# EXHIBIT 1003
# DECLARATION OF
# DR. MICHAEL IAN SHAMOS

**IN THE UNITED STATES PATENT AND TRADEMARK OFFICE**

| | | |
|---|---|---|
| *In re* patent of Tuttle et al. | § | Attorney Docket No.: IVI766 |
| | § | |
| U.S. Patent 8,407,722 | § | |
| | § | |
| Issue Date: March 26, 2013 | § | Customer No.: 165774 |
| | § | |
| Filing Date: March 30, 2006 | § | |
| | § | |
| For: Asynchronous Messaging Using a Node Specialization Architecture In the Dynamic Routing Network | § | |

**DECLARATION OF DR. MICHAEL IAN SHAMOS**

## Table of Contents

## I. BACKGROUND AND QUALIFICATIONS

1. My name is Michael Shamos, and I am over 21 years and otherwise competent to make this Declaration. I make this Declaration based on facts and matters within my own knowledge and on information provided to me by others, and, if called as a witness, I could and would competently testify to the matters set forth herein.

2. I have been retained as a technical expert witness in this matter by counsel for the Requester, Unified Patents, LLC ("Requester") to provide my independent opinions on certain issues requested by counsel for Requester relating to the accompanying Request for Reexamination of U.S. Patent No. 8,407,722 ("the '722 Patent"), challenging Claims 1, 6, 7, 14-17, 20-23, 26-29, and 32-35 (the "Challenged Claims"). My compensation of $650 per hour in this matter is not based on the substance of my opinions or on the outcome of this matter. I have been informed that Intellectual Ventures I LLC ("Patent Owner") is the purported owner of the '722 Patent. I note that I have no financial interest in Requester, Patent Owner, or the '722 Patent, and I have no other interest in the outcome of this matter.

3. I hold the title of Distinguished Career Professor in the School of Computer Science at Carnegie Mellon University in Pittsburgh, Pennsylvania. I am a member of two departments in that School, the Institute for Software Research and Societal Systems Department and the Language Technologies Institute. I was a founder and Co-Director of the Institute for eCommerce at Carnegie Mellon from 1998-2004 and from 2004-2018 was Director of the eBusiness Technology graduate program in the Carnegie Mellon University School of Computer Science. I am now the Director of the M.S. in Artificial Intelligence and Innovation program at Carnegie Mellon.

4. I received an A.B. (1968) from Princeton University in Physics; an M.A. (1970) from Vassar College in Physics; an M.S. (1972) from American University in Technology of Management, a field that covers quantitative tools used in managing organizations, such as statistics, operations research and cost-benefit analysis; an M.S. (1973), an M.Phil. (1974) and a Ph.D. (1978) from Yale University in Computer Science; and a J.D. (1981) from Duquesne University.

5. I have taught graduate courses at Carnegie Mellon in Electronic Commerce, including eCommerce Technology, Electronic Payment Systems, Electronic Voting and eCommerce Law and Regulation, as well as Analysis of Algorithms. Since 2007, I have taught an

annual course in Law of Computer Technology. I currently also teach Artificial Intelligence and Future Markets.

6. I am also Visiting Professor at the University of Hong Kong. Since 2001, I have taught an annual graduate course there entitled Electronic Payment Systems. This is one of only a handful of graduate courses in the world on the subject of electronic payments.

7. I am the author and lecturer in a 24-hour video course on Internet protocols and have taught computer networking, wireless communication and Internet architecture since 1999. I also delivered a course for McKinsey & Company consultants on Mobile Communication.

8. From 1979-1987, I was the founder and president of two computer software development companies in Pittsburgh, Pennsylvania, Unilogic, Ltd. and Lexeme Corporation.

9. I am a named inventor on the following five issued patents relating to electronic commerce: U.S. Patent Nos. 7,330,839, 7,421,278, 7,747,465, 8,195,197 and 8,280,773.

10. I am an attorney admitted to practice in Pennsylvania and have been admitted to the Bar of the U.S. Patent and Trademark Office since 1981. I have not been asked to offer any opinions on patent law in this proceeding.

11. I have previously testified in a large number of cases concerning computer technology. My curriculum vitae in Exhibit 1008 contains a list of cases in which I have testified in the last ten years.

12. As part of my work in forming my opinions in connection with this proceeding, I have reviewed the following materials, which I believe those in the field would reasonably rely upon in forming opinions regarding the subject matter of this proceeding:

| Exhibit | Title |
|---|---|
| Ex. 1001 | U.S. Patent No. 8,407,722 to Tuttle *et al.* ("*'722 Patent*") |
| Ex. 1002 | File History of U.S. Patent No. 8,407,722 ("*'722 File History*") |
| Ex. 1004 | U.S. Patent No. 6,999,991 to Ikeda ("*Ikeda*") |
| Ex. 1005 | U.S. Patent No. 6,480,883 to Tsutsumitake ("*Tsutsumitake*") |
| Ex. 1006 | European Patent Application No. EP1043671 to Bird *et al.* ("*Bird*") |
| Ex. 1007 | U.S. Patent No. 6,990,591 to Pearson ("*Pearson*") |
| Ex. 1008 | Curriculum Vitae of Dr. Michael Ian Shamos |
| Ex. 1009 | U.S. Patent Publication No. 2003/0058277 to Bowman-Amuah ("*Bowman-Amuah*") |
| Ex. 1010 | U.S. Patent No. 5,774,660 to Brendel *et al.* ("*Brendel*") |
| Ex. 1011 | U.S. Patent No. 6,807,558 to Hassett *et al.* ("*Hassett*") |
| Ex. 1013 | U.S. Patent No. 6,643,682 to Todd ("*Todd*") |
| Ex. 1014 | U.S. Patent No. 5,913,032 to Schwartz *et al.* ("*Schwartz*") |

| | |
|---|---|
| Ex. 1015 | U.S. Patent No. 6,886,044 to Miles *et al.* ("*Miles*") |
| Ex. 1016 | U.S. Patent No. 7,209,955 to Major *et al.* ("*Major*") |
| Ex. 1017 | U.S. Patent No. 6,778,530 to Greene ("*Greene*") |
| Ex. 1018 | U.S. Patent No. 7,580,919 to Hannel *et al.* ("*Hannel*") |
| Ex. 1019 | RFC 1700 - Assigned Numbers ("*RFC 1700*") |
| Ex. 1020 | U.S. Patent No. 6,775,692 to Albert *et al.* ("*Albert*") |
| Ex. AA | Claim Chart Comparing Challenged Claims of the '722 Patent to Prior Art |

## II. CLAIM CONSTRUCTION

13. I am informed that, in performing an obviousness analysis, it is necessary to understand the scope of the claims. I am also informed the first step in an unpatentability analysis, therefore, involves construing the claims. Second, the construed claim language is then compared to the disclosures of the prior art.

14. I am informed that claims are generally given their ordinary and custom meaning as understood by one of ordinary skill in the art at the time of the invention, in light of the patent specification. I am informed that, for the purposes of claim construction, while expert testimony may be helpful to provide background on the technology at issue, to explain how an invention works, to ensure an understanding of the technical aspects of the patent is consistent with that of a person having ordinary skill in the art, or to establish that a particular term in the patent had a particular meaning in the pertinent field at the time of the invention. However, I am also informed testimony from a technical expert is generally less reliable than the patent itself and its prosecution history in determining the meaning of claim terms, and it is not the role of technical experts to supplant courts and PTAB judges in applying principles of claim interpretation. However, the "Background of the Technology" that I provide in Section III.D of my opinion sheds light on background on the technology at issue in this proceeding and, hopefully, provides the adjudicators with an understanding of certain pertinent technical aspects that a person having ordinary skill in the art would have appreciated by the time of the '722 Patent.

15. I have applied the meaning of the claim terms of the Challenged Claims that is generally consistent with the terms' ordinary and customary meaning, as a person having ordinary skill in the art would have understood them in the context of the '722 Patent at the time of the alleged invention.

## III. OPINION

### A. Summary of My Opinions

16. In my opinion, the elements taught in the claims of the '722 Patent are taught by the combination of one or more references discussed below in light of the knowledge and skill that a person of ordinary skill in the art would have had by the time of the priority date of the '722 Patent, namely December 18, 2000 ("Priority Date"). If Patent Owner submits additional evidence pertaining to the patentability or priority date of the '722 Patent, I intend to review such materials and update my analysis and conclusions as appropriate and as permitted under the rules of this proceeding.

17. I have also relied on my education, experience, and knowledge in the relevant field of art in forming my opinions expressed below. The "Background of Technology" that I provide in Section III.D (¶¶ 22-33) below sheds light on the background of the technology at issue in this proceeding and, hopefully, will provide the adjudicators with an understanding of certain technical aspects that are pertinent to this proceeding. In my opinion, the cited contents accurately reflect the general knowledge that a POSITA would have had by the Priority Date. This background discussion is not exhaustive—a POSITA would have been familiar with many of the facets of digital information network transmission systems that are not described, but I have endeavored to touch on concepts relevant to the technology claimed.

