UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE PATENT TRIAL AND APPEAL BOARD

FOLEON INC.,

Petitioner

v.

TURTL SURF & IMMERSE LIMITED

Patent Owner

IPR2025-01525
Patent No. 12,118,290

EXHIBIT 1003

DECLARATION OF DR. BENJAMIN B. BEDERSON
IN SUPPORT OF PETITION FOR INTER PARTES REVIEW OF
U.S. PATENT NO. 12,118,290

TABLE OF CONTENTS

EXHIBITS

Exhibit No.	Description
1001	U.S. Patent No. 12,118,290 B2 to Mason
1002	Prosecution History of U.S. 12,118,290
1004	*Curriculum Vitae* of Benjamin B. Bederson
1005	U.S. Patent Publication No. 2005/0080814 A1 to Hailey et al.
1006	International Publication No. WO 03/075191 A1 to Harrop et al.
1007	U.S. Patent No. 5,796,952 to Davis et al.
1008	U.S. Patent Publication No. 2004/0083474 A1 to McKinlay et al.
1009	U.S. Patent Publication No. 2005/0094205 A1 to Lo et al.
1010	U.S. Patent No. 4,712,174 to Minkler ("Minkler")
1011	U.S. Patent 5,956,687 to Wamsley et al. ("Wamsley")
1012	"How Users Read on the Web"
1013	*Information on RFC 3986*, RFC Editor,
1014	"How to use the Mail Merge feature in Word to create and to print form letters that use the data from an Excel worksheet," Microsoft
1015	"How People Read Online: The Eyetracking Evidence"
1016	Rong Chen, Anne Rose, and Benjamin B. Bederson. 2009. How people read books online: mining and visualizing web logs for use information. In Proceedings of the 13th European conference on Research and advanced technology for digital libraries (ECDL'09). Springer-Verlag, Berlin, Heidelberg, 364–369
1017	NYTimes, Google Acquires Urchin Software, March 29, 2005

Exhibit No.	**Description**
1018	Nataliia Bielova. 2017. Web Tracking Technologies and Protection Mechanisms. In Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security (CCS '17). Association for Computing Machinery, New York, NY, USA, 2607–2609.
1019	Tim Berners-Lee et al., *The World-Wide Web*. 37 Communications of the ACM 76 (1994).
1020	Archived webpage regarding Web Parameter Tampering (April 27, 2020).
1021	Archived webpage regarding Information exposure through query strings in url description (April 22, 2020).
1022	U.S. Patent 8,661,342 to Lu et al. ("Lu").

TABLE OF CLAIMS

Claim Limitation	Claim Language
1[PRE]	1. A system for creating an interactive document, the system comprising:
1[A]	a memory for storing a set of software instructions, and one or more processors configured to execute the set of software instructions to perform one or more operations comprising:
1[B1]	(a) generating an interactive document comprising one or more modular items, wherein the one or more modular items are pulled from a master document based at least in part on one or more personalization parameters,
1[B2]	and wherein the interactive document comprises a tracking feature to collect data indicative of user interaction with the interactive document;
1[C]	(b) creating a link between the interactive document and the master document, wherein the link comprises the one or more personalization parameters;
1[D1]	(c) rendering the interactive document with aid of a content renderer,
1[D2]	wherein the content renderer is configured to use the link to apply a change to the one or more modular items based on a change to the master document.
2	2. The system of claim 1, wherein the one or more modular items from the master document are embedded with an assembly rule.
3	3. The system of claim 2, wherein the one or more modular items in the interactive document are not embedded with the assembly rule.

4	4. The system of claim 1, wherein the one or more operations further comprise providing a graphical user interface (GUI) for receiving the one or more personalization parameters.
5	5. The system of claim 4, wherein the GUI is configured to display a preview of the interactive document upon receiving a user input indicative of at least one of the one or more personalization parameters.
6	6. The system of claim 5, wherein the preview comprises automatically navigating to at least one of the one or more modular items corresponding to the user input.
7	7. The system of claim 1, wherein the master document is stored in a pre-defined document schema.
8	8. The system of claim 7, wherein the master document is stored as structured data in a document database.
9	9. The system of claim 8, wherein the structured data further stores an assembly rule associated with the one or more modular items.
10[PRE]	10. A computer-implemented method for creating an interactive document, comprising:
10[A1]	(a) generating an interactive document comprising one or more modular items, wherein the one or more modular items are pulled

	from a master document based at least in part on one or more personalization parameters,
10[A2]	and wherein the interactive document comprises a tracking feature to collect data indicative of user interaction with the interactive document;
10[B]	(b) creating a link between the interactive document and the master document, wherein the link comprises the one or more personalization parameters;
10[C1]	(c) rendering the interactive document with aid of a content renderer,
10[C2]	wherein the content renderer is configured to use the link to apply a change to the one or more modular items based on a change to the master document.
11	11. The computer-implemented method of claim 10, wherein the one or more modular items from the master document are embedded with an assembly rule.
12	12. The computer-implemented method of claim 11, wherein the one or more modular items in the interactive document are not embedded with the assembly rule.
13	13. The computer-implemented method of claim 10, wherein the one or more operations further comprise providing a graphical user interface (GUI) for receiving the one or more personalization parameters.
14	14. The computer-implemented method of claim 13, wherein the GUI is configured to display a preview of the interactive document

	upon receiving a user input indicative of at least one of the one or more personalization parameters.
15	15. The computer-implemented method of claim 14, wherein the preview comprises automatically navigating to at least one of the one or more modular items corresponding to the user input.
16	16. The computer-implemented method of claim 10, wherein the master document is stored in a pre-defined document schema.
17	17. The computer-implemented method of claim 16, wherein the master document is stored as structured data in a document database.
18	18. The computer-implemented method of claim 17, wherein the structured data further stores an assembly rule associated with the one or more modular items.

I. INTRODUCTION

1. I, Benjamin B. Bederson, offer this declaration in support of the Petition for Inter Partes Review of U.S. Patent No. 12,118,290 ("the '290 Patent"). I am over the age of 18, competent to make this declaration, and have personal knowledge of the facts set forth below. If called to testify, I could and would testify honestly, under oath, to the matters set forth herein.

2. I am not an attorney. I have not been asked to offer any legal opinions. I have been informed and understand the law to be applied for determining invalidity, which I explain in places below. I have applied the law told to me in developing my technical opinions in this Declaration.

3. All statements made in this declaration regarding the disclosures and teachings of the prior art references and/or the state of the relevant technology are from the perspective of a person of ordinary skill in the art ("POSITA").

II. BACKGROUND AND QUALIFICATIONS

4. My curriculum vitae, which includes a more detailed summary of my background, experience, and publications, is attached as Exhibit 1004.

5. I received a Bachelor of Science degree in Computer Science with a minor in Electrical Engineering from Rensselaer Polytechnic Institute ("RPI") in 1986. I received a Master of Science degree and a Ph.D. in Computer Science from New York University ("NYU") in 1989 and 1992, respectively.

6. Since 1998, I have been a Professor of Computer Science at the University of Maryland ("UMD"), where I have had joint appointments at the Institute for Advanced Computer Studies and the College of Information Studies (Maryland's "iSchool") and am currently Professor Emeritus. I was also Associate Provost of Learning Initiatives and Executive Director of the Teaching and Learning Transformation Center from 2014 to 2018. I am a member and previous director of the Human-Computer Interaction Lab ("HCIL"), the oldest and one of the best known Human-Computer Interaction ("HCI") research groups in the country.

7. I am also co-founder and co-director of the International Children's Digital Library ("ICDL"), a web site launched in 2002 that provides the world's largest collection of freely available online children's books from around the world with an interface aimed to make it easy for children and adults to search and read children's books online. I was also co-founder and Chief Scientist of Zumobi, Inc. from 2006 to 2014, a Seattle-based startup that is a publisher of content applications and advertising platforms for smartphones. I was also co-founder and prior Chief Technology Officer of Hazel Analytics from 2014 to 2023, a data analytics company. In addition, I have for more than 25 years consulted for numerous companies in the area of user interfaces, including EPAM, Lockheed Martin, Logitech, Microsoft, NASA Goddard Space Flight Center, the Palo Alto Research Center, and Sony.

8. For more than 30 years, I have studied, designed, and worked in the field of computer science and human-computer interaction. My experience includes 30 years of teaching and research, with research interests in human-computer interaction and the software and technology underlying today's interactive computing systems. This includes the design and implementation of user interfaces and software applications using a wide range of technologies including web servers and websites, databases, embedded systems, and mobile devices, including smart phones and PDAs.

9. At UMD, my research has been in the area of HCI, a field that relates to the development and understanding of computing systems to serve users' needs. Researchers in this field are focused on making universally usable, useful, efficient, and appealing systems to support people in their wide range of activities. My approach is to balance the development of innovative technology that serves people's practical needs. I have combined my hardware and software skills throughout my career in HCI research, building various interactive electrical and mechanical systems that couple with software to provide an innovative user experience. Example systems that I have built include Pad++ (1994 software for zoomable user interfaces), KidPad (1997 software to support children collaboratively writing stories), PadPrints (1998 software to graphically show interactive web browsing histories), PhotoMesa (2001 software for end users to

browse personal photos), Pocket PhotoMesa (2004 software for browsing of personal photos on a mobile device), DateLens (2002 software for end users to use their mobile devices to efficiently access their calendar information), LaunchTile (2005 "home screen" software for mobile devices to allow users to navigate apps in a zoomable environment), SpaceTree (2001 software for end users to efficiently browse very large hierarchies), ICDL (as described above), and StoryKit (a 2009 iPhone app for children to create stories).

10. Pad++ included a custom web browser that I built from scratch and distributed as open source software at the Pad++ web site. That web browser was described in numerous publications including "Pad++: A Zoomable Graphical Interface for Exploring Alternate Interface Physics."[1] I implemented a rendering engine from scratch for that web browser which supported the version of HTML available at the time (HTML 1.0) as well as an embedded scripting language. That web browser included the ability to display embedded web pages that could be zoomed along with the rest of the pages. The following images are from that custom web browser that I built, which used a "zoomable user interface" so one page could be shown larger while other recently visited pages would be displayed smaller.[2]

[1] Bederson, B. B. & Hollan, J. D. (1994). Pad++: A Zooming Graphical Interface for Exploring Alternate Interface Physics, ACM Conference on User Interface Software and Technology, 17-26.
[2] Bederson, B. B. et al. (1996). Pad++: A Zoomable Graphical Sketchpad for Exploring Alternate Interface Physics, 7, Journal of Visual Languages & Computing.

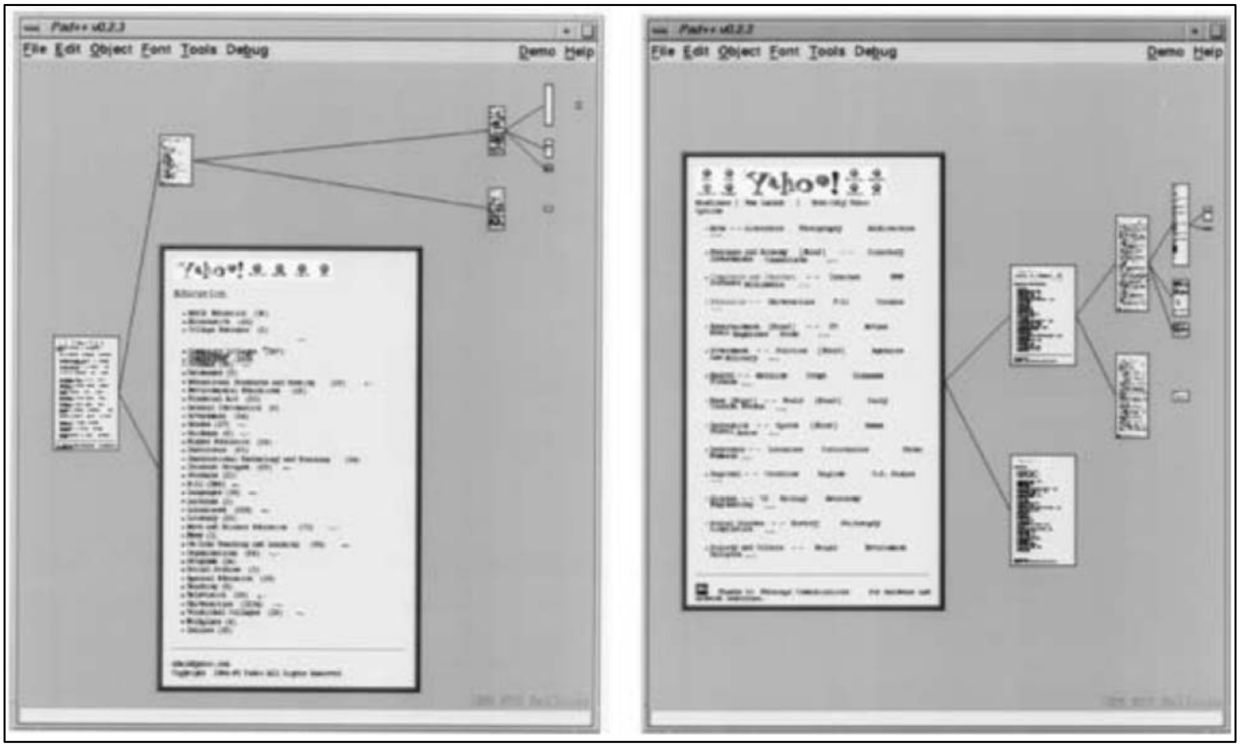


Figure 5 from "Pad++: A Zoomable Graphical Sketchpad for Exploring Alternate Interface Physics"

11. By 1998, my team had combined Pad++ with Netscape Navigator (a standard web browser of the time) to offer a graphical web history, visually showing what pages a user had visited with an application we called "PadPrints."[3] We did this by implementing a web "proxy," which was a standard way at the time to modify the built-in behavior of web browsers. Some proxies work by intercepting all web traffic coming from and going to a web browser. Once intercepted, the proxy can do whatever it wants with the traffic, caching it for better performance, simply

[3] Bederson, B. B. et al. (1998). PadPrints: Graphical Multiscale Web Histories. In Proceedings of the 11th Annual ACM Symposium on User Interface Software and Technology (UIST '98). Association for Computing Machinery, New York, NY, USA, 121–122. https://doi.org/10.1145/288392.288582

passing the traffic through and monitoring it, modifying the data itself, or holding that connection and opening new connections. PadPrints' proxy worked by monitoring the traffic and letting PadPrints know what the user was doing so PadPrints could display the web history.

