client browsers 120a-120d communicate with both the Web server 110 and the image server 100 via the Internet 130.”)
---	---

for media content; wherein the second request contains information, the information indicating a second original media content, second set of content generation operations, and second set of transformation operations;	• [0135] (“In addition, generated media is cached such that further requests for the same media require little overhead.”) • <i>See also</i> [0065], [0092], [0135], [0108] [0110], [0111], [0137]-[0138], [0140], [0143], Figs. 7, 9, 17-18, 20.
--	---

197. Samaniego discloses that the server is in communication with “[m]ultiple client browsers,” and that such browsers generate requests for media content. Samaniego, [0137]-[0138]. Samaniego discloses [2.a] as it discloses that “multiple client browsers” send “requests” for media content containing the claimed information, as I discussed in elements [1.a] and [1.b]. Each of these subsequent requests is a second request from a second client browser and is formatted with the same information discussed in elements [1.a] and [1.b].

(j) Element [2.b]

198. **Element [2.b]** recites “**wherein the second request is received substantially concurrently with receiving the first request.**” As I discussed with respect to element [2.a] and Figure 1 as annotated for element [1.pre], Samaniego discloses that multiple client browsers are connected to, and communicate with, the

server. Samaniego, [0137]-[0138]. These client browsers each communicate over the Internet, and issues requests for media content. *Id.* When two users concurrently access the same page, two requests would be received substantially concurrently. A person having ordinary skill in the art would have understood this to not only render obvious, but to actually disclose that such concurrent requests would occur in a system such as the one I discussed above with respect to element [2.a] because it is commonplace for two users to concurrently navigate to the same webpage in the Internet. *See also* Samaniego, [0054], [0065], [0090], [0091], [0092], [0094], [0126], [0136]-[0138], [0140], [0143], [0146], [0149], Figs. 1, 17-19.

(k) Element [2.c]

199. As set forth in the table below, Samaniego discloses this element:

[2.c] wherein the first original media content and the second original media content are the same media content;	• [0135] (“In addition, the generated media is cached so that further requests for the same media require little overhead.”). • <i>See also</i> [0066], [0091]-[0092], [0099], [0136]-[0138], [0143], Fig. 6.
---	---

200. Samaniego discloses that when multiple requests for content are received at the server, the “further requests” may be “for the same media.” Samaniego, [0135].

(l) Element [2.d]

201. Element [2.d] recites “determining whether a second intermediate media content is available having been previously created using **the second original media content and the second set of content generation operations.**”

This element recites performing the same determination with respect to the “second original media content” and “second set of content generation operations” as the determination I addressed in element [1.c].

(m) Element [2.e]

202. Element [2.e] recites “responsive to determining that the second intermediate media content is available, **creating a second optimized media content for the first playback device by performing the second set of transformation operations on the second intermediate media content.**” This element recites performing the same steps responsive to a determination as I discussed with respect to element [1.d], only with respect to the “second,” rather than “first,” “optimized media content,” and “transformation operations,” and “intermediate media content.”

(n) Element [2.f]

203. Element [2.f] recites “responsive to determining that the second intermediate media content is not available, **creating a second optimized media content for the first playback device by creating the second intermediate content using the second original media content and the second set of content**

generation operations, and performing the second set of transformation operations on the second intermediate media content.” This element recites performing the same steps responsive to a determination as I discussed in element [1.e], only with respect to “second” content items and operation steps.

(o) Element [2.g]

204. Element [2.g] recites “sending the second optimized media content to the second playback device.” This element recites the same “sending” operation is performed on a “second” optimized media content as was recited, and I discussed, in element [1.f] with respect to a “first” optimized media content.

(p) Claims 3 and 4

205. As set forth in the table below, Samaniego discloses this element:

[3]/[4] The method of claim 2, wherein the first set of content generation operations and the second set of content generation operations are [the same/different], and the first set of transformation operations and the second set of transformation operations are different.	• [0114] (“Table B herein below provides the claimed script commands according to a preferred embodiment of the invention. Additional commands may be added as needed.”) • [0136] (“Because the invention takes the original media, content generation procedures, and proprietary URL tags as inputs for generating the Web media, it is possible to modify any of these inputs and have the system automatically update the media on the associated Web pages.”) • [0156] (“Table E below is a list of example supported content creation commands according to a preferred embodiment of the invention. Additional commands may be added as necessary.”) • See also Tables B, E, [0139], [0046]-[0050].
--	---

206. See my discussion of elements [1.b]-[1.c] discussing the “transformation” and “content generation” operations, as well as my discussion of claim 2. I previously discussed the difference between claims 3 and 4 as part of my discussion of Ground 1 in Section IX.A.3(p). Samaniego discloses that requests for content are handled using “**script commands**” (transformation operations) and “**content creation commands**” (content generation operations). A person having ordinary skill in the art would have understood that the same media content can be subjected to the same or different script commands and/or content creation commands based on different requests.

(q) Claim 5

207. As set forth in the table below, Samaniego discloses this element:

[5] The method of claim 1, further comprising: determining whether a previously-generated first optimized media content is available for reuse, wherein the previously-generated first optimized media content was created using the first original	• [0148]-[0149] (“Upon receipt of the page 301, the browser contacts the system 100 requesting media specified in the URL. The system parses the URL 2100 to determine the content generation procedure 2140 to execute, any corresponding input parameters to be used by such procedure, any dynamic content processing 2150 to be performed by dynamic media procedures, any user profile information 2160 to be used to modify the resulting image, and any cache control HTTP headers 2190 the site instructs to accompany the resulting image. The parser generates a unique primary lookup key 2110 for the specified resulting media. If the key corresponds to an existing generated media 2180, such media is returned immediately to the browser
---	---

media content, the first content generation operations, and the first transformation operations; responsive to determining that the previously-generated first optimized media content is available, sending the previously-generated first optimized media content to the first playback device.	120 through a media cache 2120, and the transaction is complete.”) • <i>See also</i> [0064], [0065], [0104], [0108], [0111], [0138], [0152], Figs. 7, 10, 22.
--	---

208. See my discussion of claim 1 above. As I explained with respect to element [1.c], upon receiving a request from a client browser, the system performs a “primary” lookup for the specified media content by determining content generation and transformation (for example, “dynamic content processing”) operations and then generating a “primary lookup key.” Samaniego, [0148]-[0149]. The system then determines whether the requested content, including required processing, is available using this “primary lookup key” and, if it is, “such media is returned immediately to the browser 120 through a media cache 2120, and the

transaction is complete.” *Id.* This successful lookup of content in a primary cache discloses claim 5.