### B. Summary of the '722 Patent and Its Prosecution History

18. I have reviewed, had input into, and endorse, the discussions in the Summary and Prosecution History of the '722 Patent sections of the Request, which I hereby incorporate by reference. *See* Request, Sections I.A.1., I.A.2.

### C. Level of Skill of a Person Having Ordinary Skill in the Art

19. I was asked to provide my opinion as to the level of skill of a person having ordinary skill in the field of the '722 Patent at the time of the claimed invention (referred to herein as a "POSITA"). I have been instructed to assume is that the time of the claimed invention of the '722 Patent is the Priority Date. In determining the level of skill of a POSITA, I was asked to consider several factors, including the type of problems encountered in the art, the solutions to those problems, the rapidity with which innovations are made in the field, the sophistication of the

technology, and the education level of active workers in the field. I also placed myself back in the time frame of the claimed invention and considered the colleagues with whom I had worked at that time.

20. In my opinion, a POSITA would be a person having, as of the Priority Date: (1) a Bachelor's degree in an academic area emphasizing computer science, electrical engineering, or equivalent degree, or closely related field, or equivalent relevant experience; and (2) at least two years of work experience with Internet routing. Additional education could substitute for less experience and vice versa. Such a person of ordinary skill in the art would have been capable of understanding the '722 Patent and the prior art references discussed herein, and would have been able to make and use the claimed inventions without undue experimentation.

21. Based on my education, training, and professional experience in the field of the claimed invention, I am familiar with the level and abilities of a POSITA. Additionally, I met at least these minimum qualifications at least as of the Priority Date. Further, although my qualifications may exceed those of the hypothetical person defined above, my analysis and opinions regarding the '722 Patent have been rendered from the perspective of a POSITA during the relevant time frame.

## D. Background of the Technology

22. Information Search and Retrieval Systems were well-known at the time of the alleged invention of the '722 Patent, including systems with many of the functions described by the '722 Patent as discussed below.

### *i. Transferring Information Through the Internet*

23. As explained by Bowman-Amuah, the architecture of the Internet, or Web, follows a conventional client-server model. *Bowman-Amuah* (Ex. 1009), [0007]. "The terms "client" and "server" are used to refer to a computer's general role as a requester of data (the client) or provider of data (the server)." *Id.* In a conventional environment:

> Web browsers reside in clients and Web documents reside in servers. Web clients and Web servers communicate using a protocol called "HyperText Transfer Protocol" (HTTP). A browser opens a connection to a server and initiates a request for a document. The server delivers the requested document, typically in the form of a text document coded in a standard Hypertext Markup Language (HTML) format, and when the connection is closed in the above interaction, the server serves a passive role, i.e., it accepts commands from the

client and cannot request the client to perform any action.

*Id.*; *see also Brendel* (Ex. 1010), 1:15-2:52; *Tsutsumitake* (Ex. 1005), 1:7-50 (both describing communications between client and server computers over the Internet). Accordingly, it was well-known that communications over the Internet took place between data servers and clients, which are both computers. Further, a POSITA would have understood that computers include processors, which execute software stored in a memory to carry out their operations.

24. It was well-known to use push services to provide updates to web pages, including updates to elements of a web page. For example, Hassett explains that push systems:

> bring[] new information to the user's desktop once an initial selection of news or other information items has been selected by the subscriber/user. In short, push technology is a system wherein each subscriber receives information, files and/or advertising from a network server for display at their local workstation on a refreshed and dynamic basis whenever a predetermined criteria, usually involving idleness of the local workstation, is met.

*Hassett* (Ex. 1011), 1:34-44. Push systems were commonly included in client browsers. *Bowman-Amuah* (Ex. 1009), [0716].

25. Publish/subscribe systems were also known systems for transferring information and updated information through digital networks. For example, Todd describes a known publish/subscribe system developed by the Transarc Corp. (a wholly owned subsidiary of the assignee of the present patent application, IBM Corp.), which is shown in the below figure. *Todd* (Ex. 1012), 1:51-54. Todd further discloses:

> Publishers 11 and 12 connect to the publish/subscribe broker network 2 and send published messages to broker network 2 which distributes the messages to subscribers 31, 32, 33, 34. Publishers 11 and 12, which are data processing applications which output data messages, connect to broker network 2 using the well known inter-application data connection protocol known as remote procedure call (or RPC) (other well known protocols, such as asynchronous message queuing protocols, can also be used). Each publisher application could be running on a separate machine, alternatively, a single machine could be running a plurality of publisher applications. The broker network 2 is made up of a plurality of distribution agents (21 through 27) which are connected in a hierarchical fashion which will be described below as a "tree structure". These distribution agents, each of which could be running on a separate machine, are data processing applications which distribute data messages through the broker network 2 from publishers to subscribers. Subscriber applications 31, 32, 33 and 34 connect to the broker network 2 via RPC in order to receive published messages. Publishers 11 and 12 first connect via RPC directly to a root distribution agent 21 which in turn connects via RPC to second level distribution agents 22 and 23 which in turn connect via RPC to third level distribution agents 24, 25, 26 and 27 (also known as "leaf distribution agents" since they are the final distribution agents in the tree structure). Each distribution agent could be running on its own machine, or

alternatively, groups of distribution agents could be running on the same machine. The leaf distribution agents connect via RPC to subscriber applications 31 through 34, each of which could be running on its own machine."

*Id.* at 1:54-2:19.

### *ii. Registering for Website Updates*

26. It was also known to cause client devices to register for updates (e.g., subscribe) to data objects (such as published pages of data) in response to receiving the page. For example, Schwartz describes a publish/subscribe system implemented in a client/server environment. *Schwartz* (Ex. 1013), Abstract, 7:55-8:6. Schwartz teaches that a publisher sends published pages to subscribers. *Id.* at Abstract, 2:50-65. In response to receiving these pages, the subscriber is prompted to accept the registration, which registers the subscriber to automatically receive updates to the data objects. *Id.* at Abstract, 2:50-65, 3:61-4:2.

### *iii. Routing Based on Data Categorization*

27. Routing information over the Internet was frequently done by categorizing data and routing based on the categories of data. For example, Miles teaches a system that routes data based on a specific service or product. *Miles* (Ex. 1014), Abstract. Specifically, Miles taught that specific nodes of a routing network were assigned to a "type" and these types of nodes handled specific data categories. *Id.* at 7:13-34, 12:28-59.

28. In addition to categorization based on the content (e.g., product or service), it was well-known in the art that users were interested in receiving all updates from a specific organization or information source (e.g., a sender of publisher). For example, Major describes methods and systems for triggering the receipt of real-time data items and notes that one method for categorizing data to be sent to users is based on whether the sender is a "preferred sender." *Major* (Ex. 1015), 6:21-26, 8:66-9:7, 9:57-10:21, 11:25-31, 19:27-34. Accordingly, Major teaches that one way of categorizing data is by filtering data based on the sender of the data.

29. Another method of categorizing data for routing was based on port number. For example, Greene teaches sorting packets based on flow classification rules, which

> may include fields that specify an IP destination address range, a destination port range, an IP source address range, a source port range, and a protocol. A number of such rules make up a classification table. For each incoming packet, the classification table is searched to

> determine which rule, if any, applies to the packet; a rule is said to apply to a packet only if the packet's header values in all five fields (destination address, destination port, source address, source port, and protocol) satisfy the corresponding ranges specified in the rule.

*Greene* (Ex. 1016), 2:33-43.

30. As explained by Hannel:

> . . . the messages used in Internet protocols are carried in packets called datagrams. Each such packet has a header which contains information indicating the source and destination of the packet. The source and destination are each expressed in terms of IP address and port number. A port number is a number from 1 to 65535 used to individuate multiple streams of traffic within a computer. Services for well-known Internet protocols (such as, HTTP or FTP) are assigned well known port numbers that they 'listen' to.

*Hannel* (Ex. 1017), 3:34-42. One known well-known port number was the network news transport protocol, which utilized port 119. RFC 1700 (Ex. 1018), 44; *Hannel* (Ex. 1017), 42:12-26 (discussing network news transfer protocol).

### ***iv. Proxy Arrangements***

31. Proxy arrangements were well-known and commonly used on the Internet. In computer networking, proxies act as intermediaries or stand ins between a requesting device (e.g., a client) and a host device (e.g., a server). For example, Albert describes the use of a proxy machine for security purposes, where a proxy machine serves as a stand-in for a host machine storing sensitive information. *Albert* (Ex. 1019), 3:64-4:19. Albert also notes that proxies can also be used for other purposes such as load balancing. *Id.* at 2:15-46.