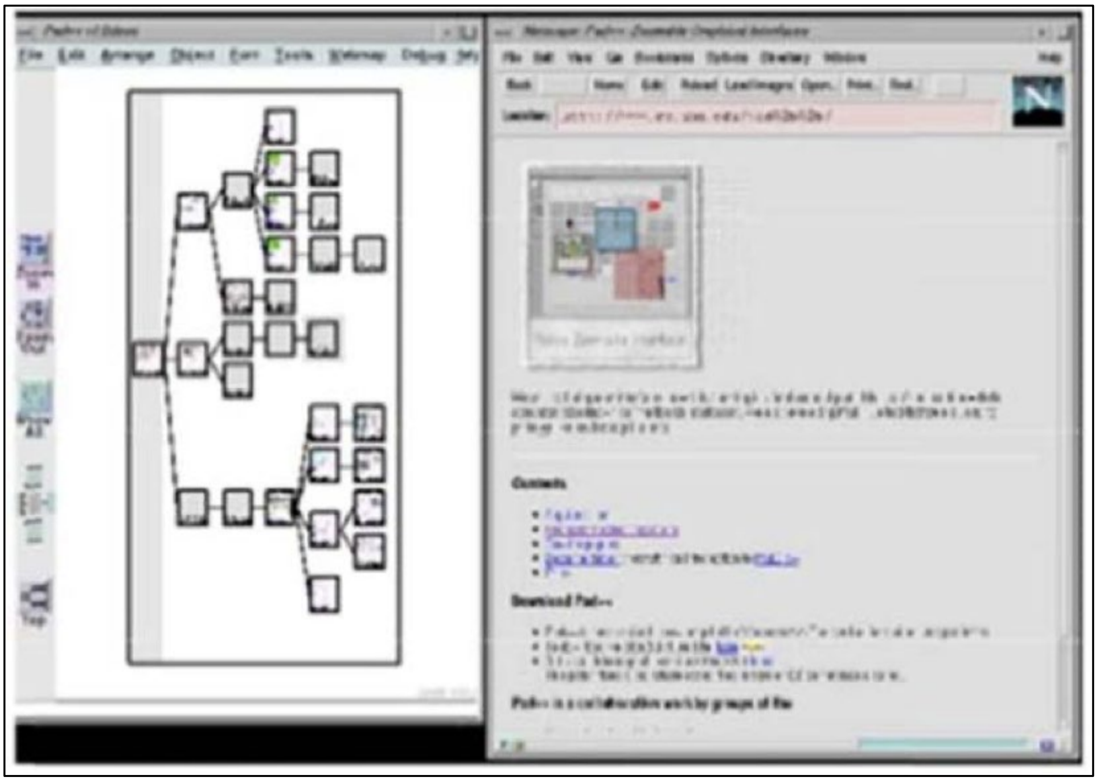


Figure 1 from PadPrints paper

12. At the International Children's Digital Library (ICDL), I have since 2002 been the technical director responsible for the design and implementation of the web site, www.childrenslibrary.org (originally at www.icdlbooks.org). In particular, I have been closely involved in designing the user interface as well as the software architecture for the web site since its inception in 2002.

Screenshot taken February 19, 2017 by Benjamin B. Bederson from http://www.childrenslibrary.org/icdl/SimpleSearchCategory?ids=25&langid=&pnum=1&cnum=1& text=&lang=English&ilang=English

13. Beginning in the mid-1990s, I have been responsible for the design and implementation of numerous other web sites in addition to the ICDL. For example, I designed and built my own professional web site when I was an Assistant Professor of Computer Science at the University of New Mexico in 1995 and have continued to design, write the code for, and update both that site (which I moved to the University of Maryland in 1998, currently at http://www.cs.umd.edu/~bederson/) as

well as numerous project web sites, such as Pad++, http://www.cs.umd.edu/hcil/pad++/.

14. I started a new web-based project in 2009 to bring people together on a website to collaboratively translate text for children's books from one language to another. This resulted in numerous publications, including a 2011 paper[4] that described a second generation of this work. In the numerous versions of this project, we used several web-based collaboration techniques, including by synchronous and asynchronous collaboration – allowing people to see the work of others on their own views of our website either immediately as others were doing the work (synchronously) or later after they had completed the work (asynchronously).

[4] Bederson, B. B. et al. (2011). MonoTrans2: a new Human Computation System to Support Monolingual Translation. In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '11). Association for Computing Machinery, New York, NY, USA, 1133–1136. https://doi.org/10.1145/1978942.1979111

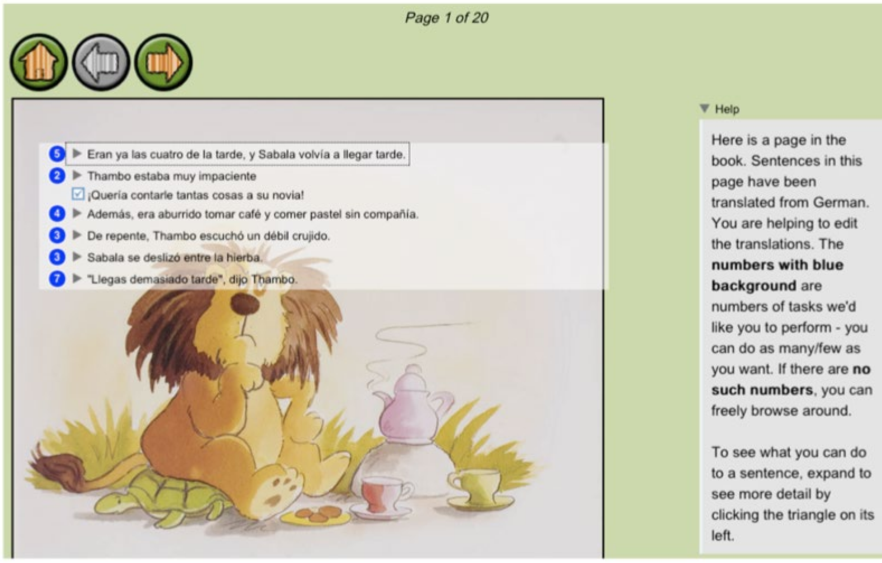


Figure 1. The MonoTrans2 UI for the target language speakers. Notice 1) the number in a blue circle indicating tasks before sentences; and 2) the contextual help on the right. The background image is a scan of the actual book (with original text removed) to provide context.

15. My work has been published extensively in more than 160 technical publications, and I have given about 100 invited talks, including 9 keynote lectures. I have won a number of awards including the Brian Shackel Award for "outstanding contribution with international impact in the field of HCI" in 2007, and the Social Impact Award in 2010 from Association for Computing Machinery's ("ACM") Special Interest Group on Computer Human Interaction ("SIGCHI"). ACM is the primary international professional community of computer scientists, and SIGCHI is the primary international professional HCI community. I have been honored by both professional organizations. I am an "ACM Distinguished Scientist," which "recognizes those ACM members with at least 15 years of professional experience

and 5 years of continuous Professional Membership who have achieved significant accomplishments or have made a significant impact on the computing field." I am a member of the "CHI Academy," which is described as follows: "The CHI Academy is an honorary group of individuals who have made substantial contributions to the field of human-computer interaction. These are the principal leaders of the field, whose efforts have shaped the disciplines and/or industry, and led the research and/or innovation in human-computer interaction." The criteria for election to the CHI Academy are: (1) cumulative contributions to the field; (2) impact on the field through development of new research directions and/or innovations; and (3) influence on the work of others. I have received two "Test of Time" awards from IEEE InfoVis for my 2001 and 2002 work on visualizing hierarchies, including those containing digital photos.[5] I received the Janet Fabri Memorial Award for Outstanding Doctoral Dissertation for my Ph.D. work in robotics and computer vision.

16. I have appeared on radio shows numerous times to discuss issues relating to user interface design and people's use and frustration with common technologies, web sites, and mobile devices. My work has been discussed and I have been quoted by mainstream media around the world over 120 times, including by

[5] https://ieeevis.org/year/2021/info/awards/test-of-time-awards; https://ieeevis.org/year/2022/info/awards/test-of-time-awards.

the New York Times, the Wall Street Journal, the Washington Post, Newsweek, the Seattle Post-Intelligencer, the Independent, Le Monde, NPR's All Things Considered, New Scientist Magazine, and MIT's Technology Review.

17. I have designed, programmed, and publicly deployed dozens of user-facing software products that have cumulatively had millions of users. I am the co-inventor of 14 U.S. patents and 20 U.S. patent applications.

III. SUMMARY OF MATERIALS REVIEWED AND CONSIDERED

18. All of the opinions contained in this Declaration are based on the documents I reviewed and my knowledge and professional judgment. In forming the opinions expressed in this Declaration, while drawing on my experience and knowledge regarding graphical user interfaces, I read and considered the '290 Patent, its prosecution history, as well as any other material referenced herein.

19. My opinions are guided by my appreciation of how a person of ordinary skill in the art ("POSITA") would have understood the claims of the '290 Patent at the time of the filing, which I have been asked to assume is July 6, 2020.

IV. RELEVANT PATENT LAW AND LEGAL STANDARDS

20. I am not an attorney. I understand that in writing this Declaration I am obliged to follow existing applicable law. I have therefore been asked to apply the following legal principles to my analysis, and I have done so.

A. *Date of Invention*

21. I have been asked to use the date of the filing of the earliest application to which priority is claimed (*i.e.*, July 6, 2020) as the date of invention for purposes of my analysis.

B. *Person of Ordinary Skill in the Art (POSITA)*

22. I understand that my assessment of the claims of the ’290 Patent must be undertaken from the perspective of what would have been known or understood by a person having ordinary skill in the art, reading the ’290 Patent on its relevant filing date and in light of the specification and file history of that patent. I often refer to such a person as a POSITA in this Declaration.

23. I am informed and understand that various factors can be considered in determining a POSITA. I am informed and understand that those factors include: (1) the educational level of the inventor(s); (2) the type of problems encountered in the art; (3) prior art solutions to those problems; (4) the rapidity with which innovations are made; (5) sophistication of the technology; and (6) education level of active workers in the field.

24. I understand that Foleon, Inc. proposed that a POSITA in the field of the ’290 Patent as of July 6, 2020, would have been someone knowledgeable about and familiar with familiar with electronic document assembly systems and methods pertinent to the ’290 Patent. A POSITA would have had a bachelor’s degree in

Computer Science or equivalent training, and approximately two years of programming experience developing client/server systems that create and render documents. Lack of work experience can be remedied by additional education, and vice versa.

25. I agree with this level of ordinary skill for the ’290 Patent. Based on my experience, I understand and know of the capabilities of a person of this skill level as of the time the ’290 Patent was filed. I am familiar with how a POSITA would have understood and used the terminology found in the ’290 Patent at the time of its filing, and with the state of the art at that time.

26. My analysis and opinions presented herein regarding the claims of the ’290 Patent have been and are based on the perspective of a POSITA as defined above as of July 6, 2020, the assumed priority date for the ’290 Patent. As described above and as shown in my CV, I have extensive experience in the field of electronics in the relevant timeframe.