(r) Element [6.a]

209. As set forth in the table below, Samaniego discloses this element:

[6.a] The method of claim 1, further comprising: determining whether the host computer has sufficient processing resources to create the first optimized media content for the first playback device;	• [0135] (“The system takes as input the client connection, server traffic, content generation procedures, and proprietary tags placed within the URL to generate optimized media for the client.”) • Table C (“Optimizes Web server traffic by adjusting the bandwidth of graphics; Optimizes images for client connection speed; Allows clients to specify the quality of images on a Web site”) • <i>See also</i> [0044], [0051], [0065], [0091], [0102], [0108], [0111], [0134]-[0135], [0140], [0143], [0148]-[0149], [0152], Figs. 6, 7, 10, 20, 22, Tables E-F.
---	---

210. Samaniego discloses that when the system determines how to process content, it takes as an input the “client connection” and “server traffic”—which a person having ordinary skill in the art would have understood describes using these processing resources needed for creating the first optimized media content as an input when generating the content. Samaniego, [0135]. By teaching the use of “server traffic” and adjusting “bandwidth” of graphics while also “[o]ptimiz[ing] images for client connection speed,” Samaniego teaches that the system uses both client and server-side processing and communication capabilities to determine how

to transcode content. Based on these teachings, Samaniego not only renders obvious, but actually discloses that the system uses these processing resource inputs to determine whether there are sufficient processing resources to generate requested content. *See id.*

(s) Element [6.b]

211. Element [6.b] recites “responsive to determining that the host computer does not have sufficient processing resources to create the first optimized media content, **determining an alternate first set of content generation operations or an alternate first set of transformation operations.**”

212. See my discussion of determining sufficient processing resources in with respect to element [6.a]. As I discussed above with respect to elements [1.b], [1.c], and [1.d], this information is used when determining the generation operations and transformation operations to be performed. As I explained for element [6.a], Samaniego’s teachings that the system will “optimize” content when taking in these inputs describing processing capacity not only renders obvious but actually discloses that the content is “optimized” by, for example, **adjusting the bandwidth of graphics.** *See* Samaniego, Table C.

(t) Element [6.c]

213. Element [6.c] recites “**creating an alternate first optimized media content using the** alternate first set of content generation operations or the alternate first set of transformation operations.”

214. See my discussion above of elements [6.a] and [6.b]. As discussed for elements [1.b]-[1.d], once **transformation and content generation steps** have been identified, the system, Samaniego not only renders obvious but actually discloses **performing those operations**.

(u) Claim 7

215. As set forth in the table below, Samaniego discloses this claim:

[7] The method of claim 6, wherein the first optimized media content is at a first level of quality and the alternate first optimized media content is at a second level of quality, wherein the first level of quality is higher than the second level of quality; wherein a level of quality of a media content is measured based on a compression format, a bit rate, and an image resolution of a media content sent to a playback device.	• Table B (“Convert(rtype @ <bit-depth> {dither @ <value 0..10>}) This command converts the image to the specified type/bit-depth. ... Save Save([type @ <image-type>]) This command saves a media to the specified file. Set Resolution SetResolution([dpi @ <value>] [xdpi @ <value>] [ydpi @ <value>]) This command changes the DPI of the image in memory.”) • Table C (“Reads and writes various file formats: BMP, GIF, JPG, PNG, TIF, PICT, TGA, PSD, FPX; Optimizes Web server traffic by adjusting the bandwidth of graphics; Optimizes images for client connection speed; Allows clients to specify the quality of images on a Web site”) • <i>See also</i> [0005], [0032], Tables E-F (similar to Tables B-C).
--	---

216. See my discussion of claim 6 and element [1.f] above. Samaniego discloses several commands used to optimize images to be sent to a client as discussed for claim 6 above. Images are saved to a file of file types such as “jpg” (a well-known lossy compression format) or TIF (a well-known lossless format)—disclosing a compression format. Samaniego, Tables B-C; *see also, e.g.*, Ex. 1033 (Gerald Graef, *Graphics Formats for Linux*, 23 LINUX J., March 1996), 2-4 (summarizing certain details of different image formats, explaining that “JPEG is a standard for lossy compression of images” and “LZW and LZ77 are the best lossless schemes...both GIF and TIFF, two of the three most common formats, use it.”). The “Convert” command is used to specify a “bit-depth” for the image—disclosing a bit rate for the content. Samaniego, Table B. And the image resolution is set using a “Set Resolution” command—disclosing an image resolution of a media content. *Id.* A person having ordinary skill in the art would have understood that these different operations impact image quality: lossy compression algorithms are lower quality than lossless; lower bitrate content is lower quality than higher bitrate content; and smaller resolution images are lower quality than high resolution images. *See* Samaniego, Tables B-C. Based on these teachings, Samaniego not only renders obvious but actually discloses generating these second quality level (lower quality) graphics using these parameters because lower quality images generally take fewer resources to process.

D. Ground 5: Samaniego in View of Tso Renders Obvious Claims 2-4

217. While, as the Examiner found, Samaniego discloses each of the claims including element [2.b]’s recitation of “concurrent[]” user requests, as I discussed above with respect to Ground 1 in Section IX.A.1, Tso also teaches “concurrently” handling two client requests for the same content as recited for claim 2. A person having ordinary skill in the art would have been motivated to apply these teachings to Samaniego.

218. Like Tso (as I discussed above in Section IX.A.1), Samaniego is also in the same field as the ’745 patent, networked media delivery, and indeed the ’745 patent claims priority to, and has the same specification as, Samaniego. Samaniego, [0002], [0064], [0090]; *see also* ’745 patent, Cover.