### ***v. Applets and JavaScript***

32. Applets were well-known at the time of the alleged invention of the '722 Patent. As explained by England:

> A Web page can also contain other Internet Resources such as "applets", "plugins", and scripting language. All of these technologies add intelligence and interactivity to Web pages and support a greater range of functionality. An applet is a (small) application program that typically is stored on the Web server. The applet is downloaded (i.e. transfers from the Web server to the user's PC system) with the HTML of the Web page when a Web page is requested by the user. Once the applet is downloaded, it is activated and runs on the user's PC system. A common language for writing applets is the Java programming language, a language that allows Web masters (i.e. people who design Web pages) to create animated and interactive Web pages.

*England* (Ex. 1020), 4:13-24.

33. JavaScript (which, curiously, bears no relation to Java), became a well-known

scripting language that provided programmability to HTML pages. It is one of the scripting languages referred to above by England. It was so popular that by 1998 the third edition of the *JavaScript Bible* had already been published.

## E. Ground 1: The Combination of Ikeda, Tsutsumitake, and Bird

### *i. Overview of Ikeda*

34. 18. I have reviewed, had input into, and endorse the discussions in the Overview of Ikeda section of the Request, which I hereby incorporate by reference. *See* Request, Section I.E.1.a.

35. Ikeda is analogous art to the '722 Patent because Ikeda is in the same field of endeavor as the '722 Patent and because Ikeda is reasonably pertinent to at least one problem with which the inventor of the '722 Patent was concerned.

36. Thus, the '722 Patent is in the field of "transferring information through digital networks." *'722 Patent* (Ex. 1001), 1:24-27. Ikeda is in the same field of endeavor as the '722 Patent. For example, Ikeda relates to "a push service system and a push service processing method for processing push service in a network using an IP protocol." *Ikeda* (Ex. 1004), 1:6-8. Indeed, Ikeda expressly teaches a push service used in systems that transmit (i.e., transfer) data of a host apparatus to a user apparatus over the Internet (i.e., a digital network). *Id.* at 1:12-17. Thus, Ikeda is also in the field of "transferring information through digital networks.

37. Ikeda is also reasonably pertinent to at least one problem with which the inventor of the '722 Patent was concerned. One problem identified by the '722 Patent was the need for client devices to periodically re-request web pages to obtain updated information, because it results in an increase of unnecessary traffic (e.g., in the form of user requests). *'722 Patent* (Ex. 1001), 2:38-53. The '722 Patent seeks to solve this issue by sending update messages to a routing network to which client devices have registered for updates only when updates occur and routing the message through a hierarchy of nodes. *Id.* at Abstract, 16:47-50, 17:66-18:2. Ikeda similarly teaches that a disadvantage of conventional push services is the increase of useless traffic on the network due to constant re-checking for updates by user terminals. *Ikeda* (Ex. 1004), 1:37-41 ("[T]he conventional push services where the access is executed per constant time, since a check is made as to whether or not the information in the data source is updated in each access, there arises a problem that useless traffic on the network increase."). Ikeda seeks to solve this issue by

providing routing information about updating from data servers to clients which have registered with the network and forwarding the information through a hierarchy of agents, just like the '722 Patent. *Ikeda* (Ex. 1004), 1:58-62, 3:31-61.

### *ii. Overview of Tsutsumitake*

38. I have reviewed, had input into, and endorse the discussions in the Overview of Tsutsumitake section of the Request, which I hereby incorporate by reference. *See* Request, Section I.E.1.b.

39. Tsutsumitake is analogous art to the '722 Patent because Tsutsumitake is in the same field of endeavor as the '722 Patent and because Tsutsumitake is reasonably pertinent to at least one problem with which the inventor of the '722 Patent was concerned.

40. As discussed in ¶ 36 above, the '722 Patent is in the field of "transferring information through digital networks." Tsutsumitake is in the same field of endeavor as the '722 Patent. Tsutsumitake "relates generally to a client/server type system wherein an information providing server provides information upon request from an information display client, and more particularly to a real-time information transmission system for realizing novel and advantageous information transmission from an information providing server to an information display client" over the Internet. *Tsutsumitake* (Ex. 1005), 1:7-20, 7:63-8:6. Thus, because Tsutsumitake relates to transferring information in real time between servers and clients over the Internet, Tsutsumitake is also in the field of "transferring information through digital networks."

41. Tsutsumitake is also reasonably pertinent to at least one problem with which the inventor of the '722 Patent was concerned. One problem identified by the '722 Patent was techniques for causing a client to register for updates to live objects within data representations such as websites. *'722 Patent* (Ex. 1001), 5:32-36. The '722 Patent attempted to solve this issue by causing the client to respond to a web page that includes a live object by registering for information. *Id.* at 10:58-11:20, 5:66-6:2. Similarly, Tsutsumitake teaches that clients register for updates to updateable elements of a web page in response to receiving the web page. *Tsutsumitake* (Ex. 1005), 9:60-10:32, 10:64-11:8. And just like the '722 Patent, which teaches that the client determines the presence of a live object for registration through the inclusion of an identifier in the web page, Tsutsumitake similarly teaches that a client detects an updateable element through the inclusion of a URL (which identifies the updateable element) in the web page. *Compare '722*

*Patent* (Ex. 1001), 8:25-9:49 *with Tsutsumitake* (Ex. 1005), 9:60-10:32.

### *iii. Overview of Bird*

42. I have reviewed, had input into, and endorse the discussions in the Overview of Bird section of the Request, which I hereby incorporate by reference. *See* Request, Section I.E.1.c.

43. Bird is analogous art to the '722 Patent because Bird is in the same field of endeavor as the '722 Patent and because Bird is reasonably pertinent to at least one problem with which the inventor of the '722 Patent was concerned

44. As discussed in ¶ 36 above the field endeavor of the '722 Patent is "transferring information through digital networks." Bird is in the same field of endeavor as the '722 Patent. Bird relates to "message broker providing a publish/subscribe service, and to a method of processing message content in a publish/subscribe environment." *Bird* (Ex. 1006), [0001]. As explained by Bird "[p]ublish/subscribe is a known information distribution mechanism in which information providers (Publishers) send information electronically across a network to a community of information users (Subscribers) who have chosen the topics of information they wish to receive." *Id.* at [0001]. Thus, publish/subscribe systems are systems that distribute information over a network, and because Bird describes a publish/subscribe system, Bird is in the field of "transferring information through digital networks."

45. Bird is also reasonably pertinent to at least one problem concerning the inventor of the '722 Patent. One problem identified by the '722 Patent was techniques for categorizing data for the purposes of forwarding the data and an example provided for categorizing data was to assign all messages from the same input source to one category. *'722 Patent* (Ex. 1001), 18:52-54. Similarly, Bird teaches that data can be categorized based on the publisher (e.g., input source). *Bird* (Ex. 1006), [0044], [0060].

### *iv. Motivation to Combine Ikeda and Tsutsumitake such that an input source provides a data representation, including a live object, to a client device, different from the input source (Claims [14.1(a)], [20.1(a)], [26.1(a)], [32.1(a)])*

46. As explained below, a POSITA would have combined Ikeda and Tsutsumitake such that Ikeda's push system would have been implemented to provide web pages to requesting user terminals, as well as updates to those web pages. As a result, Ikeda's data servers would have

provided data representations, including live objects, to user terminals.

47. Ikeda describes a push system that includes data servers which contain information that is accessed by user terminals via the Internet (e.g., a routing network). *Ikeda* (Ex. 1004), 19:26-28, 19:48-52, 20:21-23, Fig. 15. Thus, Ikeda teaches that data servers (*input sources*) provide information to user terminals that are different from the data servers (i.e., *client devices*), coupled to the Internet (e.g., *a routing network*)

48. A POSITA would have understood that the data servers provide this information to user terminals using a processing device. Ikeda discloses the data servers include processing sections. *Ikeda* (Ex. 1004), 19:67-20:20. As discussed in ¶¶ 23-25 above, a POSITA would have been familiar with communications via an IP network and would have understood that data servers, such as the data servers disclosed by Ikeda, are computers. A POSITA would have further understood that computers included processing devices, such as CPUs, which executed software to perform various operations, which in the case of Ikeda's data servers, would have included providing data and information to client devices. *Ikeda* (Ex. 1004), 6:47-64, 19:67-20:20. Indeed, this is supported by Tsutsumitake. *Tsutsumitake* (Ex. 1005), 5:36-38, 6:16-18. Thus, Ikeda teaches that data servers (e.g., input sources), using a processing section (e.g., processing device), provide information to user terminals that are different from the data servers (e.g., client devices), coupled to the Internet (e.g., a routing network).