C. *Claim Construction Standard*

27. In making this Declaration, I have been asked to review the terms in the ’290 Patent claims to determine what a POSITA would have understood those terms to have meant at the time of the alleged invention. In addition, I have been informed that, in an *inter partes* review proceeding, the claim terms under consideration are to be construed using the same claim construction standard used in a civil action

under 35 U.S.C. § 282(b), which is articulated in *Phillips v. AWH Corp.*, 415 F.3d 1303 (Fed. Cir. 2005) (en banc), and its progeny. I further understand that under the *Phillips* standard, the claim terms are given their ordinary and customary meaning, as would be understood by a POSITA at the time of filing, *i.e.* July 6, 2020, consistent with the specification and the prosecution history. My opinions regarding the meaning of claim terms of the ’290 Patent are based on the construction of the claim terms using the *Phillips* standard in view of the specification and the prosecution history of the ’290 Patent.

D. *Anticipation*

28. I understand that to be valid, a patent claim must be novel, and that a claim is invalid if it is anticipated by a single prior art reference. I further understand that a reference anticipates if it discloses each and every element of the claim and enables a POSITA to make and use the claimed invention without undue experimentation.

E. *Obviousness*

29. I also understand that even if a claim of a patent is not anticipated, that claim is still invalid if the differences between the claimed subject matter and the prior art are such that the subject matter as a whole would have been obvious to a POSITA at the time the invention was made.

30. I understand further that the condition "at the time the invention was made" in an obviousness determination is imposed to rule out impermissible hindsight. Accordingly, permissible reasoning takes into account only knowledge which was within the level of ordinary skill in the art at the time the claimed invention was made and does not include knowledge gleaned from the applicant's disclosure in the patent under analysis. I also understand that an opining expert is required to analyze prior art from the perspective of a POSITA and may not simply provide his or her own personal conclusions.

31. I also understand that an obviousness determination includes several factual inquiries, including: (1) determining the scope and content of the prior art; (2) ascertaining the differences between the claimed invention and the prior art; (3) resolving the level of ordinary skill in the pertinent art; and (4) taking into consideration any secondary indicia (objective evidence) of non-obviousness.

32. I also understand that such objective evidence of non-obviousness may include: (1) a long felt but unsolved need that was satisfied by the claimed invention; (2) commercial success attributable to the claimed invention; (3) unexpected results achieved by the claimed invention; (4) praise by experts of the claimed invention with factual support; (5) taking of licenses under the patent by others for reasons related to the alleged non-obviousness of the claimed invention; (6) the failure of others; (7) the teaching away of others; (8) skepticism by experts; and (9) evidence

that competitors in the marketplace have copied the invention instead of using the prior art. I also understand that, in order to establish non-obviousness, there must be a demonstrated relationship, or nexus, between any such secondary indicia and the claimed invention, *i.e.*, there must be objective evidence of non-obviousness attributable to the claimed invention.

33. I further understand that near simultaneous invention by two or more equally talented inventors working independently may or may not be an indication of obviousness when considered in light of all the circumstances.

34. I understand that a conclusion of obviousness can be based on a combination of multiple prior art references. I understand that rationales that may support a conclusion of obviousness include:

35. (A) Combining prior art elements according to known methods to yield predictable results;

36. (B) Simply substituting one known element for another to obtain predictable results;

37. (C) Using known techniques to improve similar devices (methods, or products) in the same way;

38. (D) Applying a known technique to a known device (method, or product) ready for improvement to yield predictable results;

39. (E) Using an obvious to try solution, *i.e.*, choosing from a finite number of identified, predictable solutions, with a reasonable expectation of success;

40. (F) Applying known work in one field of endeavor to yield 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;

41. (G) Applying 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 disclosures to arrive at the claimed invention.

42. I further understand that an analysis of obviousness should recognize that market demand, rather than scientific literature, often drives design trends.

43. I also understand that if a technique has been used to improve one device, and if a POSITA would recognize the technique would improve similar devices in the same way, then using that technique would be obvious to a POSITA unless actual application of that technique is beyond his or her skill.

44. I also understand that practical and common-sense considerations should guide an analysis of obviousness, because familiar items may have obvious uses beyond their primary purposes. I further understand that applying common sense does not require a specific hint or suggestion in a particular reference. Rather, applying common sense requires a reasoned explanation that avoids conclusory generalizations and false deductions.

45. I also understand that, when addressing a problem, a POSITA will often be able to fit together the disclosures of multiple publications. In this regard, I understand that an analysis of obviousness may consider inferences and creative steps that a POSITA would normally employ.

46. I also understand that a particular combination of prior art may be proven obvious merely by showing that it was obvious for a POSITA to try that combination. For example, when there is a design need or a market pressure to solve a problem, and/or there exists a finite number of identified, predictable solutions with a reasonable expectation of success, a POSITA has good reason to pursue the known options within his or her technical grasp. I understand that if this leads to anticipated success, it is likely that the result is not of innovation but of ordinary skill and common sense.

47. I also understand that a combination of familiar elements according to known methods is presumed to be obvious when that combination does no more than yield predictable results. Work known in one field of endeavor may prompt variations of that work for use in either the same field or a different field, based on design incentives and other market forces. If a technique has been used to improve some device, and a POSITA can implement a predictable variation of that technique, then that variation is likely unpatentable.

48. I further understand that to be proper for use in an analysis of obviousness, a reference must teach art that is analogous to the claimed invention. I also understand that under a correct analysis, any need or problem known in the field of endeavor at the time of filing and addressed by the claimed invention can provide a reason for combining the elements in the manner claimed.

49. I understand that a claim can be obvious in light of a single reference, without the need to combine references, if the elements of the claim that are not found explicitly or inherently in that single reference can be supplied by the common sense or technical knowledge of a POSITA. For example, combining two embodiments disclosed adjacent to each other in a prior art patent likely does not require a leap of inventiveness.

50. I also understand that a claimed invention may be obvious if it merely involves simple substitution of one known element for another to obtain predictable results.

51. I understand further that, in an analysis of obviousness, two items of prior art need not be like two puzzle pieces that must fit together perfectly. For example, a claimed invention may be found obvious if a POSITA would view rearrangement as an obvious matter of design choice.

52. I also understand that an analysis of obviousness requires a comparison between the properly construed claim language and the prior art on a limitation-by-limitation basis.

F. *Standard of Proof*

53. I understand that the standard to prove unpatentability in an inter partes review proceeding is by a preponderance of the evidence, which means more likely than not.

V. TECHNOLOGICAL BACKGROUND

A. *Modular Document Generation*

54. Creating documents from modular or variable content has been known for decades. For example, in 1987, U.S. Patent No. 4,712,174 to Minkler describes "[a] computer based system for generating text from a predetermined data base, in either prose or poetic form, in response to a plurality of input data provided by an operator in an interactive mode of operation with the computer." EX1010-Minkler, Abstract. Even four decades ago, Minkler describes that "it is well known in the prior art to generate textual material by combining prewritten sentences, paragraphs or sections of text retrieved from data base files." EX1010-Minkler, 1:12-13. In particular, Minkler's system "provide[s] a pseudorandom computer printout of text in response to a plurality of input data items related to the intended recipient and/or sender of that text." EX1010-Minkler, 2:17-20. In the 90's systems for generating

documents from variable data related to a specific field were also known. For example, U.S. Patent 5,956,687 to Wamsley et al. ("Wamsley") describes a system that "automatically generate[s] a demand letter and calculate[s] settlement amounts from information gathered in the record during pursuit of [a] claim." EX1011-Wamsley, abstract.

55. As described in more detail below, both Hailey and Harrop, both published in the early 2000s, also describe systems for generating documents from variable or modular data or content. For example, Harrop teaches "[a] document assembly or document automation system includes an assembler for generating an instance document on the basis of a source document and one or more logic source documents referenced by the source document." EX1006-Harrop, abstract. Similarly, Hailey teaches that "[d]ocument production systems that insert customer-specific content into a pre-fabricated document are known." Hailey's system, among other things, "create[es] documents by assembling document components and applying precedence and rules to create a desired document." EX1005, ¶2.

56. More generally, a range of products in the early 2000s generated documents from templates. For example, many word processing programs supported a "mail merge" feature which used a word processing document as a template with fields marked with a delimiter. Those fields would get filled in using data from a data source as a spreadsheet or database generating a copy of the filled in document

for each record in the data source. I used this feature myself in Microsoft Word in the early 2000s. According to Microsoft's website (EX1014), Microsoft introduced "mail merge" support into Microsoft Word in 2003. "In Microsoft Office Word 2003 and in earlier versions of Word, point to Letters and Mailings on the Tools menu, and then click Mail Merge Wizard.' Additionally, The Microsoft Word mail merge feature is described in some detail in U.S. Patent 8,661,342 to Lu et al. ("Lu"). As illustrated in FIG. 4 and described: "A mail merge type option is displayed to allow the user to select the type of mail merge being created. A few non-limiting examples of the type of mail merge can include a letter, envelope, brochure, label, email, and so on. A template selection option is also displayed which allows the user to select a blank document or an existing mail merge template to use. The term 'mail merge template' as used herein is meant to include a template that contains wording, macros and/or other settings to use as a starting point when creating a new mail merge document. Record merge options are displayed to allow the user to select which records should be merged, such as the selected records on the current page, all records on the current page, or all records on all pages. Data field option is displayed to allow the user to specify which data fields should be included in the mail merge operation." EX1022, 5:17-32. Lu describes that the invention can be implemented in Microsoft Word: "In one implementation, one or more of the techniques described herein can be implemented as features within a word processing program such as MICROSOFT® Office Word". EX1022, 3:2-5, Fig. 4.

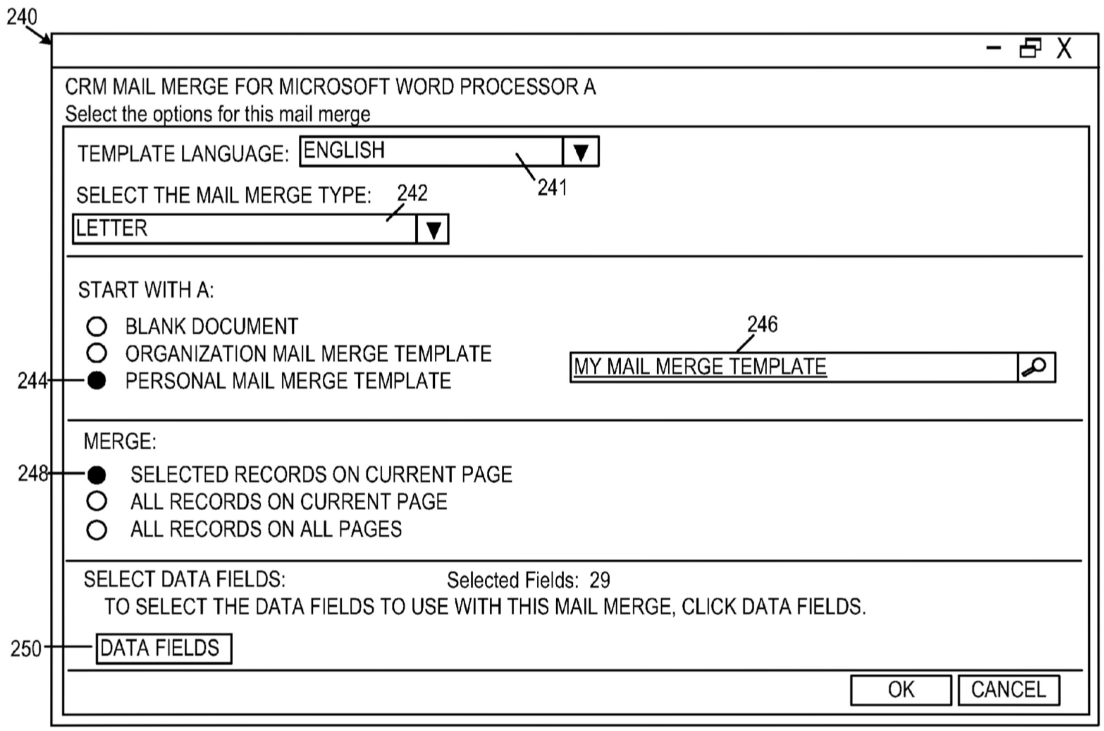


FIG. 4

57. As recognized by Minkler, Wamsley, Hailey, Harrop, and others, the use of assembly rules to instruct or influence which content or text to include in a generated document. Minkler teaches an algorithm with a series of assembly rules that direct which words, how many syllables, which pronouns, etc. should be included in the generated poem. *See* EX1010-Minkler, 16:53-17:32. Wamsley teaches generating a demand letter with rules for which paragraphs and information to include in the letter. EX1011-Wamsley, 23:6-39. As discussed in sections VII

and VIII below, Harrop and Hailey also describe using assembly rules to determine what content to include in a generated document.