219. While Samaniego teaches that it is advantageous to handle “identical requests ... without regeneration of images,” Tso teaches how to avoid generating two copies of an image for concurrent requests by using multiple-thread processing. Samaniego, [0054]; Tso, 6:45-50. A person having ordinary skill in the art would have been motivated to modify Samaniego with Tso’s teachings of multiple-thread processing. As was well-understood with such processing, using multiple-thread processing allows a system to achieve the beneficial and predictable result of serving two concurrent requests for the same media content in parallel using different processing threads. Similarly, while Samaniego teaches optimizing web server

traffic by adjusting the bandwidth of graphics, Tso teaches how to perform such optimizations by tracking processing load on a server and reducing the quality of content served to, for example, non-VIP clients. Samaniego, [0115], Table C; Tso, 6:39-54. A person having ordinary skill in the art would have been motivated to modify Samaniego with Tso's teachings to achieve the beneficial and predictable result of ensuring that priority customers are efficiently provided with high quality content.

220. **First**, Tso teaches handling concurrent requests for the same media—*e.g.*, before the output from the first request is cached—by using multiple-thread processing to concurrently serve the requests in parallel using separate threads, while Samaniego teaches that it is advantageous to handle multiple requests without regenerating the same content. Tso, 6:51-63; Samaniego, [0110]. Specifically, Samaniego discloses that whenever content is processed, it is cached and served in response to other requests to avoid unnecessary processing. Samaniego, [0110]. Tso addresses the case of those requests arriving concurrently, before the output from a first request is cached, by using multiple-thread processing to detect the concurrent requests and retrieve the object only “once.” Tso, 6:51-63. A person having ordinary skill in the art would have been motivated to modify Samaniego with Tso's teachings to reduce unnecessary processing that otherwise occurs if the same content is processed twice in the same way, as would happen without Tso's multiple-thread

processing teachings when the output of the first request is not yet cached when the second request is received.

221. **Second**, Tso teaches improving the efficiency and speed of the system by using multi-threaded processing to eliminate the need to “wait[] for a hypertext object” to be added to cache before reusing the object because the second request is handled using the content already in the process of being transcoded. Tso, 6:51-63. This improves speed in two ways: first, with respect to second requests, the request does not need to wait for a new transcoding operation begins for the media to be prepared in response to the request, and instead the second request is handled when the transcoding process in response to the first request is complete. Second, because the content is being transcoded only a single time for both requests, this frees up transcoding processing resources, allowing the system to handle other transcoding requests more quickly. *See id.* Samaniego recognizes these speed benefits, and discloses that it is advantageous to improve the “speed” to “add images, video, and sound to sites.” Samaniego, [0045]. A person having ordinary skill in the art would have been motivated to modify Samaniego with Tso’s teachings to improve the efficiency and speed of the system.

222. **Third**, Tso’s teaching of managing processing load complements Samaniego’s teaching of adjusting the quality of images based on client “connection speed” and “server traffic.” Samaniego, [0115], Table C. Tso discloses using the

“condition of the device” to adjust content processing—which expands on Samaniego’s teaching by identifying similar parameters to “server traffic,” including “current processing load” at a transcoding server. Tso, 6:64-7:14. A person having ordinary skill in the art would have found it obvious to further consider “current processing load” at a server when determining how to adjust the bandwidth of images to beneficially maximize the use of server processing resources.

223. In light of these teachings, a person having ordinary skill in the art would have had a reasonable expectation of success in modifying Samaniego with Tso’s teachings of multi-threaded processing. Indeed, Tso teaches that its disclosed multi-threaded access to cached objects uses a “standard” interface and does not “require any particular configuration” for the cache, while Samaniego teaches using a cache to avoid regenerating content for subsequent requests. Tso, 4:48-5:7, 5:2-7; Samaniego, [0135], [0138]. A person having ordinary skill in the art therefore would have found it straightforward and obvious to modify Samaniego with Tso’s teachings, yielding a system in which Samaniego’s teachings are used to concurrently serve content to multiple users using multi-threaded processing. A person having ordinary skill in the art likewise would have had a reasonable expectation of success because Tso merely discloses using factors to consider when adjusting the bandwidth of images that are similar to those already disclosed in Samaniego.

E. Ground 6: Samaniego in View of Lawler Renders Obvious Claims 6-7

224. While Samaniego already teaches optimizing content by using alternative processing instructions when the host computer does not have sufficient processing resources, and Samaniego also teaches measuring quality using bit rate or resolution, a person having ordinary skill in the art would have been motivated to apply Lawler's teachings of selecting alternative, lower quality media formats to Samaniego. As discussed above in Section IX.B.1, Lawler, like Samaniego is in the same field as the '745.

225. Samaniego discloses using server load—which a person having ordinary skill in the art would have understood includes processing capacity—to determine the content generation and transformation operations. Samaniego, [0091], [0135]. Lawler provides supplemental teachings of selecting alternative, lower quality content, including lower in quality based on bit rate or resolution, depending on CPU processing capacity to avoid interrupting delivery of content to clients. *E.g.* Lawler, 2:47-61, 5:57-6:18, 9:25-35. A person having ordinary skill in the art thus would have been motivated to modify Samaniego with Lawler's teachings by using an alternative, lower quality format for content based on processing capacity for at least the following independent reasons.

226. **First**, both Samaniego and Lawler describe systems for transmitting content to multiple clients. *E.g.*, Samaniego, [0094]; Lawler, 5:40-56, 6:13-16.

Lawler discloses that such systems suffer from capacity limitations when serving multiple clients in such systems because “deliver[ing] programming to multiple subscribers or viewers” can result in “[c]apacity limitations” when “IT system 10 lacks the resources required to provide the service in the requested format.” Lawler, 5:40-56; 6:13-16. A person having ordinary skill in the art would have been motivated to modify Samaniego with Lawler’s teachings of using “alternative” formats to advantageously avoid interrupting content delay to clients when serving multiple clients concurrently. Lawler, 6:14-16, 2:53-55.