49. Ikeda broadly discloses that its push system is used in any networked environment such as the Internet. A POSITA would have understood such an environment to include the provision of web pages, including live objects, as well as updated live object data provided over the Internet. However, to the extent more would have been required, a POSITA would have turned to Tsutsumitake, which provides an example of employing a push system to provide updated live object data in web pages. *Tsutsumitake* (Ex. 1005), 1:7-12, 3:20-26, [Abstract], 10:10-15. A POSITA would have turned to Tsutsumitake in particular because Tsutsumitake discloses a similar system to Ikeda (e.g., servers providing information to clients over the Internet) and also discloses a push system. Tsutsumitake describes a push system that operates to update an element (e.g., a live object) of a web page. *Tsutsumitake* (Ex. 1005), 1:7-12, 3:20-26, [Abstract], 10:10-15.

50. In more detail, Tsutsumitake describes an environment that includes information providing servers that provide information to clients in the form of web pages over the Internet. *Id.* at 1:22-28. Tsutsumitake's clients receive these web pages and display them to users. *Id.*

Tsutsumitake notes that push systems are commonly used in such environments to update content of the web pages. *Id.* at 1:7-2:7. Thus, Tsutsumitake describes a content providing server (e.g., an input source) providing a web page (e.g., a data representation) to a client that is different than the content providing server (e.g., a client device).

51. A POSITA would have understood that Tsutsumitake's web pages include live objects because they include elements (e.g., text values) that are updated in real time. The '722 Patent teaches that an "'object' is any datum or data at the client 114 that can be individually identified or accessed," such as "elements of web pages such as text characters and strings, images, frames, tables, audio, video, applets, scripts, HTML, XML, and other code forming the web page, variables and other information used by applets, scripts and/or code, URLs embedded in the web page, etc." *'722 Patent* (Ex. 1001), 6:66-7:6. The '722 Patent teaches that the "[p]roperties of a live object can be dynamically updated in real-time at the client 114." '722 Patent (Ex. 1001), 6:63-66.

52. Tsutsumitake teaches that an object of the invention is to provide a real-time information transmission system that allows information updated on the server side to be efficiently reflected on the client side. *Tsutsumitake* (Ex. 1005),3:27-33. Tsutsumitake discloses that portions of a webpage (i.e., a live object) may be updated dynamically when the client receives an event from the server. *Tsutsumitake* (Ex. 1005), [Abstract]. Tsutsumitake provides one example of a web page that monitors the state of current in a specific control device in a plant system. *Tsutsumitake* (Ex. 1005), 13:31-34, Fig. 7. The page displays a current value display section and a state display section. *Id.* at 13:34-37, Fig. 7. An HTML expression of the page shows that the values in both sections are individually identified in the HTML and are individually accessed for updating because they are updated based on different rules. *Id.* at 14:4-15, Fig. 8, 13:38-55, 13:58-65. Both of these sections are updated by the server in real time. *Tsutsumitake* (Ex. 1005), 13:56-57. Thus, a POSITA would have understood these two values to be elements (e.g., objects) of a web page. The elements on Tsutsumitake's web pages are "live" because they can be individually accessed, are individually identified, and are updated in real-time at the client by the information provider. Accordingly, Tsutsumitake further teaches that the webpage (e.g., data representation) includes at least one updateable element (e.g., live object).

53. Tsutsumitake further teaches that the web page include "event requests," which are associated with elements of a web page. *Id.* at 13:38-55, 14:4-16. When a web page includes an

event request, then Tsutsumitake's client provides the event request to the event request unit 113. *Id.* at 9:60-67. Thus, Tsutsumitake's client recognizes that a webpage includes an updateable element when it detects an event request on a web page.

54. As noted above, because Ikeda does not teach express examples of where the push system is implemented, and because Tsutsumitake teaches a similar networked environment that uses a push system, a POSITA would have implemented Ikeda with Tsutsumitake's teachings to provide web pages and updates to these web pages (which is an application taught by Tsutsumitake). In the proposed Ikeda-Tsutsumitake combination, Tsutsumitake provides implementation details for employing Ikeda's push system—which is disclosed as operating in any network environment—in the context of providing and updating web pages (an application taught by Tsutsumitake). With respect to providing web pages to user terminals, in the proposed Ikeda-Tsutsumitake combination, Ikeda's data servers would have stored data relating to web pages and data relating to changes to the web pages, just as Tsutsumitake's servers stored web page data and updated web page data. Ikeda's user terminals would have requested a web page from the data servers, just as Tsutsumitake's clients requested web pages from information providing servers. Ikeda's data servers would have provided these web pages to the user terminals (using the CPU of the server) in response to the requests. Upon receipt, the user terminals would have displayed the web page, including updateable elements of a web page (i.e., live objects), to a user. The display of the web page and elements of the web page, as well as the ability of the client to process update events from the server destined for the object, demonstrate that the elements are recognizable by the client device. In other words, as implemented with Tsutsumitake's teachings, Ikeda's client would have recognized the inclusion of an "event request" in a web page that indicates the presence of an updateable element.

55. With respect to providing updates to web pages, in the proposed combination, user terminals would have registered for updates to information, as already taught by Ikeda. *Ikeda* (Ex. 1004), 19:40-56, 20:10-14. In the proposed combination, these registrations would have been registrations for updates to the web page as discussed in detail in ¶¶ 61-68 below. The data servers would have transmitted information about updating the contents of the servers to user terminals through a hierarchy of agents, as already disclosed by Ikeda. *Ikeda* (Ex. 1004), 19:48-65, 20:21-21:12. Thus, in the proposed combination, the information about updating sent by Ikeda's push service would have been information about updating a web page, which is discuss further in ¶¶ 69-

73 below. And the user terminals would have used the received information to update the web pages, as discussed further in ¶¶ 82-84 below.

56. A POSITA would have made the proposed combination for the following reasons: 1) the combination is the combining of known prior art elements according to known methods to yield predictable results; and 2) there is some teaching, suggestion, or motivation in the prior art that would have led one of ordinary skill to modify the prior art reference or to combine prior art reference teachings to arrive at the claimed invention.

57. A POSITA would have been motivated to make the proposed combination because it is the combination of known prior art elements (e.g., Ikeda's push service and Tsutsumitake's conventional teachings of a real-time information transmission system from a server to a client in a WWW framework) according to known methods (e.g., using a push service to update web page content) to yield predictable results (e.g., to utilize Ikeda's data servers to provide websites that include updateable elements to requesting user terminals for display). A POSITA would have recognized that using push systems to provide updates to websites was well-known in the prior art, as discussed in ¶¶ 23-25 above, including as taught by Tsutsumitake. Indeed, Tsutsumitake expressly teaches that website updates were communicated to client devices by data services using a push service and further acknowledges that such use was common and conventional. *Tsutsumitake* (Ex. 1005), 1:7-3:26. Accordingly, implementation of Ikeda with Tsutsumitake's teachings would not have required undue experimentation and would have been well-within the skillset of a POSITA.

58. Second, a POSITA would have been motivated to implement Ikeda's system to provide web pages updates because there is some teaching, suggestion, or motivation in the prior art that would have led one of ordinary skill to combine Ikeda's teachings and Tsutsumitake's teachings to arrive at the claimed invention (e.g., to implement Ikeda's push system for real-time information transmission from a server to a client in a WWW framework). For example, Tsutsumitake expressly suggests that a push service would have been used to update web page content. *Tsutsumitake* (Ex. 1005), 1:7-3:26. Accordingly, use of Ikeda's push service to update web page content would have been an obvious implementation in light of Tsutsumitake's teachings.

59. Third, a POSITA would have been motivated to implement Ikeda's system to provide web page updates because the proposed modification is application of a known technique

(e.g., Ikeda's push system) to a known device ready for improvement (e.g., conventional client/server information transmission systems in a WWW framework) to yield predictable results (e.g., decrease useless traffic on the Internet). For example, Ikeda expressly teaches that advantages of the push service system include decreasing useless traffic on the Internet. *Ikeda* (Ex. 1004), 21:49-59. Thus, a POSITA would have implemented Ikeda to update web pages to achieve this express advantage in the web page update context.

60. A POSITA would have had a reasonable expectation of success in implementing Ikeda's push service for website updates. For example, the use of a push service to update web pages was well-known and common in the prior art, including as taught by Tsutsumitake. *Tsutsumitake* (Ex. 1005), 1:7-3:26. Thus, a POSITA would have reasonably expected to succeed in implementing Ikeda's push system to provide updates to web pages. Further, a POSITA would also have had a reasonable expectation of success given the similarities between the Ikeda and Tsutsumitake systems. For example, Ikeda teaches servers that provide data to user terminals over a network. *Ikeda* (Ex. 1004), 19:24-39, Fig. 15, 1:12-53. Similarly, Tsutsumitake teaches servers that provide data to clients over a network. *Tsutsumitake* (Ex. 1005), 1:7-36, 3:20-26, 7:63-8:6, 8:33-10:15. Thus, because Ikeda and Tsutsumitake describe similar systems, implementation of Ikeda with Tsutsumitake's teachings would have been simple, straightforward, and well-within the skillset of a POSITA. Indeed, Ikeda's system already includes the functionality required to provide web pages to user terminals. Accordingly, implementation of Ikeda as described would not have required undue experimentation would have yielded the predictable result of utilizing Ikeda's push system to provide updates to web pages provided by data servers.