B. *User Tracking*

58. People have been tracking how users read and interact with websites practically since the beginning of the web. Back in 1997, there was an article by well-known usability expert Jakob Nielsen (who is someone I worked with in the mid-1990s when we both worked at Bellcore.) EX1015. The article analyzed how people use the web—recognizing that they are much more likely to "scan" then fully read webpages. Given the economic impact of the web and amount of money involved in advertisements, paid content, and services, it isn't surprising that people put quite a bit of effort into tracking how webpages are used. Nielsen's company actually went on to study and offer multiple editions of paid reports providing detailed analysis of this kind of data. *See, e.g.*, EX1015.

59. Given the importance of this data, people don't rely only rely on experts to provide such tracking and analysis – they do it themselves. I know this because I did this myself to understand how people were reading books in our International Children's Digital Library. I wrote a paper in 2009 that described how we "analyze and track what we call *book reading sessions* (BRS) with reasonable accuracy." EX1016, 2. The paper reported all kinds of information about tracking book reading

including number of books read, the percentage of books read, the time spent reading, etc.

60. More generally, there are several commonly used ways to track how people use the web. At the simplest level, there are "web logs". These are logs of every URL request to a web server that are kept by essentially all web server systems. These can then be analyzed to determine which pages are being viewed, from where, for how long, by how many different people, etc. There are numerous services that support such analysis of web logs. One such system is the freely available Google Analytics, which was created in 2005 when Google acquired Urchin Software. EX1017. I remember this since I immediately started using Google Analytics on several websites that I was responsible for, including the International Children's Digital Library.

61. People also use a range of other technologies to track web use. As described in this 2017 summary of a tutorial about web tracking technologies, people use a wide variety of tracking technologies "from simple cookies to advanced cross-device fingerprinting." EX1018, Abstract.

62. As discussed below, Davis, a patent filed in 1997, describes the tracking of how people use documents including monitoring "time, keyboard events, mouse events, and the like, in order to track choices and selections made by a user". EX1007, abstract. Davis describes performing this tracking "by use of a tracking

program embedded in the resource and executable by a client." EX1007, 4:6-8. Davis explains that "the tracking program is embedded in an HTML document (such as a Web site, a Web page, or part of a Web page–*e.g.* an 'ad banner'). EX1007, 5:14-17. It also describes that "the tracking program of the present invention may be implemented in the JAVA programming language." EX1007, 10:48-50.

C. *URL Parameters*

63. The term "uniform resource identifier" or "URI" has been deeply rooted in industry terminology for decades. The term was popularized in the early 1990s by Tim Berners-Lee—an individual known as the founder of the World Wide Web (the "Web")—and several other internet trailblazers. EX1019. URLs are the specific kind of URI that have come to be used to locate documents using the Web. As formalized in the 2005 internet standard "RFC3986 (2005)", "[t]he term 'Uniform Resource Locator' (URL) refers to the subset of URIs that, in addition to identifying a resource, provide a means of locating the resource by describing its primary access mechanism (*e.g.*, its network 'location')." EX1013, 7.

64. The standard defines URLs to include an optional set of additional keyword pairs formally called a "query", and is commonly described as URL parameters. It gives an example labeling the various components. EX1013, 16. In this example, the query ("name=ferret") shows a URL parameter where the parameter name is "name" and the value is "ferret".

```
foo://example.com:8042/over/there?name=ferret#nose
\_/   \______________/\_________/ \_________/ \__/
 |           |            |            |        |
scheme     authority       path        query   fragment
```

65. In my experience, these URL parameters are ubiquitous throughout the web and have been for decades. As I showed above in the Qualifications section, I used them myself in the International Children's Digital Library to specify queries and to identify particular books. In that example (which I duplicate here), I used the several URL parameters including "pnum" to have a value of 1 (meaning to search for the first page), and "lang" to have a value of "English" (meaning to search for books in English).

Screenshot taken February 19, 2017 by Benjamin B. Bederson from http://www.childrenslibrary.org/icdl/SimpleSearchCategory?ids=25&langid=&pnum=1&cnum=1& text=&lang=English&ilang=English

66. The simplicity of URL parameters lies in their being plain text, and being able to be added onto an existing URL. In my experience, these reasons help to make them ubiquitous. They are used throughout the web to access search engines, databases, e-commerce sites, newspapers, and service providers of all kinds.

VI. OVERVIEW OF THE ’290 PATENT

67. The ’290 Patent describes a document assembly system and method for automatically assembling digital content or documents based on a set of modular content and parameters corresponding to the ultimate reader of the assembled document. *See, e.g.*, EX1001, 1:38-44 (“In particular, methods and systems of the present disclosure may provide automation and modular features allowing for creation of personalized content (*e.g.*, report, brochures, digital magazines, product guides, sales playbooks, sales proposals, whitepapers, newsletters, company handbooks, annual reports, etc.) with reduced time and improved personalization.”); 2:10-20 (“In some embodiments, the value of the one or more parameters is specified by a user via a user interface. In some cases, the user interface displays one or more personalization fields that are automatically generated based on the first set of rules. For instances, at least one of the one or more personalization fields is generated based on a second set of rules associated with the second modular item. In some instances, the one or more processors are further configured to: upon receiving an input in a given personalization field, display at least a portion of the content from

the first document that is related to the given personalization field."). The patent describes that the assembled interactive documents "may include a variety of types of electronic documents such as report[s], brochures, digital magazines, case studies, product guides, sales playbooks, sales proposals, whitepapers, newsletters, company handbooks, annual reports, presentations, and various others that may be utilized in various other fields." Id., 4:44-49.

68. The patent describes these generated documents as being "interactive" documents and broadly defines "interaction." For example, the patent describes interaction when a user has "read, edited, shared, [and/or] saved" a document. EX1001, 23:17-31 (" In some cases, the interaction data may be analyzed to extract metrics and usage analytics measuring the degree to which slides/pages were provided or made available to and viewed by customers (*e.g.*, the number of times a content item was pitched to customers), to extract performance metrics measuring the ***degree to which readers have engaged with (e.g., read, edited, shared, saved)*** the interactive content (*e.g.*, ***the dwell time on an item, such as the amount of time that the content was displayed in a content viewer*** (*e.g.*, web browser or other application), to extract impact analytics indicative of the content's impact on business metrics (*e.g.*, average revenue per pitch of a content item that may be obtained from other systems, such as customer relationship management (CRM) systems, marketing automation systems, etc.), and various other analytics."). In

another example, interaction amounts to a user "brows[ing], filter[ing], or select[ing] a content item (*e.g.*, page, video) for viewing." EX1001, 25:17-19 ("the personalized document may include interactive features for a reader to actively interact with one or more content items. For instance, ***a reader may be permitted to browse, filter, or select a content item (e.g., page, video) for viewing***."). Thus, the patent broadly defines "interaction" to include even passive activities like selecting a document to read or watching a video.

69. Related to the discussion of use "interaction," the patent describes collection of "data indicative of" a user's interaction with a document. EX1001, 23:16-17 ("For instance, data may be collected indicative of user interaction with the document.") However, despite broadly describing various categories or types of interactions that the system looks for, the patent does not describe what the collected data looks like. That is, the patent does not describe whether the collected data is a particular type of data, stored in a specific document, or otherwise has a particular form. Considering that a POSITA would be well aware of the various ways data can be stored in an electronic form, it is not surprising that the patent does not describe these details and, instead, appears to rely on a POSITA's knowledge to fill gaps in the patent's disclosure.

70. Even without describing the particularities of the collected data, the patent describes using the collected data to analyze "usage analytics", "performance

metrics," "impact analytics," and "various other analytics" to "personalize further interactive content/document[s]." EX1001, 23:17-32.

71. Regarding the creation of personalized interactive documents, the '290 Patent describes using a "content creation and management system 121" which "permits users (*e.g.*, creators, personalizers, analyzers, administers, etc.) to create content and content items, generate a set of rules associated with a content item, customize a document[,] and manage various aspects of the platform." EX1001, 8:20-25. Content items—also referred to as "modular content items"—may be "a page, a slide, a video, an image, a chart, [and/or] a text file…." EX1001, 15:51-55 ("A content item such as a page, a slide, a video, an image, a chart, a text file of a document may be stored with one or more different representations (*e.g.*, mobile rendering, desktop rendering, rendering in a browser, rendering in a desktop application, etc.").

72. The modular content items "may be automatically selected from one or more documents to be assembled into a personalized document assembly based on pre-determined assembly rules" which "define the conditions to meet in order to include a content item into a derivative/output document." EX1001, 5:55-60. The patent refers to these "conditions" defined in the assembly rules as "personalization parameters." EX1001, 5:60-67, 20:1-9. For example, the personalization parameter may require a certain geographic location (*e.g.*, state) or revenue for the content

associated with the rule to be displayed in the generated document. EX1001, 16:48-51.

73. The patent describes that the assembly rules may be "embedded" in a modular content item and offers Figure 4 (reproduced below) as an illustrative depiction of "at least one document page embedded with one or more assembly rules." EX1001, 3:51-53.

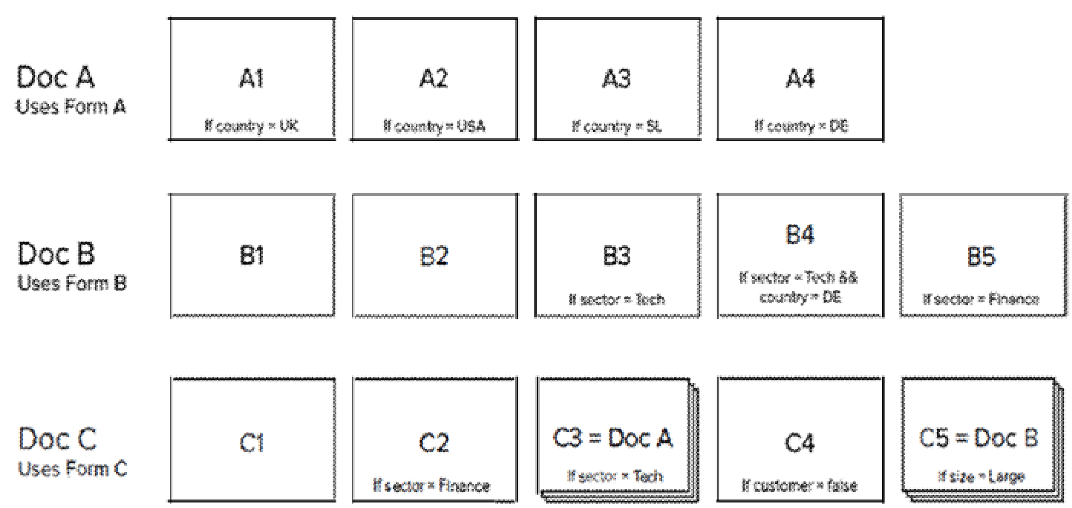


FIG. 4

74. As seen above, page A1 includes the rule "if country = UK" such that page A1 would be included in the generated document if the user location is "UK." EX1001, 18:64-19:8 ("FIGS. 4-6 show an example of assembling a document upon a personalization request. As illustrated in FIG. 4, documents A, B, C may each include a plurality of content items such as document pages A1, A2, A3, A4 for document A, document pages B1, B2, B3, B4, B5 contained in document B and

documents pages C1, C2, C3, C4, C5 contained in document C. In some cases, at least one content item of a document may be associated with assembly rules as described above. For instance, page A1 has embedded rule 'if country=UK' that when the condition is met (*i.e.*, the personalization request indicates country=UK) page A1 will be selected or displayed.") Beyond figure 4 and other similar figures, the patent does not give an example (*e.g.*, example code, file type, pointer, etc.) of how an assembly rule would be "embedded" in a content item. Further, the '290 patent provides no explanation, example, nor discussion of an interactive document either including or excluding these assembly rules.

75. The patent describes a "content renderer for rend[er]ing the assembled document." EX1001, 22:55-60 ("Referring back to FIG. 2, in some embodiments, the document assembly engine 203 may comprise a content renderer for rending the assembled document. The content renderer may be configured to execute a script (*e.g.*, JavaScript) associated with the content and display its output in the representation as defined by the script."). The function of content renderer is described only at a high level including that the content renderer may "execute a script (*e.g.*, JavaScript);" may "render the content or document as a web-page in a web browser;" and/or may "render content according to a brand theme/style mapping." EX1001, 22:55-67. However, the patent does not provide any example of what a "content renderer" may be, other than describing the intended function.

76. The patent also describes and claims the creation of a "link or connection between the source document and the derivative document." EX1001, 6:11-13. When an output document is created "the personalization parameters (*e.g.*, fields in the personalization form) that were provided during the personalization process and a reference to the source document may be stored in the system as a link or connection to the source document(s)." EX1001, 19:36-41. When the same output document is accessed by a reader, "the source document may be loaded and the stored link may be used to dynamically apply changes from the source documents." EX1001, 19:42-44. Aside from a reference to the "stored link" and the "link or connection," the patent offers no elaboration for what form the "link or connection" takes (*e.g.*, is it a data structure, a table, a dynamic reference, etc.). Further, the patent offers no description or explanation for how the personalization parameters are stored either as part of or along with the described "link or connection."