227. **Second**, Samaniego discloses that it would be advantageous to service requests with “little overhead.” Samaniego, [0135]. A person having ordinary skill in the art would have recognized that Lawler’s teaching of using alternative requirements advantageously reduces overhead as described in Samaniego by using “less resources” for alternative content when additional processing capacity is unavailable. Lawler, 7:18-43; *see also id.* 5:40-56; 6:13-16.

228. **Third**, Lawler provides complementary teachings of measuring quality. Samaniego discloses that its system allows clients to “specify the quality” of content, and that it would be advantageous to allow a system to specify an “optimum quality” for content, and describes a system that allows “image quality and compression to be set.” Samaniego, [0052], [0032], [0045], Tables C, F. A person having ordinary skill in the art would have found it obvious to further modify

Samaniego's teachings of quality based on compression with Lawler's teachings of measuring quality using bit rate and resolution, yielding a system that is able to measure quality using bit rate, resolution, and compression. Lawler, 6:7-12, 2:53-61, 6:37-50.

229. In light of the above teachings, a person having ordinary skill in the art would have had a reasonable expectation of success in applying Lawler's teachings. Indeed, Samaniego already teaches that "system features" include optimizing web server traffic by "adjusting the bandwidth of graphics." Samaniego, Tables C, F. Modifying Samaniego with Lawler's teachings would have been a straightforward application of those teachings in a manner that advantageously improves the efficiency of the system in servicing many user requests. For example, applying Lawler's teachings, as part of the process of optimizing web content, the system will determine that it does not have processing capacity to generate a preferred format of content and, as a result, use an alternative format of content instead in order to optimize the system to quickly provide content back to the user.

230. With respect to [6.a], as discussed within Section IX.C.2 regarding element [6.a], Samaniego discloses that the system uses "server traffic" as an input when generating optimized media. Samaniego, [0051], [0065], [0091], [0135], Tables C, F. Lawler further discloses determining whether the host computer has sufficient processing resources for the first optimized media content for the first

playback device (*see* Section IX.B.3) by, e.g., determining whether the system “has available the resources,” including “capacity” in “CPU 66,” to satisfy a “ProcessorRequirement” for generating content. Lawler, Fig. 4, 9:25-35. A person having ordinary skill in the art would have been motivated to modify Samaniego’s teaching of using “server traffic” as an input with this teaching from Lawler, yielding a system in which Samaniego’s system uses a determination of whether the server has sufficient processing resources as an input when generating optimized media for content—rendering this element obvious.

231. With respect to [6.b], as discussed within Section IX.C.2, for elements [1.b]-[1.d] and [6.b], Samaniego discloses that the server determines content generation and transformation operations based on system inputs. Samaniego, [0066], [0094], [0097], [0148]-[0149]. Lawler further discloses responsive to determining that the host computer does not have sufficient processing resources, determining an alternate media content (*see* Section IX.B.3) by, e.g., “identif[y]ing alternative resources” for delivering media content “in an alternative format that requires fewer IT system resources.” Lawler, 2:53-61. A person having ordinary skill in the art would have been motivated to modify Samaniego with this teaching, yielding a system in which, when in [6.a] the system uses processing resources as an input and determines that it does not have enough resources, the system responds by

using alternative content generation or transformation operations for which it does have available processing resources—disclosing this element.

232. With respect to [6.c], as discussed within Section IX.C.2 regarding elements [6.c] and [1.b]-[1.d], Samaniego discloses creating optimized media content using content generation operations and transformation operations. Samaniego, [0066], [0148]-[0149]. A person having ordinary skill in the art would have been motivated to modify Samaniego with Lawler’s teachings as discussed above for [6.a]-[6.b] to generate alternative content generation and transformation operations; the resulting system generates media using the alternative, rather than preferred, content generation or transformation operations—disclosing this element.

233. With respect to [7.a], as discussed above for claim 6, Samaniego in view of Lawler renders obvious generating the first optimized media. Lawler further discloses the first optimized media content is at a first level of quality, and an alternate first optimized media content is at a second level of quality, wherein the first level of quality is higher than the second level of quality (*see* Section IX.B.3) by using, e.g., an alternative media format that “requires fewer IT system resources.” Lawler, 2:53-61. A person having ordinary skill in the art would have found it obvious to modify Samaniego with this teaching to generate media of an alternative format that uses fewer processing resources when the determination in [6.a] indicates there are insufficient processing resources; this alternative format is of a lower

quality because “higher resolution or quality” is generated “at the expense of greater IT system resources requirements.” Lawler, 2:53-61, 5:40-6:12.

234. With respect to [7.b], as disclosed above in Section IX.C.2 with respect to element [7.b], Samaniego discloses wherein a level of quality of a media content is measured based on a compression format (Samaniego, [0032]). Lawler further discloses wherein a level of quality of a media content is measured based on a bit rate, and an image resolution of a media content sent to a playback device, *see* Section IX.B.3. A person having ordinary skill in the art would have been motivated to modify Samaniego by applying Lawler’s teachings to measure quality using bit rate and image resolution by using these properties in addition to compression format to measure quality because each of these features separately affects the processing required to generate the alternative format of content—and therefore the quality of content that can be generated. Lawler, 2:53-61, 4:59-65, 5:40-6:12.

X. SECONDARY CONSIDERATIONS

235. I have not seen any evidence in the prosecution history of the ’745 patent or any related application that I considered of any arguments regarding secondary considerations exist. Nor have I seen any evidence that any such evidence would have a nexus to any of the Challenged Claims 1-7. Moreover, as shown above with respect to the prior art referenced in Section IX, any alleged solutions to problems or unexpected results in the ’745 patent were already well known.

XI. CONCLUSION

236. In summary, I have concluded that the Challenged Claims 1-7 are anticipated by and rendered obvious over the prior art references as discussed above.

237. Specifically, in my opinion, for at least the reasons above, the Challenged Claims 1-5 are rendered obvious over Tso in view of Huang, claims 6-7 are rendered obvious over Tso in view of Huang and Lawler, claims 1-7 are anticipated by and obvious over Samaniego, claims 2-4 are rendered obvious by Samaniego in view of Tso, and claims 6-7 are rendered obvious by Samaniego in view of Lawler.