***v. Motivation to Combine Ikeda and Tsutsumitake such that in response to receiving the live object, the client device determines an object identifier of the live object and registers for updates of the live object (Claims [14.1(b)], [20.1(b)], [26.1(b)], [32.1(b)])***

61. Ikeda teaches that a client device registers for updates (e.g., registers for updates) of information from a data server, by submitting a registration request to a low-level agent coupled to the Internet (e.g., a node in the routing network). *Ikeda* (Ex. 1004), 19:40-44, 20:10-14. The registration request includes a data type and client connection information (e.g., address of the agent, and "terminal information D15 such as equipment type and the like, and another terminal information D16 held by the users such as equipment type, telephone number," and address). *Id.*

at 19:40-44, 8:58-9:1. Accordingly, registering the user terminal with the routing network provides client information to an agent in the routing network. Ikeda does not expressly teach that the registration request is sent in response to receiving the live object of the data representation and does not expressly teach that the client determines an object ID of the live object.

62. However, Tsutsumitake teaches that in response to receiving a web page including an updateable element from a server, a client computer determines an object ID and also registers for updates to the updateable element. *Tsutsumitake* (Ex. 1005), 9:60-10:6, 11:43-55. In more detail, Tsutsumitake teaches that when a webpage includes updateable elements, the web page includes an "event request." *Id.* at 10:20-29. When an event request is included in the page, the client recognizes the event and submits the event request, which at least includes a URL. *Id.* at 9:60-10:6. Thus, Tsutsumitake teaches causing a client device (e.g., client) to respond to a live object of the data representation (e.g., updateable element of a web page) by registering for updates.

63. Tsutsumitake further teaches that the client determines an object ID in response to receiving and recognizing an event request associated with an updateable element. First, Tsutsumitake teaches that a registration request from a client includes an event ID and also a connection ID. *Id.* at 11:43-55, Abstract, 12:64-13:5. A POSITA would have understood that the event ID would have been sent by the client device when registering for updates to the web page (e.g., when sending the event request). For example, Tsutsumitake teaches that a registry is maintained that includes an event ID. *Tsutsumitake* (Ex. 1005), 11:43-62. Thus, Tsutsumitake's event ID at least renders obvious an object identifier as claimed. However, in the alternative this event ID is not an object identifier, as claimed, Tsutsumitake teaches that event requests are "expressed by an attribute 'URL' in a tag 'EVENT' in a tag 'EMBED.'" *Tsutsumitake* (Ex. 1005), 10:20-29. A POSITA would have recognized that such an attribute URL is an identifier because it would have been unique to each updateable element in a web. Because Tsutsumitake teaches that this URL is then used for registration (*Tsutsumitake* (Ex. 1005), 9:60-10:6), a POSITA would have understood that the URL is an object identifier. For example, the URL would have served as the event ID.

64. In the proposed Ikeda-Tsutsumitake combination, Ikeda's user terminals, upon receiving a web page from a server, would have recognized that the web page included an event request, as taught by Tsutsumitake. *Tsutsumitake* (Ex. 1005), *Id.* at 9:60-10:6, 11:43-55. Upon

recognizing the event request, Ikeda's client would have submitted a registration request to the routing network, as taught by Ikeda. *Ikeda* (Ex. 1004), 19:40-44, 20:10-14. In the proposed combination, the user terminal (e.g., client) would have determined an event ID of the event request (e.g., the URL) and included this in the registration request, per Tsutsumitake's teachings, and the user terminal would also have included the client connection information and data type already included in Ikeda's registration request.

65. A POSITA would have made the proposed combination for the following reasons: 1) the combination is the use of a known technique to improve similar devices in the same way; and 2) the combination is combining known prior art elements according to known methods to yield predictable results.

66. A POSITA would have been motivated to make the proposed combination because it is the use of a known technique (e.g., Tsutsumitake's known technique of providing an event request associated with updateable website elements to trigger registration for updates to those elements) to improve similar devices (e.g., Ikeda's push system) in the same way (e.g., to register for updates to elements of a web page upon recognizing an event request). A POSITA would have recognized that while Ikeda teaches that, in a push service, user terminals submit registration requests, Ikeda does not teach how a user terminal submits these requests. Thus, a POSITA would have looked to other references describing push services to determine how to implement a registration request submission. Tsutsumitake teaches a push service in which registration requests are submitted in response to receiving web pages with elements that are to be updated via the push service. *Tsutsumitake* (Ex. 1005), 9:60-10:29. And Tsutsumitake's environment is very similar to the environment disclosed by Ikeda (e.g., both disclose data servers providing information to user terminals or clients over the Internet). *Tsutsumitake* (Ex. 1005), 7:63-8:23; *Ikeda* (Ex. 1004), 19:24-29. Accordingly, a POSITA would have been motivated to modify Ikeda to submit registration requests in response to recognizing the inclusion of event requests on a received web page.

67. A POSITA would have been motivated to make the proposed combination because Tsutsumitake provides implementation details for Ikeda's push system, which is silent as to how registration submissions are provided. A POSITA would have been familiar with push services, which were well-known for use in data transfers over the Internet, including as demonstrated by both Ikeda (*Ikeda* (Ex. 1004), 19:24-39, Fig. 15, 1:12-53) and Tsutsumitake (*Tsutsumitake* (Ex.

1005), 1:7-3:26) and as discussed in ¶¶ 23-25 above. Further, it was known in the art to trigger actions of a client device—including registration for updates to information—in response to receiving a web page, as taught by Tsutsumitake, and as discussed in ¶ 26 above. Thus, a POSITA would have modified Ikeda with Tsutsumitake's teachings to submit registration requests in response to receiving web pages that include updateable elements. Accordingly, the proposed combination because it is the combination of known prior art elements (e.g., Tsutsumitake's known technique of providing an event request associated with updateable website elements to trigger registration for updates to those elements in a push service and Ikeda's known push service) according to known methods (e.g., including event requests on web pages transmitted from a server to a client) to yield predictable results (e.g., to submit a registration request in response to receiving a web page including an updateable element).

68. A POSITA would have had a reasonable expectation of success in making the proposed modification given the similarities in both the architecture and functions of the systems of Ikeda and Tsutsumitake. For example, both Ikeda and Tsutsumitake describe information push systems used in a networked environment. *Ikeda* (Ex. 1004), 1:58-62; *Tsutsumitake* (Ex. 1005), 1:7-3:26. Further, both Ikeda and Tsutsumitake's systems include data servers which provide information to user terminals or clients in response to requests from those clients. *Ikeda* (Ex. 1004), 1:12-23; *Tsutsumitake* (Ex. 1005), 1:7-36, 3:20-26, 7:63-8:6, 8:33-10:15. Accordingly, implementation of Ikeda with Tsutsumitake's teachings would have been straightforward and well-within the skillset of a POSITA. Indeed, such a combination would have required only minor software modifications to Ikeda's system to include event requests in web pages provided by the data servers and to include additional information in user terminal registration requests.

***vi. Motivation to Combine Ikeda and Tsutsumitake such that an input source sends an update message to the routing network that identifies the live object and contains update data that updates a property of the live object (Claims [1.1], [7.2], [14.2], [20.2], [26.2], [32.2])***

69. Ikeda teaches that the data servers update information held and managed by the servers and provide messages about the updated information to routing network. *Ikeda* (Ex. 1004), 19:48-52, 20:21-23, 6:60-64. As discussed in ¶¶ 48, 23 above, Ikeda's data servers included processing sections, which a POSITA would have understood performed the operations of the data server, including sending messages about updated information.

70. Ikeda discloses that the data servers include information for data stored within the server, this data included data ID, data contents, and changed data. *Id.* at 9:53-58, Fig. 6. Ikeda notes that at least data type information such as ID and data contents is essential information for realizing the push service. *Id.* at 9:58-60. Ikeda further teaches that forwarding is accomplished in the push service by agents comparing information about updating with information in their registries, such as data ID because in order to compare the data ID, it must have been included in the update message. *Id.* at 20:31-38. Thus, a POSITA would have recognized that the update message includes the data ID, thereby identifying the live object. Further, Ikeda teaches that agents may pass data contents and changed data directly to user terminals. *Id.* at 21:29-44, 19:60-64, 21:6-22, 11:55-65, 12:11-17. Thus, Ikeda teaches that the update message sent by the data server includes update data.