77. Figure 15 of the patent (reproduced below) purports to depict a "user interface" that "display[s] content live when a user completes a personalization form."

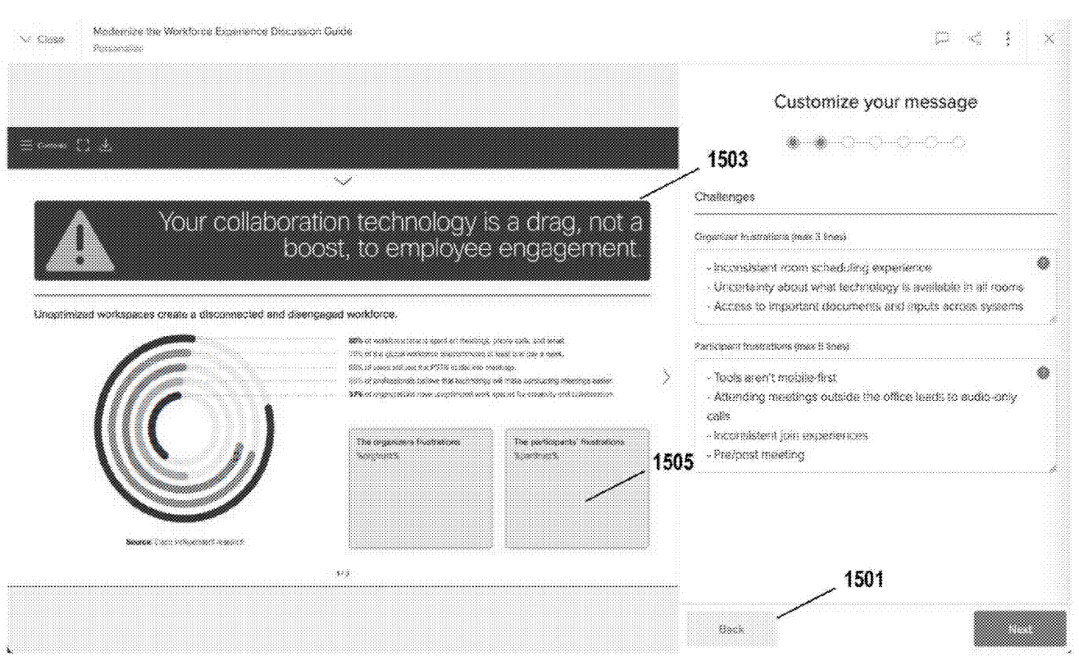


FIG. 15

EX1001, Fig. 15, 22:43-45. The patent states that "[u]pon receiving an input in a field of the personalization form, the document 1503 may automatically navigate to the relevant page and highlight the content (*e.g.*, blocks 1505) that is related to the field." EX1001, 22:48-51. The patent does not describe whether "navigation" requires any specific action (*e.g.*, scrolling, zooming, panning, etc.) other than showing the user the portion of the document that is "related to the field] of the personalization form.

VII. <u>OVERVIEW OF THE PRIOR ART</u>

A. *Harrop*

78. Just like the '290 Patent, Harrop teaches a computer-based document assembly system that generates different output documents (called "instance

documents") from XML source documents and personalized information provided by a user or database. EX1006-Harrop, 8:1-20.

79. In one embodiment, a user may use their computer's web browser to access the document assembly system. Over a network connection the assembly system accesses assembly modules 108, invokes a processing engine 202 to generate an instance document, and uses rendering engine 204 to send the rendered document to the user's web browser. EX1006-Harrop, 7:8-19. Figure 2 of Harrop (reproduced below and annotated to include the names of each component) depicts many of the components of Harrop's system. EX1006-Harrop, Fig. 2 (reproduced, annotated, below).

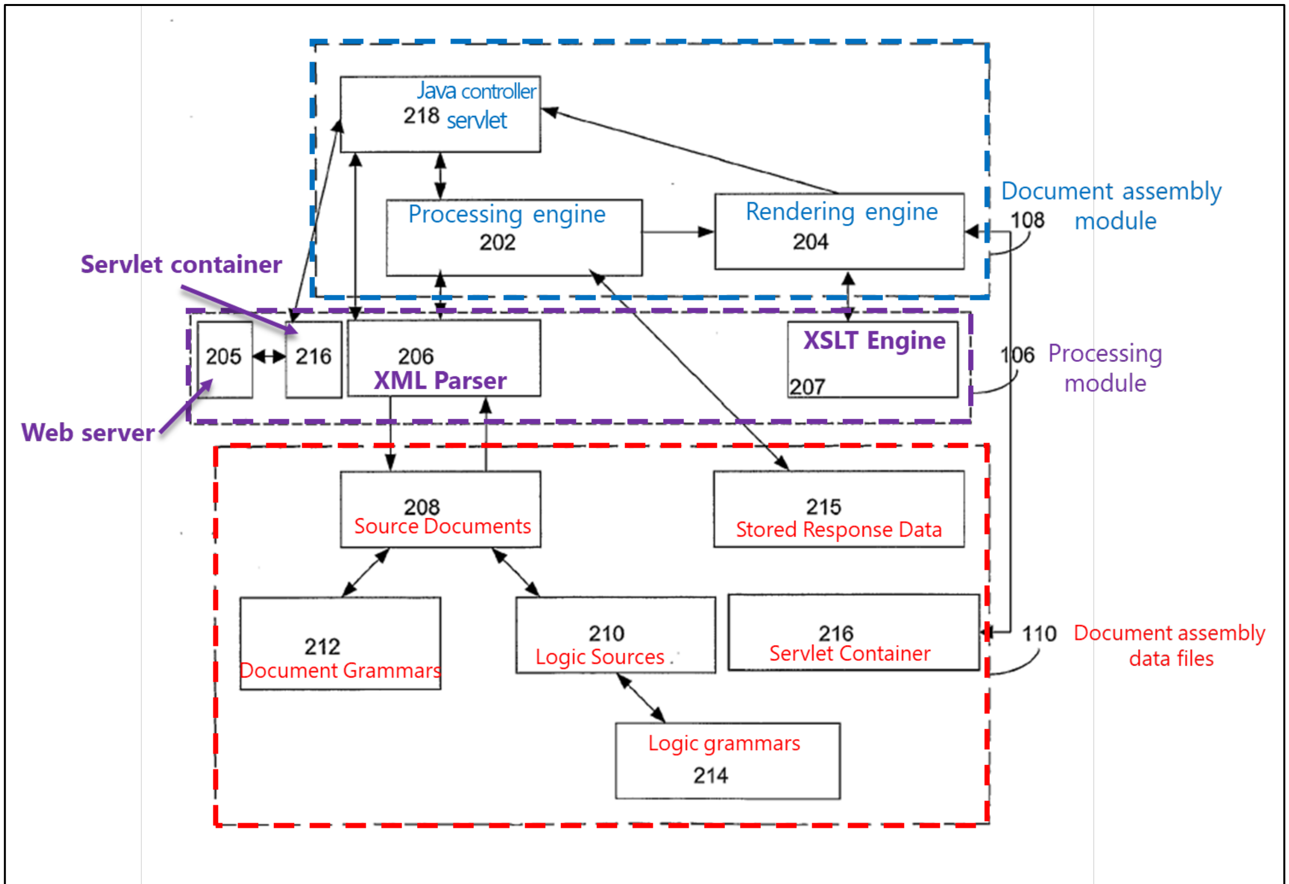
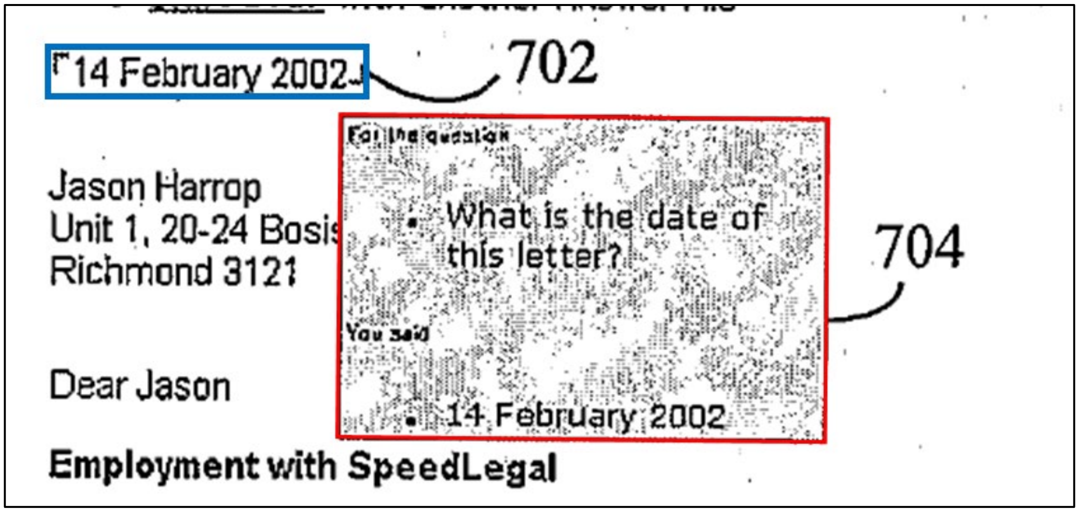


Figure 2

80. Harrop's system uses XML source documents 208 and information provided by a user or a data source to generate one or more instance documents. Ex1006-Harrop, 8:1-5. Where a user is the data source, the processing engine 202 uses logic sources 210 to generate questions and solicit necessary responses from the user; together, Harrop refers to the question and answers as "interview data" stored as "response data 215." EX1006-Harrop, 8:5-7, 9:15-19, 11:23-26, 12:8-12. The logic sources function as rules which, when evaluated by the processing engine and combined with interview data, instruct which text from the XML source

document should be included in the generated instance document. EX1006-Harrop, 9:11-15. Harrop describes that logic sources 210 may be separate from the source documents 208 or the logic sources may be incorporated in the source documents through embedded references to the logic sources. Ex1006-Harrop, 20:4-12.

81. As part of the question-and-answer interview process, Harrop's processing engine can reset questions as requested by the user and re-display the question to solicit a new answer. EX1006-Harrop, 15:10-15; 68:7-9, 68:20-22. Once a new answer is provided (and all questions are answered), a new instance document may be generated. EX1006-Harrop, 15:15-16. Specifically, Harrop describes "hypertext anchors" as part of a generated HTML instance document that, when a user moves their mouse over the hypertext anchor, a pop-up window showing the question and answer that led to the text being included. EX1006-Harrop, 15:8-10. Further, when the hypertext anchor is clicked "the processing engine 202 generates a new interview round including the relevant question, allowing the user to provide a different response to that question." EX1006-Harrop, 15:10-13. Figure 7 (reproduced-in-part below and annotated) depicts the hypertext anchor and the pop-up window showing the relevant question and answer. Fig. 7 (annotated).



Specifically, Harrop describes that the pop-up 704 is generated on mouse-over based on "JavaScript and the anchor element's HTML onMouseOver event." EX1006-Harrop, 68:14-18.

82. After the processing engine 202 has resolved all the processing-instructions logic, the rendering engine 204 renders the instance document into a desired format, such as HTML. EX1006-Harrop, 9:30-10:7. Additionally, in one embodiment, Harrop's system can generate a document "in HTML (without controls), XML, or in rich text format (RTF)." EX1006-Harrop, 15:20-23.

B. *Hailey*

83. Just like the '290 Patent, Hailey teaches a document generation system that uses input information regarding the document needed (*e.g.*, "type of transaction, number of parties, amount of loan, price, governing law, etc.") and generates an appropriate output file based on the input information. EX1005, ¶7.

Hailey's output file (or document) may be in one of a variety of file formats including PDF and HTML. EX1005, ¶42.

84. Hailey's output documents are built from content and rules (cross-references called "Xref") held within a knowledge base. EX1005, ¶¶44-45. Figure 4 illustrates one example of how the content (objects), rules (Xrefs), and knowledge base 54/120 can be arranged in Hailey's system. EX1005, ¶64, Fig. 4 (annotated below).