238. To the extent it is argued that any further disclosure is required for an element in the Challenged Claims 1-7 that I have identified as being disclosed, a person having ordinary skill in the art would certainly have found that element obvious to include based on the same explicit disclosure or inherent disclosures and analysis I have identified above.

239. I reserve the right to supplement my opinions in the future to respond to any arguments that Patent Owner or its expert(s) may raise and to take into account new information as it becomes available to me.

240. 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 the case and that cross-examination will take place within the United States. If cross-examination is required of me, I will appear for cross-examination within the United States during the time allotted for cross-examination.

241. I hereby declare that all statements made herein of my own knowledge are true and that all statements 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 Section 1001 of Title 18 of the United States Code.

Executed this 10th day of January 2023, in Atlanta, Georgia

Respectfully submitted,



Prof. Vijay K. Madisetti, Ph.D.

Appendix A

Dr. Vijay K. Madiseti

Fellow, IEEE

vkm@madiseti.com

Cell: 770-527-0177

Address:

**56 Creekside Park Drive
Johns Creek, GA 30022**

Employment:

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

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

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 (2016-2022) 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-)

Causam v. Ecobee, Itron, et al

Case No. ITC 337-TA-1277

Expert for Causum

Technology: Load Management

Submitted Expert Reports and Deposition (6/22)

Masimo v. Apple

Case No. ITC 337-TA-1276 and related cases

Expert for Masimo

Technology: Pulse Oximetry

Submitted Expert Reports, Deposition & Trial

TQ Delta v. Nokia, Adtran, CommScope, 2Wire, et al

Cases in Delaware & Texas

Expert for TQ Delta

Technology: DSL and Cable Modems

Submitted Expert Reports, Depositions & Hearings.

Paltalk Holdings v. Cisco Systems Inc.

Western District of Texas (6:21-CV-00757-ADA)

Expert for Paltalk Holdings

Technology: VOIP Technologies

Submitted Expert Reports and Deposition

Ollnova Technologies v. Carrier Group

Southern District of Florida (Case 9:22-cv-80388-DMM)

Expert for Ollnova

Technology: IoT

Submitted Report and Deposition.

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, LoganTree, Unified Patents, BMC Software, Masimo, Sentius, BMW, Facebook, PAWS, Google., 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., Arpikanondt, A.

Springer, Boston: MA, Springer, 2004, 280 pp.

6. The Digital Signal Processing Handbook – Second Edition.

Madiseti, V. K. (2009)

CRC Press, Boca Raton, Fla.

7. Cloud Computing: A Hands-On Approach

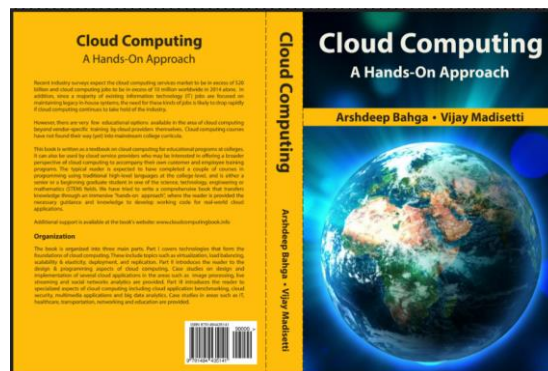
A Bahga, V. Madiseti (2013)

Amazon CreateSpace Publishing, 2013, 454 pp.

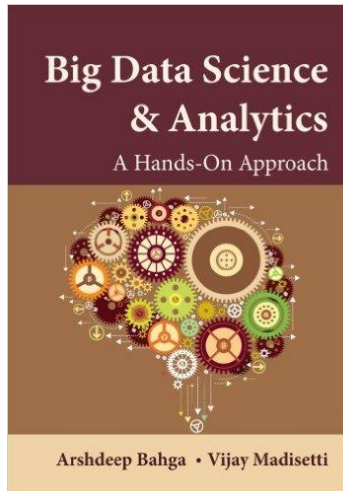
8. Internet of Things: A Hands-On Approach

A Bahga, V. Madiseti (2014)

Amazon CreateSpace Publishing, 2014, 450 pp.



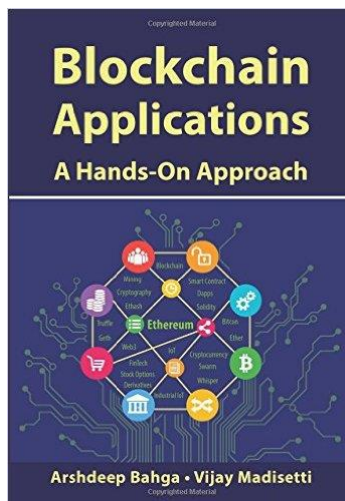
9. Big Data Science & Analytics: A Hands-On Approach
A Bahga, V. Madiseti (2016)
 Amazon CreateSpace Publishing, 2016, 542 pp.



10. Blockchain Applications: A Hands-On Approach

A Bahga, V. Madiseti (2017)

Amazon CreateSpace Publishing, 2017, 380 pp.



Edited Books & Collection of Papers

1. Advances in Parallel & Distributed Simulation

Madiseti, V.K.; Nicol, D., Fujimoto, R. (Editors)

San Diego, CA: SCS Press, 1991, 200 pp.

2. Modeling, Analysis, Simulation of Computer & Telecommunications Systems

Madiseti, V., Gelenbe, E., Walrand, J. W. (Editors)

Los Alamitos: CA, IEEE Computer Society Press, 1994, 425 pp.

3. Modeling & Simulations on Microcomputers

Madiseti, V.K. (Editor)

San Diego, CA: SCS Press, 138 pp. 1990.

Editorship of Journals & Transactions

1. IEEE Design & Test of Computers

Special Issue: Reengineering Digital Systems

April – June 1999 (Vol 16, No 2)

Madiseti, V.K (Editor)

Los Alamitos: CA, IEEE Computer Society Press, 1999.

2. IEEE Design & Test of Computers

Special Issue: Rapid Prototyping of Digital Systems

Fall 1996 (Vol 13, No 3)

Madiseti, V., Richards, M. (Editors)

Los Alamitos: CA, IEEE Computer Society Press, 1994, 425 pp.