71. As discussed in ¶¶ 46-50 above, a POSITA would have implemented Ikeda's push system to update web page information pursuant to Tsutsumitake's teachings. Accordingly, in the Ikeda-Tsutsumitake system, Ikeda's update information would have been information updating elements of the stored web pages. This information would have included the information taught by Ikeda—namely, an identification of the data (i.e., live object), and the contents of the updated data. Indeed, it is necessary to identify which live object is to be updated. Accordingly, in the Ikeda-Tsutsumitake combination, Ikeda's update message would have identified the live object just as Ikeda teaches that an update messages identifies the data and would have also contained update data, just as Ikeda's update message contained update data.

72. Finally, in the proposed Ikeda-Tsutsumitake combination, the update data would have updated a property of the live object. Tsutsumitake teaches that changes to data elements update a property of the data element. For example, Tsutsumitake teaches that a current value is displayed as a number and this number is updated in real-time. *Tsutsumitake* (Ex. 1005), 13:31-37, 13:58-65, 14:14-28. The '722 Patent teaches that properties of live objects are "any modifiable data related to the object or referenced with respect to the object" and include properties affecting visual aspects, such as "content, color, typeface, size, formatting, or other attribute of text, images, or other data displayed by the object." *'722 Patent* (Ex. 1001), 7:35-38. Thus, because Tsutsumitake teaches changing an attribute of a text value on a web page, Tsutsumitake teaches updating a property of a live object (as opposed to re-sending the entire page). And because, as discussed in ¶¶ 51, 70 above, in the Ikeda-Tsutsumitake combination, Ikeda's update message

includes data contents for updating data of a web page, the Ikeda-Tsutsumitake update message would have contained data for updating a property of an element of a web page (i.e., a live object). Accordingly, in the proposed combination, the messages about updated information in Ikeda, when implemented according to Tsutsumitake's teachings, would have been messages regarding changes to data elements of the web page. Thus, in the Ikeda-Tsutsumitake, Ikeda's data servers would have sent information about updating (e.g., update messages) that included an identifier (e.g., data ID) of the updateable element (e.g., live object) and update data for the date element (data contents and/or changed data for the live object), which updates a property of the data element by visually changing the text value.

73. A POSITA would have implemented Ikeda with Tsutsumitake's teachings for the same reasons discussed in ¶¶ 46-60 above.

***vii. Motivation to Combine Ikeda (as modified by Tsutsumitake) and Bird such that a gateway device at the routing network is configured to identify a category of the update message based on the input source, to determine a node type to which the identified category maps, and to route the update message to the node, having the node type , at the routing network (Claims [1.2], [7.3], [14.3], [20.3], [26.3], [32.3])***

74. I agree with the discussions of Ikeda set forth in the Request (*see* Section I.E.2.d.) and in Exhibit AA, including that Ikeda teaches a high-level agent (*a gateway device at the routing network*) determines a type of data represented by the information about updating (*is configured to identify a category of the update message*) and determines which nodes are assigned to handle such data types and routes the information about updating to the determined node (*determine a node . . . to which the identified category maps, and to route the update message to the node . . . at the routing network*). Ikeda does not expressly teach that the category of the update message is based on input source. However, it is taught by Bird.

75. Bird discloses a publish / subscribe system in which a subscriber (such as a client device) registers to receive updated information from publishers. *Bird* (Ex. 1006), [0060]. Bird teaches that subscribers or clients specify a type or topic of data to which they wish to receive updates. *Id.* Bird expressly discloses that an input source (i.e., a sender or publisher of information) is one category of information from which a user may wish to receive. *Bird* (Ex. 1006), [0060], [0003]. Bird further teaches that this is determined by reviewing the header of a packet of

information. *Id.* at [0044].

76. A POSITA would have modified Ikeda with Bird such that the name of a publisher of information (i.e., Ikeda's data sources) is one of the data types for which a client computer may register. Modified as such, one type of for which a user terminal would have requested registration would have been data corresponding to a specific input source. As modified, the high-level agent would have determined a data type by looking to a sender identifier retrieved from the message header of a published message, as expressly taught by Bird. *Bird* (Ex. 1006), [0044]. This would then have been compared to the data types for which various low-level agents are registered to receive, as taught by Ikeda.

77. A POSITA would have made the proposed combination for the following reasons: 1) the combination is the use of a known technique to improve similar devices in the same way; 2) known work in one field of endeavor may prompt variations of it for use in either the same field or a different one based on design incentives or other market forces if the variations are predictable to one of ordinary skill in the art; and 3) the combination is combining known prior art elements according to known methods to yield predictable results.

78. A POSITA would have been motivated to make the proposed combination because the proposed modification is the use of a known technique (e.g., Bird's known technique of categorizing data messages based on input source) to improve similar devices (e.g., Ikeda's push service) in the same way (e.g., providing users with increased options in subscribing for updates). A POSITA would have been familiar with registering for updates from particular publishers or senders, as discussed in ¶¶ 27-28 above. A POSITA would have recognized that such a modification would have provided increased options to a user for registering for updates. A POSITA would have understood that providing a wider variety of options would have been desirable to users, thus enhancing Ikeda's push service. Thus, a POSITA would have recognized that including Brid's teachings would have improved Ikeda's push system. Accordingly, a POSITA would have been motivated to modify Ikeda to allow users to select an input source or sender as a data type.

79. A POSITA would have been motivated to make the proposed combination because known work in one field of endeavor (e.g., Bird's known categorization of data based on input source in publish/subscribe systems used for transferring information over the Internet) would have prompted variations of it for use in the same field (Ikeda's push system used for transferring

information over the Internet) based on market forces (e.g., increasing options of types of data for user registration) and such variations (e.g., categorizing data based on input source during website updates) would have been predictable to a POSITA (e.g., the modifications would have been simple and straightforward to a POSITA). As noted in ¶ 78 above, a POSITA would have recognized the benefits to using an input source as a type of data (e.g., provide a wide variety of options to a user), as taught by Bird. And as discussed in ¶¶ 36, 44 above, both Bird and Ikeda relate to transferring information over a digital network. Further, both Bird and Ikeda describe similar systems including data servers sending information to user terminals. *Ikeda* (Ex. 1004), 1:12-36, 19:16-29; *Bird* (Ex. 1006), [0002], [0005], [0030]. Thus, the result of such a combination would have been predictable to a POSITA given the similarity between the systems and given the fact that a POSITA would have been familiar with Internet communications.

80. A POSITA would have been motivated to modify Ikeda with Bird's teachings because push systems, such as the system taught by Ikeda, were well-known systems in which users request updates to specific data types, as discussed in ¶ 25 above and as taught by Ikeda (*Ikeda* (Ex. 1004), 1:24-30). It was well-known in the art that one type of data in which users would have been interested was data from a specific publisher or sender, including as disclosed by Bird and as discussed in ¶¶ 75, 78 above. Accordingly, a POSITA would have modified Ikeda with Bird's teachings to register with a data type of a specific publisher. Thus, the proposed is the combination of known prior art elements (e.g., Bird's known technique of categorizing data messages based on input source with Ikeda's known push system) according to known methods (e.g., examining the header of a data message) to yield predictable results (e.g., routing the data message to an appropriate node type for forwarding a the data message to a client).

81. A POSITA would have had a reasonable expectation of success in making the proposed modification given the similarities in the architecture of the systems of Ikeda and Bird. For example, Ikeda describes a push system, which provides information to users over the Internet. *Ikeda* (Ex. 1004), 1:12-36, 19:16-29. Similarly, Bird describes a publish/subscribe system, which provides information to users over the Internet. *Bird* (Ex. 1006), [0002], [0005], [0030]. A POSITA would also have had a reasonable expectation of success in making the proposed modification given the similarities in functions of the systems of Ikeda and Bird. In Ikeda's push system, users request updates for specific types of information. *Ikeda* (Ex. 1004), 19:40-44, 8:58-9:1. Likewise, in Bird's publish/subscribe system, users request updates to specific types of

information. *Bird* (Ex. 1006), [0039]. Additionally, as discussed in ¶¶ 27-28 above, it was well-known to subscribe to information from specific publishers or senders. Accordingly, modification of Ikeda with Bird's teachings would have been straightforward and well-within the skillset of a POSITA. Indeed, such a combination would have required only minor software modifications to Ikeda's system to indicate a specific publisher or data source as a type of data and to look to the sender ID in the header when routing such data.

***viii. Motivation to Combine such that the user terminal processes information about updating to update an element of a web page (Claims [1.7], [7.8(b)], [14.5], [20.5], [26.5], [32.5])***

82. Ikeda teaches that upon receiving the information about updating, the client device either obtains the information from the data server, or the contents can be transmitted to the client by the agents. *Ikeda* (Ex. 1004), 19:60-64, 21:4-44. Ikeda does not expressly teach that the information about updating updates a property of a live object. However, Tsutsumitake teaches that web pages include data elements, such as text values (e.g., current value of electric current). *Tsutsumitake* (Ex. 1005), 13:31-37. Tsutsumitake expressly teaches that a text value is changed as a result of an update. *Id.* at 13:58-65, Fig. 7.