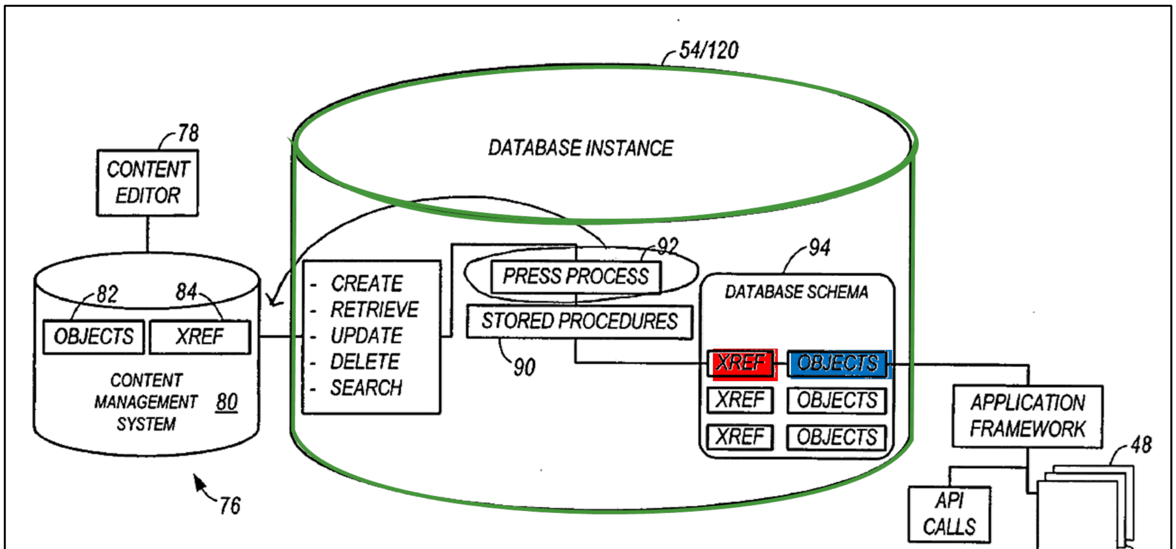


FIG. 4

85. As seen in figure 4 above, the knowledge base 54/120 is depicted as a "database instance." *Id.*, Fig. 4, ¶95 ("embodiments of the invention may include a database or knowledge base"). In addition, Hailey describes that the xref-object pairs shown in database schema 94 are stored as associated pairs. EX1005, ¶64.

86. In one example, Hailey describes one type of input information (location) where the appropriate consumer protection clause is included in the instance document depending on the location of the transaction (*e.g.*, New York). EX1005, ¶5.

C. *Davis*

87. Davis teaches "a method for tracking the use and interaction of a user with a resource…by use of a tracking program embedded in the resource." EX 1007-Davis, 4:3-5. Davis teaches that, even in 1997, including user tracking features in an HTML document was known. Specifically, Davis teaches at least one embodiment where "the tracking program is embedded in an HTML document (such as a Web site, a Web page, or part of a Web page—*e.g.* an 'ad banner.'" EX1007, 5:13-16.

88. Davis teaches embedding a tracking program in an HTML document that is sent from a server to a client with the tracking program executing on the client machine. EX1007, 8:6-8. The tracking program "may monitor various indicia. such as time, mouse events, keyboard events, and the like, in order to track a user's interaction with the Web page." *Id.*, 8:6-13. Thus, Davis explains, that in one embodiment "the tracking program may simply monitor the amount of time the user spends interacting with the Web page, or may monitor details of choices (such as links) made by individual users within a particular Web page." *Id.*, 8:13-20. In another embodiment, "[t]he tracking program may monitor operation of a peripheral

input device connected to the client machine, such as a keyboard or mouse, keep a record of which choices, if any, are made by a user, and may monitor the length of time the user has displayed the Web page in addition to the time spent interacting with a particular part of it." *Id.*, 8:30-38. Davis explains that tracking these user interactions "provide[s] meaningful data to the server concerning the user's interaction with the Web page." EX1007, 8:64-9:2.

89. Davis teaches that "[t]he tracked information may be used to assemble resources geared toward the user's interests." EX1007, 14:47-48. For example, "[b]ased upon the historic user profiles created in the server database, downloading of information to the same client on a subsequent visit to the same or different Web page may be done on a more intelligent basis." EX1007, 14:48-52. The tracked information can be used to create user profiles that include "details of choices created by individual users within a particular resource." EX1007, 4:24-32. Davis teaches using this tracked information to generate resources (*e.g.*, HTML documents) based on, for example, a historic user profile. Id., 4:33-36. Thus, just like Harrop and Hailey, Davis also teaches a system that creates customized documents based on personalization parameters (*e.g.*, a user's profile).

D. *McKinlay*

90. As described below, McKinlay is relied upon only in Ground 2 as an alternative to Ground 1. *See* §VIII.B, *infra*. Specifically, McKinlay provides details

of one example for how to implement a pop-up window based on JavaScript and HTML coding similar to Harrop's description of hypertext anchors.

91. McKinlay teaches one example of using URL parameters in a URL hyperlink and JavaScript code to generate a pop-up window with particular information. EX1008, ¶86. In addition, McKinlay provides example URL and JavaScript code used to generate the desired pop-up window. *Id.*, ¶¶86-93. Figure 14 depicts one example of McKinlay's described pop-up window 1402. EX1008, ¶83, Fig. 14.

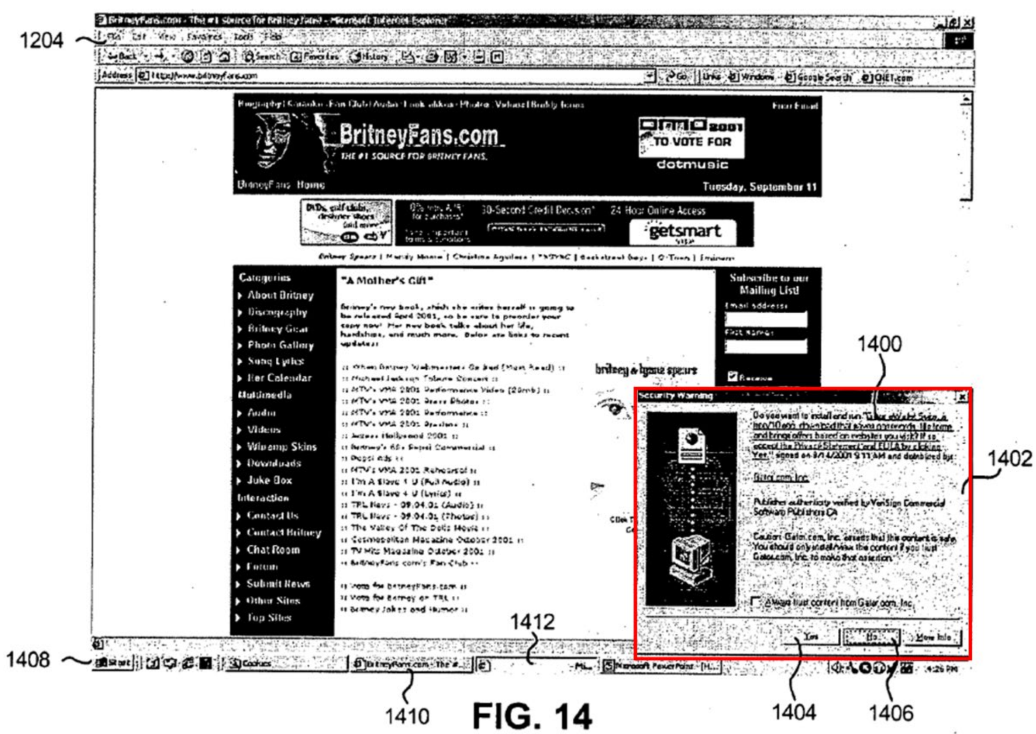


FIG. 14

E. *Lo*

92. Lo teaches a document creation and printing system that allows multiple documents to be created from a single document template, "where each document is customized for its intended readers." EX1009, ¶¶5, 79. Lo's customized documents are "are created by inserting variable data into a document template." *Id.*, ¶6.

93. Further, Lo teaches using "live proofing" to "display[] variable data merged with the variable document template interactively and on-demand." *Id.*, ¶248. As shown in figure 20 (reproduced below), Lo's live proofing shows "what part of [a] record would look like if printed" including "extra borders 2004 and colors 2006…to make visible the constraints defined within the variable document template." EX1009, ¶428, Fig. 20.

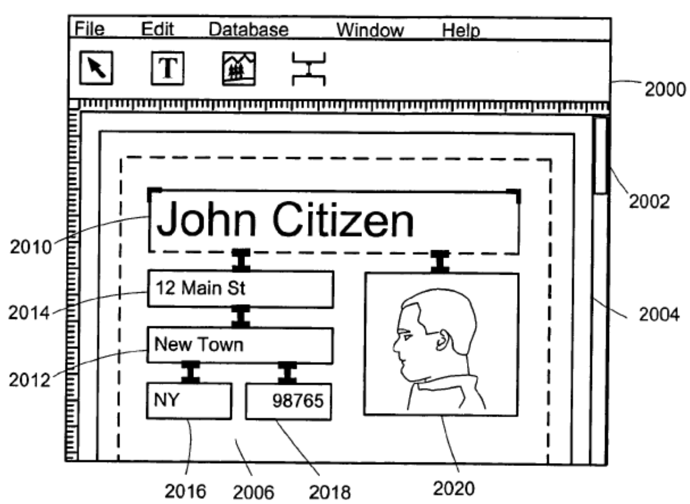


Fig. 20

94. As seen in Fig. 20 above, Lo's "live proofing" view shows only the portion of the document that is relevant to the edited variables in the document template. More specifically, Lo teaches that a "record" is a set of data used to produce one instance document. EX1009, ¶87. Thus, Lo teaches display what a "part" of a record (*i.e.*, data set) will look like—as opposed to displaying the entire data set.

95. Lo explains that "[l]ive proofing is advantageous over a separate merging process because errors in the design of the template are likely to be found earlier through interactive navigation." EX1009, ¶429.

VIII. SPECIFIC GROUNDS FOR UNPATENTABILITY

A. *Ground 1: Claims 1-5, 7-14, and 16-18 are rendered obvious by Harrop, Hailey, and Davis*

96. As explained in greater detail in the following paragraphs (¶¶84-152), it is my opinion that document-assembly systems were well known to a POSITA at the time of filing of the '290 Patent, and any limitations recited in the claims of the '290 Patent would have been obvious to a POSITA in view of Harrop, Hailey, and Davis.

1. Reasons to combine

97. In my opinion, a POSITA would have been motivated to combine Harrop, Hailey, and/ or Davis to arrive at the features of claims 1-5, 7-14, and 16-18 for numerous reasons presented below.

a. Analogous Art

98. In my opinion, Harrop, Hailey, and Davis are in the same field of endeavor as the ’290 Patent, namely document assembly and creation. *See* EX1001, abstract; 1:36-56. Hailey, for example, is directed to the creation of documents by assembling document components and applying rules to create a derivative document. EX1005, abstract, ¶2. Harrop is directed to document assembly and systems and methods for document assembly, or the generation of a derivative document from one or more source documents. EX1006, abstract, ¶¶2-3. Davis also teaches generating a document (HTML web page) based on personal information (*e.g.*, a user profile). EX1007, 14:46-52. Further, the ’290 Patent generally describes document creation and assembly systems and methods for use on the Internet. *See* EX1001, 4:63-5:24. Harrop, Hailey, and Davis also relate to the application of document creation systems and methods for use on the Internet, including standard hyper-text markup language (“HTML”) documents. *See* EX1005, ¶42 (output file in HTML format); EX1006, 6:28-7:6 (document assembly system can be accessed by a user over the Internet), 15:18-28 (output document in HTML format); EX1007, 4:1-10, 4:33-36, 5:14-18 (collecting user tracking information and using it to generate a customized HTML document).

b. <u>Motivation to Combine</u>

99. In my opinion, a POSITA would find it obvious to combine the teachings of Hailey, Harrop, and Davis.

100. A POSITA would have recognized that the systems of Hailey and Harrop are closely related and would have found it obvious to combine various teachings within each reference to improve the system of the other reference. For example, a POSITA would have found it obvious to include Harrop's teaching of linking portions of the interactive document, the master document, and the personalization parameters through hypertext anchors in a generated HTML document. In particular, a POSITA would have appreciated Harrop's described benefit of the feature as a way for users to inspect portions of a final document quickly (*i.e.*, via a pop-up) and edit those sections by adjusting the input variables. EX1006, 67:28-68:18. Hailey describes a similar user-input interface in its origination platform and teaches soliciting further answers from a user when more information is needed to generate a document. EX1005, ¶¶40-41. Accordingly, it is my opinion that a POSITA would have been motivated to include Harrop's teaching of hypertext anchors which generate a pop-up with associated questions and answers, and which reset associated answers when clicked to allow a user to make relatively quick changes to the content included in a portion of a generated document.

101. Further, as discussed above, Davis teaches including a tracking features in an interactive document. EX1007, 5:14-18; *see* ¶¶87-89, *supra* (incorporated here). Davis teaches embedding a tracking program in an HTML document to gain information regarding user preferences, collect information regarding the specific, interactions, and create user profiles based on the tracked information. *See* EX1007, 4:3-8 ("[T]he present invention is to provide a method for tracking the use and interaction of a user with a resource downloaded from a server on a network by use of a tracking program embedded in the resource and executable by a client."), 4:19-23, ("another object of the present invention is to provide means for creating a database of user profiles for use by advertisers and/or marketers to determine the effectiveness and value of network-based advertisements and/or marketing resources"), 4:33-36 ("another object of the present invention to provide means for assembling a resource, such as a Web page or a highly targeted ad banner, in accordance with a historic user profile."). Specifically, Davis's tracking feature monitors various user interactions including "time, mouse events, keyboard events," (*id.*, 8:13-20) as well as, "details of choices created by individual users within a particular resource." *Id.* 4:24-32. In my opinion, a POSITA would have recognized that Davis teaches using these tracked interactions to assembly a customized resource (*e.g.*, HTML webpage) based on a user's profile which includes the tracked information. *Id.*, 4:33-36.