3. IEEE Transactions on Circuits & Systems II

Associate Editor: 1993-1995.

4. International Journal in Computer Simulation

Associate Editor: 1990-1993

5. International Journal in VLSI Signal Processing

Editorial Board: 1995 - Present

Recent Issued US Patents

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

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

Refereed Journal Publications

1. Trends in the Electronic Control of Mine Hoists

*Madiseti, V. and Ramlu, M.,
IEEE Transactions on Industry Applications, Vol IA-22, No. 6,
November/December 1986. Pages 1105-1112*

2. Multilevel range/NEXT performance in digital subscriber loops

Brand, G.; Madiseti, V.; Messerschmitt, D.G.;
Communications, Speech and Vision, IEE Proceedings I [see also IEE

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

3. Seismic migration algorithms on parallel computers

Madiseti, V.K.; Messerschmitt, D.G.;
Signal Processing, IEEE Transactions on [see also Acoustics, Speech, and
Signal Processing, IEEE Transactions on] ,Volume: 39 , Issue: 7 , July 1991
Pages:1642 – 1654

4. Asynchronous algorithms for the parallel simulation of event-driven dynamical systems

Madiseti, V.K.; Walrand, J.C.; Messerschmitt, D.G.:
ACM Transactions on Modeling and Computer Simulation, v 1, n 3, July 1991,
Pages: 244-74

5. Synchronization mechanisms for distributed event-driven computation

Madiseti, V.K.; Hardaker, D.:
ACM Transactions on Modeling and Computer Simulation, v 2, n 1, Jan. 1992,
Pages: 12-51

6. Efficient VLSI Architectures for the Arithmetic Fourier Transform (AFT)

Kelley, B.T.; Madiseti, V.K.;
Signal Processing, IEEE Transactions on [see also Acoustics, Speech, and
Signal Processing, IEEE Transactions on] ,Volume: 41 , Issue: 1 , January
1993
Pages:365-378

7. The fast discrete Radon transform. I. Theory

Kelley, B.T.; Madiseti, V.K.;
Image Processing, IEEE Transactions on ,Volume: 2 , Issue: 3 , July 1993
Pages:382 – 400

8. The Georgia Tech digital signal multiprocessor

Barnwell, T.P., III; Madiseti, V.K.; McGrath, S.J.A.;
Signal Processing, IEEE Transactions on [see also Acoustics, Speech, and
Signal Processing, IEEE Transactions on] ,Volume: 41 , Issue: 7 , July 1993
Pages:2471 – 2487

9. The MIMDIX Environment for Parallel Simulation

Madiseti, V.K.; Hardaker, D.; Fujimoto, R.M.:
Journal of Parallel and Distributed Computing, v18, no. 4, August 1993,
Pages: 473-83.

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

18. Error correcting run-length limited codes for magnetic recording

Jaejin Lee; Madiseti, V.K.;

Magnetics, IEEE Transactions on ,Volume: 31 , Issue: 6 , Nov. 1995

Pages:3084 – 3086

19. Virtual prototyping of embedded microcontroller-based DSP systems

Madiseti, V.K.; Egolf, T.W.;

Micro, IEEE ,Volume: 15 , Issue: 5 , Oct. 1995

Pages:9 – 21

20. Constrained multitrack RLL codes for the storage channel

Lee, J.; Madiseti, V.K.;

Magnetics, IEEE Transactions on ,Volume: 31 , Issue: 3 , May 1995

Pages:2355 – 2364

21. Rapid digital system prototyping: current practice, future challenges

Madiseti, V.K.;

Design & Test of Computers, IEEE ,Volume: 13 , Issue: 3 , Fall 1996

Pages:12 – 22

22. Conceptual prototyping of scalable embedded DSP systems

Dung, L.-R.; Madiseti, V.K.;

Design & Test of Computers, IEEE ,Volume: 13 , Issue: 3 , Fall 1996

Pages:54 – 65

23. Advances in rapid prototyping of digital systems

Madiseti, V.K.; Richards, M.A.;

Design & Test of Computers, IEEE ,Volume: 13 , Issue: 3 , Fall 1996

Pages:9

24. Combined modulation and error correction codes for storage channels

Jaejin Lee; Madiseti, V.K.;

Magnetics, IEEE Transactions on ,Volume: 32 , Issue: 2 , March 1996

Pages:509 – 514

25. Model-based architectural design and verification of scalable embedded DSP systems-a RASSP approach

Dung, L.-R.; Madiseti, V.K.; Hines, J.W.;

Chapter in VLSI Signal Processing, IX, 1996.

Pages:147 – 156

26. Low-power digital filter implementations using ternary coefficients

Hezar, R.; Madiseti, V.K.;

Chapter in VLSI Signal Processing, IX, 1996.,

Pages:179 – 188

27. All-digital oversampled front-end sensors

Romdhane, M.S.B.; Madiseti, V.K.;

Signal Processing Letters, IEEE, Volume: 3 , Issue: 2 , Feb. 1996

Pages:38 – 39

28. Modeling COTS components in VHDL

Calhoun, S., Reese, R; Egolf, T., Madiseti, V.K.;

Journal of VLSI Signal Processing, Volume: 14 , Issue: 2 , Nov 1996

Pages: 24 – 31

29. VHDL-Based Rapid Systems Prototyping

Egolf, T.; Madiseti, V.K.;

Journal of VLSI Signal Processing, Volume: 14 , Issue: 2 , Nov 1996

Pages: 40-52

30. Interface design for core-based systems

Madiseti, V.K.; Lan Shen;

Design & Test of Computers, IEEE ,Volume: 14 , Issue: 4 , Oct.-Dec. 1997

Pages:42 - 51

31. Incorporating cost modeling in embedded-system design

Debardelaben, J.A.; Madiseti, V.K.; Gadiant, A.J.;

Design & Test of Computers, IEEE ,Volume: 14 , Issue: 3 , July-Sept. 1997

Pages:24 – 35

32. On homomorphic deconvolution of bandpass signals

Marenco, A.L.; Madiseti, V.K.;

Signal Processing, IEEE Transactions on [see also Acoustics, Speech, and

Signal Processing, IEEE Transactions on] ,Volume: 45 , Issue: 10 , Oct. 1997

Pages:2499 – 2514

33. A case study in the development of multi-media educational material: the VHDL interactive tutorial

Gadiant, A.J.; Stinson, J.A., Jr.; Taylor, T.C.; Aylor, J.H.; Klenke, R.H.;

Salinas, M.H.; Madiseti, V.K.; Egolf, T.; Famorzadeh, S.; Karns, L.N.; Carter,

H.W.;

Education, IEEE Transactions on ,Volume: 40 , Issue: 4 , Nov. 1997

Pages:17 pp.