83. As discussed in ¶¶ 46-60 above, a POSITA would have implemented Ikeda's push service according to Tsutsumitake's teachings to provide web pages and provide updates web pages. In the proposed combination, Ikeda's information about updating would have been used to update an element of a web page, just as Tsutsumitake teaches updating an element of a web page.

84. A POSITA would have been motivated to make the proposed modification, and had a reasonable expectation of success in making the proposed modification, for the same reasons discussed in ¶¶ 46-60 above.

***ix. Ikeda Teaches providing the data representation to the client device includes providing the live object that causes the client device to register with a client proxy of the routing network (Claims 15, 21, 27, 33)***

85. Ikeda teaches that a client registers with a low-level agent, which serves as a substitute for a user terminal. *Ikeda* (Ex. 1004), 19:30-39. A POSITA would have understood that client proxies were devices or nodes of a network that serve as substitutes for other nodes, as discussed in ¶ 31 above. Accordingly, a POSITA would have understood that Ikeda's low-level

agent is a client proxy.

### *x. Motivation to Combine Ikeda, Tsutsumitake, and Bird with Respect to Claims 17, 23, 29, and 35*

86. As discussed in ¶¶ 61-68above, a POSITA would have modified Ikeda with Tsutsumitake to cause a user terminal to register for updates in response to receiving a web page including an updateable element.

87. As discussed in ¶¶ 62-63 above, Tsutsumitake teaches that, upon receiving a web page with an event request, a client device sends an event request registration to a data server. Tsutsumitake further teaches that this registration step may be performed by a program that is incorporated in the web page, such as an Applet or a JavaScript program. *Tsutsumitake* (Ex. 1005), 10:30-33. As discussed in ¶¶ 32-33 above, a POSITA would have been familiar with Applets and JavaScript programs and would have understood that these programs are executed by the browser of a client device. Thus, Tsutsumitake teaches that providing a web page to a client (e.g., *providing the data representation to the client device*) includes providing an Applet or JavaScript program that is executed by the client browser to register an event request (e.g., *includes providing an activation module that is executed by the client device and that registers the live object with the routing network*).

88. A POSITA would have modified Ikeda with Tsutsumitake for the same reasons discussed in ¶¶ 46-60 above. Additionally, a POSITA would have modified Ikeda with Tsutsumitake's teachings because it is the use of Tsutsumitake's known technique for registering a client device for updates to information in a push system to improve Ikeda's similar push system in the same way (e.g., by simplifying operations of the user terminal which would have enabled use by a wide variety of user terminals). A POSITA would have recognized that using an Applet or JavaScript program to perform the registration step would have been beneficial because it would have been compatible with all client devices, enabling widespread use of the push service. Thus, a POSITA would have been motivated to implement Ikeda with such functionality to achieve this benefit.

89. A POSITA would have had a reasonable expectation of success in making the proposed modification for the same reasons discussed in ¶¶ 46-60 above. Additionally, a POSITA would have had a reasonable expectation of success in making this modification because Applets and JavaScript programs were well-known and common in the prior art at the time of the alleged

invention, as discussed in ¶¶ 32-33 above. Thus, implementation of the registration step using an Applet or JavaScript program would have been simple, straightforward, and well-within the skillset of a POSITA. Such a minor modification would not have required undue experimentation and would have yielded the predictable result of downloading an Applet or JavaScript program to the user terminal which would have performed the registration steps.

### *xi. Ikeda's Teachings Regarding Controlling Update Message Traffic Through Nodes of a Routing Network (claims [1.3], [7.4(a)])*

90. A POSITA would have understood that Ikeda's push system controls update message traffic through nodes of a routing network. Ikeda expressly teaches that reducing useless traffic is one benefit of the system. *Ikeda* (Ex. 1004), 21:49-59. Ikeda explains that the use of agents, which includes low-level agents (e.g., nodes having node types to which update messages are to be routed based on a mapping of categories of update messages to node types), reduces useless traffic on the Internet. *Id.* A POSITA would have understood that, because Ikeda's system only transmits information about updating to nodes registered to receive specific types of information, the amount of traffic through the nodes would be less than if information about updating were transmitted to all nodes regardless of registration.

### *xii. Motivation to Combine Ikeda, Tsutsumitake, and Bird with Respect to Claim 6*

91. As discussed in ¶¶ 46-60 above, as modified with Tsutsumitake, Ikeda's system would have been used to provide updates to web pages. And as discussed in ¶¶ 69-73 above, in the Ikeda-Tsutsumitake combination Ikeda's data servers would have sent information about updating elements of web pages. Tsutsumitake teaches that text representing numerical values (e.g., current value of electric current) are examples of updateable elements included in a web page. *Tsutsumitake* (Ex. 1005), 13:31-37, Fig. 7. Tsutsumitake expressly teaches that a displayed value (e.g., a visual representation) is changed. *Id.* at 13:58-65, Fig. 7. As discussed in ¶¶ 82-84 above, in the proposed combination, Ikeda's user terminals would have processed the information about updating to update a property of a web page element. Accordingly, in the proposed Ikeda-Tsutsumitake combination, Ikeda's update information would have updated a property of the live object, including having a direct effect on the visual representation of the live object on the web page. A POSITA would have modified Ikeda with Tsutsumitake and would have had a reasonable

expectation of success in making the proposed combination for the same reasons discussed in ¶ 68 above. Thus, Ikeda in view of Tsutsumitake renders obvious *the property of the live object has a direct effect on a visual representation of the live object in a data representation* (e.g., the text value, such as number, of the updateable element of the web page is changed).

## F. Ground 2: The Combination of Ikeda, Tsutsumitake, and Pearson

### *i. Overview of the Prior Art*

92. Ikeda and Tsutsumitake are discussed in ¶¶ 34-37 and 38-41 respectively, above.

93. I have reviewed, had input into, and endorse the discussions in the Overview of Pearson section of the Request, which I hereby incorporate by reference. *See* Request, Section I.F.1.b.

94. Pearson is analogous art to the '722 Patent because Pearson is in the same field of endeavor as the '722 Patent and because Pearson is reasonably pertinent to at least one problem with which the inventor of the '722 Patent was concerned.

95. As discussed in ¶ 36 above, the '722 Patent is in the field of "transferring information through digital networks." Pearson is in the same field of endeavor as the '722 Patent. Pearson "relates generally to computer network security, and more particularly relates to methods and systems for remotely monitoring the security status of a computer network and remotely configuring communication devices connected to a computer network." *Pearson* (Ex. 1007), 1:9-13. Pearson describes configuring firewalls which are devices used during Internet communications. *Id.* at 2:38-42. Indeed, Pearson also describes communications (i.e., transfers of information over the Internet) processed by the communication device. *Id.* at 18:41-49. Thus, Pearson relates to "transferring information through digital networks."

96. Pearson is also reasonably pertinent to at least one problem concerning the inventor of the '722 Patent. For example, one problem with which the inventor of the '722 Patent was concerned was techniques for categorizing data for the purposes of forwarding the data and an example provided for categorizing data was to assign all messages from the same input source to one category. *'722 Patent* (Ex. 1001), 18:52-54. Similarly, Pearson teaches categorizing and sorting data based on input source (i.e., port number). *Pearson* (Ex. 1007), 18:40-19:60.

### *ii. Motivation to Combine Ikeda and Tsutsumitake such that an input source provides a data representation, including a live object, to a*

***client device, different from the input source (Claims [14.1(a)], [20.1(a)], [26.1(a)], [32.1(a)])***

97. For the same reasons discussed in ¶¶ 46-60 above, a POSITA would have combined Ikeda and Tsutsumitake such that an input source provides a data representation, including a live object, to a client device, different from the input source, as recited by claims [14.1(a)], [20.1(a)], [26.1(a)], [32.1(a)].

***iii. Motivation to Combine Ikeda and Tsutsumitake such that in response to receiving the live object, the client device determines an object identifier of the live object and registers for updates of the live object (Claims [14.1(b)], [20.1(b)], [26.1(b)], [32.1(b)])***

98. For the same reasons discussed in ¶¶ 61-68 above, a POSITA would have combined Ikeda and Tsutsumitake such that such that in response to receiving the live object, the client device determines an object identifier of the live object and registers for updates of the live object, as recited by claims [14.1(b)], [20.1(b)], [26.1(b)], [32.1(b)].

***iv. Motivation to Combine Ikeda and Tsutsumitake such that an input source sends an update message to the routing network that identifies the live object and contains update data that updates a property of the live object (Claims [1.1], [7.2], [14.2], [20.2], [26.2], [32.2])***

99. For the same reasons discussed in ¶¶ 69-73 above, a POSITA would have combined Ikeda and Tsutsumitake such that an input source sends an update message to the routing network, as recited by claims [1.1], [7.2], [14.2], [20.2], [26.2], [32.2].