102. Accordingly, it is my opinion that a POSITA would have been motivated to include in the systems of Hailey and/or Harrop a tracking feature that collects data indicative of user interactions with interactive documents, as taught by Davis, to use that collected data to improve the respective systems in various ways. For example, a POSITA would have been motivated to track user interaction with the interactive documents generated by Hailey and Harrop to glean information regarding user preferences to respective variations of user documents including preferences as to the formatting, size, color, file type, delivery method, etc. of the interactive documents. For example, a POSITA would have understood that tracking a user's responses to questions posed by Hailey or Harrop's system regarding a user's name, entity, or job can be tracked using the tracking program described by Davis and that information can be reused in Hailey or Harrop's systems when assembling new interactive documents. Further, as described above in the Technological Background section (¶¶54-66, incorporated here), a POSITA would have been familiar with numerous reasons to track user interaction with interactive documents because designers of interactive systems improve the quality of those systems by understanding what works well and what does not for their users. Measuring how people use those systems by seeing what they read, what they click on, what they look at first, etc. inform the designer whether the system is working as intended. For example, I have done this myself numerous times when I wanted to

determine which of two possible interface designs was faster and more accurate, and I learned to do this as a best practice within the field of human-computer interaction.

c. Reasonable Expectation of Success

103. In my opinion, the references include similar arrangements of well-understood components and use predictable technology in routine and conventional ways.

104. For example, a POSITA would have found it routine and predictable to implement Harrop's hypertext anchors in the system of Hailey through known technologies like URL parameters and/or JavaScript functions. Indeed, Harrop itself teaches that the pop-ups associated with the hypertext anchors are generated through "JavaScript and the anchor element's HTML on MouseOver event…" EX1006, 68:20-22. Similarly, Hailey teaches reviewing a generated (assembled) document and, if information is missing (*e.g.*, a loan amount), querying the assembly engine in order to request the missing information from the user or database. EX1005, ¶40. In other words, Hailey also provides a way for the system to correct a generated document by querying a user for additional information. In my opinion, a POSITA would have recognized these well-known and commonly used web-based document languages and would have found it routine to implement the same or similar JavaScript and/or HTML coding in Hailey's generated HTML output file to accomplish the same or similar pop-up generation and reset of associated

variables/questions. A POSITA would have considered the combination of these teachings nothing more than the implementation of well-known concepts in well-known systems that had been known for decades prior to the '290 Patent.

105. Further, in my opinion a POSITA would have found it routine and predictable to include the user tracking feature of Davis in the systems of Harrop and Hailey. Each of Davis, Harrop, and Hailey describe generating documents in an HTML format. Further, Davis teaches that his tracking feature can be embedded as part of an HTML document. EX1007, 5:14-15 ("In one embodiment of the present invention, the tracking program is embedded in an HTML document"). It is thus my opinion that a POSITA would have understood that implementing Davis's user tracking program would have merely required implementing known code functions (*e.g.*, Davis's tracking program) in a known way (*e.g.*, using known HTML or JavaScript functions) in known systems (*i.e.*, HTML documents).

2. <u>Claim 1</u>

a. <u>Limitation 1[PRE]: A system for creating an interactive document, the system comprising:</u>

106. Limitation 1[PRE] recites a system for creating an interactive document. Hailey, Harrop, and Davis all disclose creating documents that are interactive. In my opinion, a POSITA would have understood that Hailey, Harrop, and Davis all teach a system for creating an interactive document and that the Hailey/Harrop/Davis combination renders the elements of the preamble obvious.

107. For example, Hailey and Harrop each teach creating documents for display in a web browser. *See*, *e.g.*, EX1005, ¶42 ("The output file (or document) may be in one of a variety of file formats as is illustrated by three exemplary file format blocks (or documents) 44, 46, and 48 (.pdf, HTML, and other)"); EX1006, 15:2-6 ("the instance [*i.e.* generated] document is then displayed by the web browser, as shown in FIG. 7."). A POSITA would have understood that when a document is displayed in a web browser, a user may interact with the document in various ways including scrolling, activating links, interacting with animations and/or dynamic objects, etc. and that many of those interactions may be present in a PDF document as well. Accordingly, it is my opinion that a POSITA would have understood that Harrop and Hailey both teach creating interactive documents.

108. As discussed above, the '290 Patent describes "interaction" as including user actions such as reading, editing, savings, browsing, filtering, and/or selecting video or web page. EX1001, 6:25-32, 25:15-20; ¶¶54-64, *supra* (incorporated here). Accordingly, it is my opinion that Hailey and Harrop both teach interactive documents with at least as much interaction as the interactive documents described in the '290 Patent.

b. Limitation 1[A]: a memory for storing a set of software instructions, and one or more processors configured to execute the set of software instructions to perform one or more operations comprising:

109. The Hailey/Harrop/Davis combination renders this limitation obvious. Each of Hailey, Harrop, and Davis teach this limitation.

110. In my opinion, a POSITA would have recognized that Hailey teaches implementing its document creation system "in software which may be stored as instructions on a computer readable medium… and executed by a microprocessor or a similar device." EX1005, ¶99.

111. Harrop discloses the example of a computer server such as a Sun Fire® 15K server, and "processing modules 106, 108, and 112 of the document assembly system implemented as software modules stored on hard disk storage 104 of the server 102." EX1006, 7:23-26.

112. Davis teaches a system that can be implemented via a client computer and server, each of which includes memory, a processor, and software. EX1007, 6:35-42, 6:52-67, 7:30-49, Fig. 1, Fig. 2.

113. A POSITA would have recognized any of these example environments as teaching a memory for storing a set of software instructions, and one or more processors configured to execute the set of software instructions to perform one or more operations. Thus, in my opinion, a POSITA would have understood that

Hailey, Harrop, and Davis each teach Limitation 1[A] and, therefore, a POSITA would have found that the Hailey/Harrop/Davis combination renders this limitation obvious.

c. **Limitation 1[B1]: (a) generating an interactive document comprising one or more modular items, wherein the one or more modular items are pulled from a master document based at least in part on one or more personalization parameters,**

114. In my opinion, the Hailey/Harrop/Davis combination renders this limitation obvious. A POSITA would have recognized that each of Hailey and Harrop teach this limitation.

115. Hailey teaches using content (modular items) stored in a knowledge base (master document) and to create output files (interactive documents) based on various input values (personalization parameters). EX1005, ¶39. Similarly, the ’290 Patent describes that the master document may be “stored as structured data in a document database.” *See, e.g.*, EX1001, 27:16-18 (claim 17); 2:4-9; 13:38-14:25; 15:37-50. In my opinion, a POSITA would have understood that Hailey’s knowledge base which contains the content is a “master document” and that the various content are pulled from this database (*i.e.*, the knowledge base). Hailey teaches that the content stored in database 54/120 as groups of “objects” and associated rules (“XREF”). EX1005, ¶64, Fig. 4. These grouped rules (XREF) and objects form one or more modular items that are used to generate interactive documents 44, 46, 48.

EX1005, ¶64 ("The content is used… to create documents such as the documents 44, 46, and 48.").

116. Hailey further teaches that the content (*i.e.* objects and rules) are pulled from the database (master document) based on various data input (*i.e.* personalization parameters) into the system. EX1005, ¶40 ("[T]he document generation module 22 performs four basic functions. The document generation module 22 validates incoming data (as is represented by a validation module 24), applies precedence (as is represented by precedence module 26), applies rules to the content (as is represented by a rules application module 28), and performs document assembly (as is represented by an assembly module 30).") For example, Hailey describes processing location data (*e.g.*, Texas or California) input into origination platform 12 to select and generate an output file that includes appropriate content for the relevant location. EX1005, ¶40-41. Hailey further describes soliciting additional input data from origination platform 12 to resolve any remaining questions and complete document assembly. EX1005, ¶40 ("For example, if the amount of a loan has not been specified in the input data originally provided by the origination platform **12**, a question module **32** queries the origination platform **12** to request needed information or data **34**."), Fig. 1. Accordingly, it is my opinion that a POSITA would have recognized that the data input into the origination platform

taught by Hailey are personalization parameters which influence, at least in part, which content is assembled into the final output file.

117. Similarly, a POSITA would have recognized that Harrop teaches generating an instance document (interactive document) based on XML modules (modular items) that include text options which are pulled from one or more XML source documents (master documents) based on the values of variables relevant to the circumstances (*e.g.*, based on information specific to the parties involved and the circumstances of their relationships). EX1006, ¶¶8:1-20, 8:22-9:23.

118. In one example, Harrop teaches using a user's answers to interview questions to "determine values for variables" which "determine content for the resulting instance document." EX1006, 14:7-11. "The user's responses are used to assign values to variables, such as setting Boolean variables to a value of true or false for determining whether an associated portion of text is to be included in a generated instance document." EX1006, 9:19-22. Thus, a POSITA would have understood that whether a portion of text is to be included in the output document is at least partly based on the user's responses (*i.e.*, personalization parameters).

119. Harrop provides an illustrative example regarding the creation of a letter of employment for a prospective employee ("Jason"). EX1006, 14:21-31, Fig. 6. Figure 6 illustrates one example of an interview question at reference number 604: "What are the core hours when Jason is expected to attend the office?"; with

the user (*e.g.*, HR admin) answers: "10am – 12 pm and 2pm – 4 pm." *Id.*, Fig. 6. Once all the questions about Jason and his employment are answered, Harrop's system "generates at step 510 an instance document in HTML format on the basis of responses provided by the user, and the instance document is then displayed by the web browser, as shown in Figure. 7." EX1006, 15:1-6.

120. Thus, it is my opinion that a POSITA would have recognized that Harrop uses the variables corresponding to the questions asked and answers provided by a user as personalization parameters. The answers are applied to the appropriate XML module (modular item) to resolve an associated logic and the appropriate a portion of text is pulled from the XML source document (*i.e.*, master document) and the text is included in the instance document (*i.e.*, interactive document) is at least partly based on the values (*i.e.*, answers) of those variables (*i.e.*, personalization parameters).

121. Accordingly, a POSITA would have understood that both Hailey and Harrop teach generating an interactive document comprising one or more modular items (*e.g.*, content objects / modular text), wherein the one or more modular items are pulled from a master document (*e.g.*, database / XML source document) based at least in part on one or more personalization parameters (information related to the parties / variables holding answers as values). Thus, it is my opinion that a POSITA

would have understood that both Hailey and Harrop each teach Limitation 1[B1] and the Hailey/Harrop/Davis combination render this limitation obvious. *Id.*

d. **Limitation 1[B2]: and wherein the interactive document comprises a tracking feature to collect data indicative of user interaction with the interactive document;**

122. In my opinion, a POSITA would have found that the Hailey/Harrop/ Davis combination teaches this limitation.

123. Davis teaches a tracking feature that collects data indicative of user interaction with the interactive document. For example, Davis teaches embedding a tracking program in an HTML document, the tracking program monitoring and collecting information regarding how a user interacts with the HTML document. *See* EX1007, 8:13-20 ("the tracking program may simply monitor the amount of time the user spends interacting with the Web page. or may monitor details of choices (such as links) made by individual users within a particular Web page."). Davis uses the tracked information to generate personalized resources (*e.g.*, HTML documents). EX1007, 4:33-36 ("another object of the present invention to provide means for assembling a resource, such as a Web page or a highly targeted ad banner, in accordance with a historic user profile.").

124. In my opinion, a POSITA would find it obvious to include a tracking feature to collect data indicative of user interaction—such as the one taught by

Davis—in the systems of Hailey and/or Harrop. Accordingly, the combined teachings of Hailey, Harrop, and Davis render obvious limitation 1[B2]. *See also* ¶¶97-105, *supra* (incorporated here).