34. Adaptive mobility management in wireless networks

Jeongwook Kim; Madiseti, 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.; Madiseti, 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; Madiseti, 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.; Madiseti, V.K.;

Design & Test of Computers, IEEE ,Volume: 16 , Issue: 2 , April-June 1999

Pages:48 – 56

38. Reengineering legacy embedded systems

Madiseti, 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

Madiseti, 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; Madiseti, 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.; Madiseti, 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.; Madiseti, V.K.;

Journal of Ad Hoc Networks, Elsevier Press, 2004

- 43. Closed-form analysis of CDMA systems using Nyquist pulse**
Jatunov, L.A.; Madiseti, V. K.;
Communications Letters, IEEE (Under Revision), 2005.
- 44. Systematic Design of End-to-End Wireless Mobility Management Protocols,**
Argyriou, A.; Madiseti, V. K.;
ACM/Springer Wireless Networks (WINET), Accepted 2005.
- 45. A Novel End-to-End Approach for Video Streaming Over the Internet,**
Argyriou, A.; Madiseti, 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.; Madiseti, V. K.;
IEEE Transactions on Multimedia, Submitted for review in March 2005.
- 47. Modeling the Effect of Handoffs on Transport Protocol Performance,**
Argyriou, A.; Madiseti, 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.; Madiseti, V. K.;
ACM Transactions on Multimedia Computing, Communications & Applications, Submitted for review in April 2005.
- 49. "Electronic System, Platform & Package Codesign",**
Madiseti, 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, Madiseti, V. K.
Wireless Networks, Springer, May 2006.
- 51. "A Soft-Handoff Transport Protocol for Media Flows in Heterogeneous Mobile Networks ",**
A. Argyriou, Madiseti, 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.; Madiseti, V.K.;
Euromicro Conference, 2001. Proceedings. 27th , 4-6 Sept. 2001
Pages:264 - 269

28. Adaptability, extensibility and flexibility in real-time operating systems

Kuacharoen, P.; Akgul, T.; Mooney, V.J.; Madiseti, V.K.;
Digital Systems, Design, 2001. Proceedings. Euromicro Symposium on , 4-6 Sept. 2001
Pages:400 - 405

29. Effect of handoff delay on the system performance of TDMA cellular systems

Turkboylari, M.; Madiseti, V.K.;
Mobile and Wireless Communications Network, 2002. 4th International Workshop on , 9-11 Sept. 2002
Pages:411 - 415

30. Enforcing interdependencies and executing transactions atomically over autonomous mobile data stores using SyD link technology

Prasad, S.K.; Bourgeois, A.G.; Dogdu, E.; Sunderraman, R.; Yi Pan; Navathe, S.; Madiseti, V.;
Distributed Computing Systems Workshops, 2003. Proceedings. 23rd International Conference on , 19-22 May 2003
Pages:803 - 809

31. Performance evaluation and optimization of SCTP in wireless ad-hoc networks

Argyriou, A.; Madiseti, V.;
Local Computer Networks, 2003. LCN '03. Proceedings. 28th Annual IEEE International Conference on , 20-24 Oct. 2003
Pages:317 - 318

32. Implementation of a calendar application based on SyD coordination links

Prasad, S.K.; Bourgeois, A.G.; Dogdu, E.; Sunderraman, R.; Yi Pan; Navathe, S.; Madiseti, V.;
Parallel and Distributed Processing Symposium, 2003. Proceedings. International , 22-26 April 2003
Pages:8 pp.

33. Bandwidth aggregation with SCTP

Argyriou, A.; Madiseti, V.;
Global Telecommunications Conference, 2003. GLOBECOM '03. IEEE Volume: 7 , 1-5 Dec. 2003
Pages:3716 - 3721 vol.7

34. Software streaming via block streaming

Kuacharoen, P.; Mooney, V.J.; Madiseti, 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.; Madiseti, 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.; Madiseti, 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.; Madiseti, 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.; Madiseti, V.;
Consumer Communications and Networking Conference, 2004. CCNC 2004. First IEEE , 5-8 Jan. 2004
Pages:169 – 174

Other Publications

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

Ph.D. Students Graduated

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

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

Winner of VHDL International Best PhD Thesis Award, 1997

7. Thomas W. Egolf, 1997

Virtual Prototyping of Embedded DSP Systems

Distinguished Member of Technical Staff, Agere

8. Alvaro Marenco, 1997

On Homomorphic Deconvolution of Bandpass Signals

Professor, Texas A&M University.

Winner of GIT ECE Outstanding Teaching Assistant Award

9. Shahram Famorzadeh, 1997

BEEHIVE: A Distributed Environment for Adaptive Signal Processing

Member of Technical Staff, Rockwell.

10. Timothy J. Klausutis, 1997

Adaptive Lapped Transforms with Applications to Image Coding.

US Air Force/Univ. of Florida.