***v. Motivation to Combine Ikeda (as modified by Tsutsumitake) and Pearson such that a gateway device at the routing network is configured to identify a category of the update message based on the input source, to determine a node type to which the identified category maps, and to route the update message to the node, having the node type , at the routing network (Claims [1.2], [7.3], [14.3], [20.3], [26.3], [32.3])***

100. As discussed in ¶ 74 above, Ikeda teaches that a high-level agent (a gateway device at the routing network) determines a type of data represented by the information about updating (is configured to identify a category of the update message) and determines which nodes are assigned to handle such data types and routes the information about updating to the determined

node (determine a node . . . to which the identified category maps, and to route the update message to the node . . . at the routing network).

101. Pearson discloses that one method commonly employed in routing systems to categorize types of data was to look at a port number identifying a type of the data. *Pearson* (Ex. 1007), 18:41-19:18. As discussed in ¶¶ 96 above, and as acknowledged by *Pearson* (Ex. 1007, , 18:62-19:13), it was well-known that content providers (e.g., input sources), were assigned specific port numbers based on the types of information that they produce. Pearson teaches that this is determined by examining the header of a data packet. *Id.* at 18:41-19:18, 20:5-16. Accordingly, Pearson teaches determining a category of data based on an input source.

102. A POSITA would have modified Ikeda with Pearson's teachings of determining a data type based on port number. Specifically, Ikeda's high-level agent would have looked to a port number of an update message to determine a data type. For example, Ikeda teaches that news is one type of data that a user terminal may register to receive. *Ikeda* (Ex. 1004), 8:58-9:1. A POSITA would have been familiar with port numbers and would have known that certain port numbers were used for routing specific types of data. *Pearson* (Ex. 1007), 19:62-20:13 (citing RFC 1700); *see also* ¶¶ 29-30, *supra*. In the context of news data, a POSITA would have known that port 119 was used to route network news transfer protocol data, as discussed in ¶ 30 above. Accordingly, a POSITA would have modified Ikeda to categorize data from port 119 as news data. Once identified, the data would have been forwarded to the appropriate node as determined from the information maintained by the high-level agent (e.g., news-type low-level agents). Then the update message would have been routed to the low-level agents that are registered to receive that type of data, as taught by Ikeda.

103. A POSITA would have made the proposed combination for the following reasons: 1) the combination is the use of a known technique to improve similar devices in the same way; 2) known work in one field of endeavor may prompt variations of it for use in either the same field or a different one based on design incentives or other market forces if the variations are predictable to one of ordinary skill in the art; and 3) the combination is combining known prior art elements according to known methods to yield predictable results.

104. A POSITA would have been motivated to make the proposed combination because the proposed modification is the use of a known technique (e.g., Pearson's known technique of categorizing data based on port number) to improve similar devices (e.g., Ikeda's push service) in

the same way (e.g., providing a simple method of sorting data). A POSITA would have recognized that such a modification would have simplified Ikeda's push service forwarding. Pearson teaches that a port number is used to categorize and then forward data. Indeed, one data type taught by Ikeda for which a user may register is news data. *Ikeda* (Ex. 1004), 8:58-9:1. And a POSITA would have known that network news data was routed on port 119, as discussed in ¶ 30 above. A POSITA would have recognized that by looking at a port number for news, would have allowed for the categorization of all news data. Thus, because modification would have simplified Ikeda's data determination would have been simplified, a POSITA would have been motivated to modify Ikeda to determine a category of data by looking to the port number of the data.

105. A POSITA would have been motivated to make the proposed combination because known work in one field of endeavor (e.g., Pearson's known categorization of data based on source port number in security systems used for transferring information over the Internet) would have prompted variations of it for use in the same field (Ikeda's push system used for transferring information over the Internet) based on market forces (e.g., simplifying data transfer in Ikeda's push system) and such variations (e.g., categorizing data based on port number during website updates) would have been predictable to a POSITA (e.g., the modifications would have been simple and straightforward to a POSITA). As noted in ¶ 103above, a POSITA would have recognized the benefits to using an input source as a type of data (e.g., simplified Ikeda's push service forwarding), as taught by Pearson. As discussed in ¶¶ 36, 95above, both Pearson and Ikeda relate to transferring information over a digital network. Further, both Pearson and Ikeda describe similar systems including data servers sending information to user terminals. *Ikeda* (Ex. 1004), 1:12-36, 19:16-29; *Pearson* (Ex. 1007), 5:44-6:20, Fig. 1. Thus, the result of such a combination would have been predictable to a POSITA given the similarity between the systems and given the fact that a POSITA would have been familiar with Internet communications.

106. A POSITA would also have been motivated to make the proposed combination because it was well-known in the art to quickly and simply categorize data types based on a port number, as discussed in ¶¶ 29 above. Likewise, push systems, such as the system taught by Ikeda, were well-known systems in which users request updates to specific data types, as discussed in ¶¶ 80, 25 above. Thus, a POSITA would have modified Ikeda with Pearson's teachings to determine a category of data based on a port number (i.e., input source) and then route the data to an appropriate node for handling such data. Accordingly, a POSITA would have been motivated to

make the proposed combination because it is the combination of known prior art elements (e.g., Pearson's known technique of categorizing data based on port number with Ikeda's known push system) according to known methods (e.g., looking at the data header) to yield predictable results (e.g., determine a data type based on the port number).

107. A POSITA would have had a reasonable expectation of success in making the proposed modification given the similarities in the functions of the systems of Ikeda and Pearson. For example, Ikeda describes a system which route data over the Internet. *Ikeda* (Ex. 1004), 1:12-3:62, 19:16-29. Similarly, Pearson describes a system that routes data over the Internet. *Pearson* (Ex. 1007), 5:44-6:20, Fig. 1. A POSITA would also have had a reasonable expectation of success in making the proposed modification given the similarities in the architecture of the systems of Ikeda and Pearson. For example, in Ikeda's system, data types are determined by a high-level agent and forwarded to a low-level agent within a network based on the determined data type. *Ikeda* (Ex. 1004), 20:28-50. Likewise, in Pearson's system, data types are determined by a firewall and forwarded to another network node based on the determined data type. *Pearson* (Ex. 1007), 20:5-16. Finally, both Ikeda and Pearson perform forwarding based on registries that show the mapping of data types to destinations. *Ikeda* (Ex. 1004), 20:28-50; *Pearson* (Ex. 1007), 18:40-49. Accordingly, modification of Ikeda with Pearson's teachings would have been straightforward and well-within the skillset of a POSITA. Indeed, such a combination would have required only minor software modifications to Ikeda's system to determine the port number.

***vi. Motivation to Combine Ikeda and Tsutsumitake such that the user terminal processes information about updating to update an element of a web page (claims [1.7], [7.8(b)], [14.5], [20.5], [26.5], [32.5]***

108. For the same reasons discussed in ¶¶ 82-84 above, a POSITA would have combined Ikeda and Tsutsumitake such that the user terminal processes information about updating to update an element of a web page, as recited by claims [1.7], [7.8(b)], [14.5], [20.5], [26.5], [32.5].

***vii. Ikeda teaches providing the data representation to the client device includes providing the live object that causes the client device to register with a client proxy of the routing network (Claims 15, 21, 27, 33)***

109. For the same reasons discussed in ¶ 85-86 above, a POSITA would have combined Ikeda and Tsutsumitake such that *providing the data representation to the client device includes*

*providing the live object that causes the client device to register with a client proxy of the routing network*, as recited by claims 15, 21, 27, 33.

### *viii.* ***Motivation to Combine Ikeda, Tsutsumitake, and Bird with Respect to Claims 17, 23, 29, and 35***

110. For the same reasons discussed in ¶¶ 86-89 above, a POSITA would have combined Ikeda and Tsutsumitake such that *providing the data representation to the client device includes providing an activation module that is executed by the client device and that registers the live object with the routing network*, as recited by claims 17, 23, 29, and 35.

### *ix.* ***Ikeda's Teachings Regarding Controlling Update Message Traffic Through Nodes of a Routing Network (Claims [1.3], [7.4(a)])***

111. As discussed in ¶ 90 above, Ikeda's push system controls update message traffic through nodes of a routing network, as recited by claims [1.3], [7.4(a)].

### *x.* ***Motivation to Combine Ikeda, Tsutsumitake, and Bird with Respect to Claim 6***

112. For the same reasons discussed in ¶ 91 above, a POSITA would have combined Ikeda and Tsutsumitake such that the property of the live object has a direct effect on a visual representation of the live object in a data representation, as recited by claim 6.

## IV. CONCLUSION

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

Date: September 16, 2024

By: 

Dr. Michael Ian Shamos