11. Lan Shen, 1998

Temporal Design of Core-Based Systems

Member of Technical Staff, IBM

12. James DeBardelaben, 1998

Optimization Based Approach to Cost Effective DSP Design

Research Scientist, Johns Hopkins University

Georgia Tech ECE Faculty Award

13. Sangyoun Lee, 1999

Design of Robust Video Signal Processors

Professor, Yonsei University

US Army Sensors Lab Research Excellence Award, 1999

14. Rahmi Hezar, 2000

Oversampled Digital Filters

Member of Technical Staff, Texas Instruments

15. Yong-kyu Jung, 2001

Model-Based Processor Synthesis

Professor, Texas A&M University

16. Mustafa Turkboylari, 2002

Handoff Algorithms for Wireless Applications

Member of Technical Staff, Texas Instruments

17. Yun-Hui Fan, 2002

A Stereo Audio Coder with Nearly Constant Signal to Noise Ratio

Post-Doctoral Research Associate, Northeastern University

18. Subrato K. De, 2002

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

US Army Sensors Lab Research Excellence Award, 1999

19. Chonlameth Aripnikanondt, 2004

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

US Army Sensors Lab Research Excellence Award, 1999

20. Loran Jatunov, 2004

Performance Analysis of 3G CDMA Systems
Senior Research Scientist, Soft Networks, LLC.

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

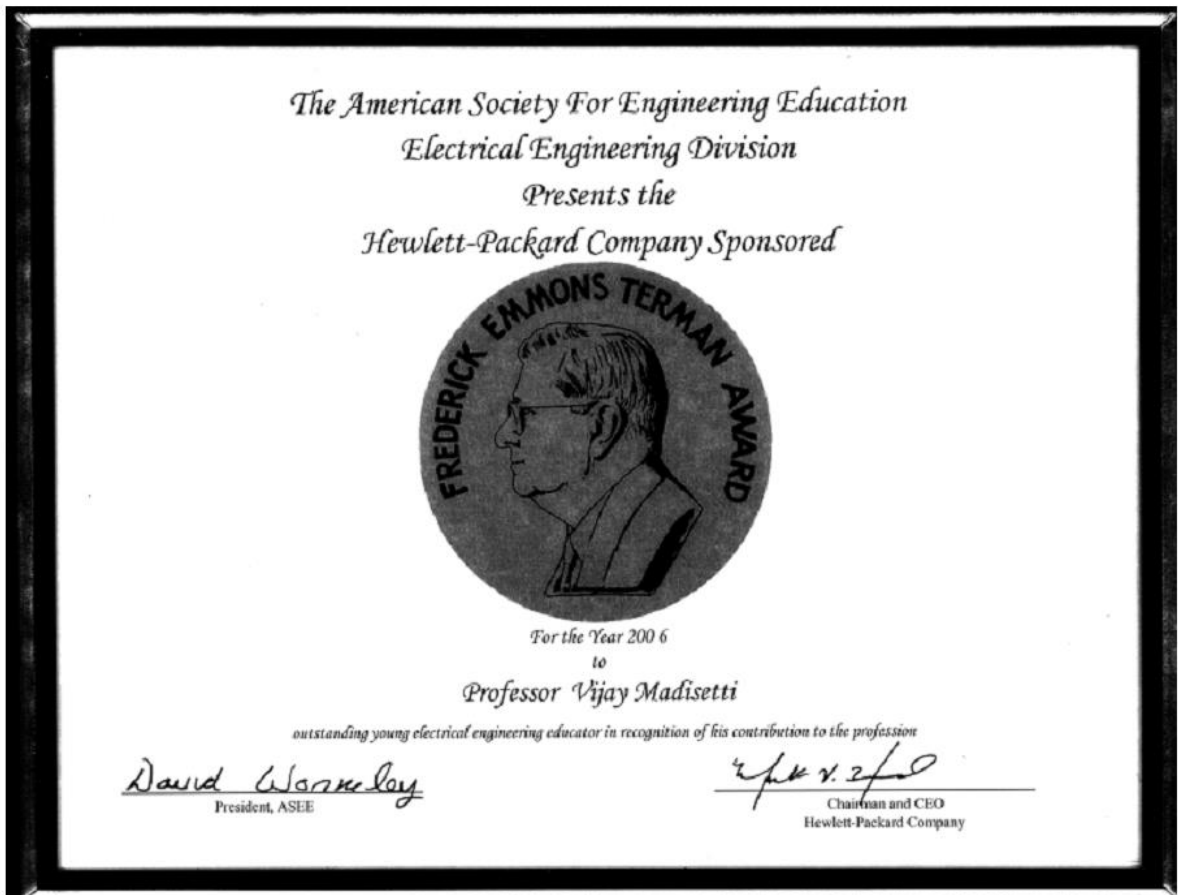
22. Pilho Kim, 2009, Scientist, VP Technologies, Inc.

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

Awards & Honors

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

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



Intellectual Property Disclosures (Georgia Tech)

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

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

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

Appendix B

APPENDIX B: LIST OF MATERIALS CONSIDERED

Ex. 1001	U.S. Patent No. 9,158,745 (“’745”)
Ex. 1002	File History of U.S. Application No. 13/752,110 (“’745 FH)
Ex. 1004	U.S. Patent No. 6,421,733 (“Tso”)
Ex. 1005	U.S. Patent No. 6,438,576 (“Huang”)
Ex. 1006	U.S. Patent No. 5,905,522 (“Lawler”)
Ex. 1007	U.S. Patent Application Publication No. US 2002/0078093 (“Samaniego”)
Ex. 1008	U.S. Patent No. 5,420,967 (“Delp”)
Ex. 1009	U.S. Patent No. 5,969,716 (“Davis”)
Ex. 1013	Redline Comparison of specifications from Samaniego and the ’745 patent
Ex. 1033	Gerald Graef, <i>Graphics Formats for Linux</i> , 23 LINUX J., March 1996, https://dl.acm.org/doi/fullHtml/10.5555/325530.325533
Ex. 1036	Luca Delgrossi et al., <i>Media Scaling in a Multimedia Communication System</i> , 2 MULTIMEDIA SYS. 172 (1994), https://link.springer.com/article/10.1007/BF01210448
Ex. 1037	Armando Fox and Eric A. Brewer, <i>Reducing WWW latency and bandwidth requirements by real-time distillation</i> , COMPUT. NETWORKS & ISDN SYS., PROCS. 5TH INT’L WORLD WIDE WEB CONF. (May 1996), https://www.sciencedirect.com/science/article/abs/pii/016975529600027X
Ex. 1038	U.S. Patent No. 6,144,996 (“Starnes”)