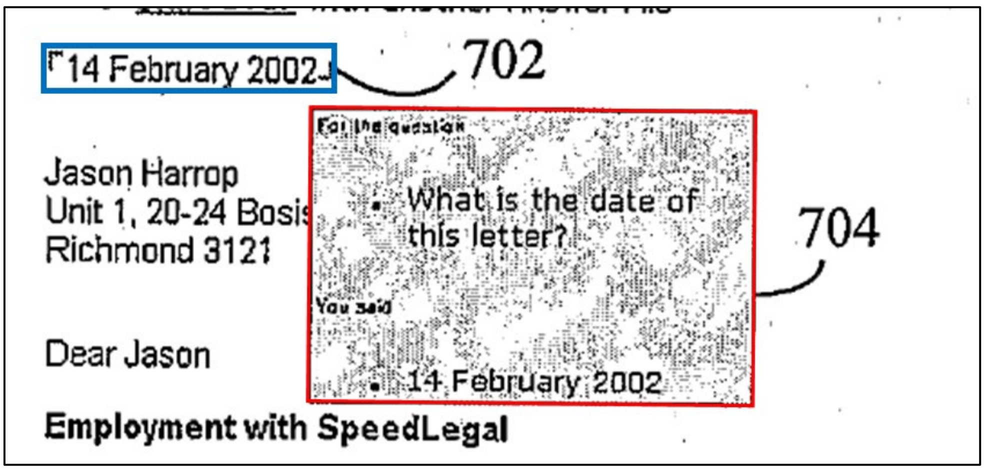
e. <u>Limitation 1[C]: (b) creating a link between the interactive document and the master document, wherein the link comprises the one or more personalization parameters;</u>

125. In my opinion, a POSITA would have found that the Hailey/Harrop/Davis combination teaches this limitation.

126. Limitation 1[C] recites creating a link between the source and output documents, where the link includes personalization parameters. The '290 Patent describes that "the personalization parameters (*e.g.*, fields in the personalization form) that were provided during the personalization process and a reference to the source document may be stored in the system as a link or connection to the source document(s)" so that changes made to source documents can be reflected when rendering derivative documents. EX1001, 19:36-41. The patent does not describe any form or data structure for the "link or connection."

127. Harrop teaches a link between the instance document (interactive document) and a master document (XML source document). For example, Harrop teaches rendering an HTML instance document where certain text is rendered as "a hypertext anchor" to indicate the question and answer that resulted in that particular

text being included in the rendered document. EX1006, 67:28-68:5. Specifically, Harrop teaches that hovering (or mousing over) the hypertext anchor "displays the question and response that resulted in the anchor text being included." EX1006, 68:1-2. Figure 7 of Harrop depicts one example of a **pop-up window** showing the question and answer associated with the **included hyperlinked text**.



EX1006, Fig. 7 (reproduced-in-part, annotated).

128. Further, Harrop teaches that when a hypertext anchor is clicked on by a user, the processing engine 202 is invoked to "reset" the questions associated with the linked text. EX1006, 68:14-18. The processing engine 202 uses XML source documents 208 and logic sources 210 to resolve all logic in the source document including all questions and required answers. *See* EX1006, 8:22-9:9, 9:30-10:7, 14:13-19, 15:1-6. Specifically, Harrop teaches that the Processing Engine 202 "resolves logic associated with an XML source document 208." EX1006, 9:5. Figures 1 and 2 show that the processing engine 202 is a component of the document

assembly modules 108, which is shown as operating on the server 102. EX1006, Figs 1 and 2, 6:28-29 ("As shown in Figure 1, a document assembly system includes a server 102 having a number of modules 106 to 112.'), 7:8-9 ("The document assembly modules 108 include a Java controller servlet 218, a processing engine 202 and a rendering engine 204."). Accordingly, a POSITA would have recognized that Harrop's processing engine 202 operates on the server 202. In addition, Harrop describes that when the hypertext anchors are clicked by a user, the processing engine will generate a new round of interview questions to allow a user to select a different response and generate a new instance document. EX1006, 15:8-16. Based on Harrop's teachings, a POSITA would have recognized that when Harrop's processing engine is invoked to reset the relevant questions, the processing engine would access the relevant XML source document and that a reference, pointer, or other information would be included in the generated HTML document for the source document be accessed.

129. A POSITA would have understood that Harrop's teaching of displaying the specific question and answer associated with a hypertext anchor and resetting the questions associated with the hypertext anchor teaches that Harrop's system includes a link between the interactive document (the HTML instance document containing the hypertext anchor) and the master document (XML source document). More specifically, a POSITA would have recognized that, for the processing engine 202

to reset the relevant question, the hypertext anchor and/or the HTML code associated with the anchor includes information that allows the processing engine to access the relevant portion (*i.e.*, the question) in the relevant XML source document. Further, a POSITA would have recognized that, to display a pop-up window which shows the relevant question and answer, Harrop's system includes information that allows the web-browser to render the relevant question and answer in the pop-up window. It is my opinion that a POSITA would have understood that the collection of information and the hypertext anchors that Harrop's system uses to perform the functions associated with the hypertext anchors creates a "link" between the interactive document (HTML instance document) and the master document (XML source document). A POSITA would have understood that Harrop's system links the hypertext anchor, the question and answer (*i.e.*, the personalization parameters), and the XML source document (master document) together. Accordingly, a POSITA would have recognized that Harrop teaches a "creating a link between the interactive document and the master document, wherein the link comprises the one or more personalization parameters," as claimed.

130. As discussed above, a POSITA would have found it obvious to include Harrop's link in the system of Hailey to connect generated interactive documents and the master documents at least for the reasons discussed above. *See* ¶¶96-105, *supra* (incorporated here). Accordingly, it is my opinion that a POSITA would have

found that limitation 1[C] is obvious in view of the Hailey/Harrop/Davis combination.

f. **Limitation 1[D1]: (c) rendering the interactive document with aid of a content renderer,**

131. In my opinion, a POSITA would have found that the Hailey/Harrop/ Davis combination teaches this limitation. As discussed with respect to limitation 1[PRE], Hailey, Harrop, and Davis each teach creating interactive HTML and/or PDF documents. *See* ¶¶106-108, *supra* (incorporated here). Further, Harrop and Davis both teach rendering the HTML file to display in a user interface. EX1006, 67:28-31. ("The instance [HTML] document can be rendered…."); EX1007, 8:53-60 ("As noted above, a client process, such as a Web browser running on the client machine, uses a TCP/IP connection to pass a request to a Web server running an HTTP service (or "daemon" under the UNIX operating system). The HTTP service then responds to the request, typically by sending a Web page formatted in the Hypertext Markup Language, or HTML, to the browser. The browser displays the Web page using local resources (*e.g.*, fonts and colors)."). Harrop further teaches a "rendering engine 204" used to render the HTML instance document on a display. EX1006, 67:28-31.

132. A POSITA would have readily understood that displaying HTML documents on a display includes rendering visual content from the HTML document.

Accordingly, a POSITA would have understood that each of Hailey, Harrop, and Davis contemplate using a web browser as a content renderer to display the generated documents. Therefore, it is my opinion that a POSITA would have recognized that Harrop, Hailey, and Davis all teach or suggest rendering a viewable image of the interactive document with the aid of a content renderer (*e.g.*, web browser).

133. Accordingly, a POSITA would have understood that each of Hailey, Harrop, and Davis individually and/or in combination with each other teach and suggest limitation 1[D1].

g. Limitation 1[D2]: wherein the content renderer is configured to use the link to apply a change to the one or more modular items based on a change to the master document.

134. In my opinion, a POSITA would have understood that the Hailey/ Harrop/Davis combination renders this limitation obvious. As discussed with respect to limitation 1[C], Harrop teaches and/or renders obvious creating a link between the interactive document and the master document, wherein the link comprises the one or more personalization parameters. ¶¶125-130, *supra* (incorporated here).

135. Harrop further teaches that the link (*i.e.*, the connection between the interactive document, master document, and variables/answers) is used to apply a change to one or modular items based on a change to the master document. That is,

a POSITA would have understood that Harrop's system is designed to render the instance document based on the XML source document and that any changes to the XML source document would result in a change to the instance document the next time the instance document is generated and rendered. EX1006, 10:5-7, 15:2-6, 15:8-16 (resetting questions, querying new answers, rendering a new HTML instance document). For example, if a user clicks the hyperlink anchor in Harrop's HTML instance document, the question will be reset, and a user will have to provide a new answer. EX1006, 15:8-16. If the source XML associated with the reset question has been modified since the HTML instance document was first rendered, then the re-rendered instance document—which now contains a new answer—will reflect the change made to the XML source document. A change to a question, text options, or answer choices in an XML source document is also a change to the master document itself.

136. To the extent that Harrop does not expressly teach that its content renderer uses Harrop's link to "apply a change to the one or more modular items ***based on a change to the master document***," it is my opinion that a POSITA would have found it obvious to modify Harrop to apply a change to the modular items based on a change to the master document. Harrop teaches using reference links to incorporate XML code module into a given XML source to automatically update the resulting instance document based on changes in the XML module. EX1006, 55:22-

24 ("Alternatively, the user can import into the document a link to the module—this can point to a current version of the module, and any modifications subsequently made to the module are automatically reflected in the resulting document."). A POSITA would have understood that Harrop teaches the advantage of using links to source documents—in this case, an XML module—so that changes to the source can be automatically updated in the resulting document. In my opinion, a POSITA considering Harrop's disclosure of this benefit and Harrop's teaching of linking the master document, instance document, and personalization parameters, as discussed regarding limitation 1[C], would have found it obvious to use Harrop's links to automatically update an instance document and the modular items (*i.e.* text) contained therein, based on changes to the source document. *See* ¶¶125-130, *supra* (incorporated here).

137. Therefore, it would have been obvious to a POSITA, to include in the systems of Harrop and Hailey a rendering engine that uses a link to apply a change to a modular item based on a change to the master document. This combination of teachings is applicable to both Harrop—since it contains the teachings and suggestions—as well as Hailey for the reasons discussed in ¶¶97-105 above (incorporated here). Accordingly, the Hailey/Harrop/Davis combination renders limitation 1[D2] obvious.

3. **Claim 2: The system of claim 1, wherein the one or more modular items from the master document are embedded with an assembly rule.**

138. In my opinion, a POSITA would have recognized that the Hailey/ Harrop/Davis combination renders obvious all the limitations of claim 1 and further renders claim 2 obvious. *See* ¶¶106-137, *supra* (incorporated here). As discussed regarding limitation 1[B1], Hailey teaches that content (modular items) are stored in the database as groups of XREF (rules) and objects. *See* ¶¶114-121, *supra* (incorporated here); EX1005, ¶¶39-41, 64. "[T]he rules take the form of rules markup created using an XML script or schema (or, more broadly, an architecture or namespace) and may be referred to as 'xref' embedded rules". EX1005, ¶45. These XREF rules 56/84 are "used to determine a set of documents" and are, therefore, assembly rules. EX1005, ¶77. Figure 4 of Hailey (annotated below) shows that the content (modular items) contained in the database 54/120 are embedded with the XREF rules since they are contained within each XREF-Object grouping that forms a content.

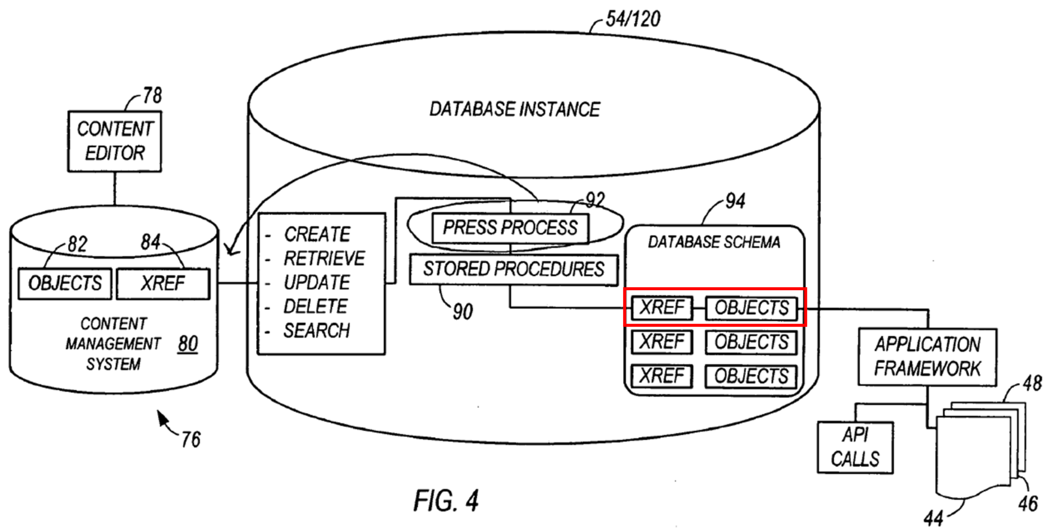


FIG. 4

139. In addition, Harrop also teaches embedding the processing instructions in the XML modules in the source documents 208 (*i.e.*, master document) with the logic sources 210. EX1006, 20:4-20. In one example of an XML source document provided by Harrop (annotated below), the logic source is embedded in a processing instruction: <?SpeedLegal?>.

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE Letter SYSTEM "file:///home/speedlegal/dtd/xhtml-letter/dtd/xhtr
v2.dtd">
<Letter>
  <?SpeedLegal
    <LogicSources>
      <LogicSource ID="LogicSource__1" uri="/files/logic/LS1WD.lgc"/>
      <LogicSource ID="LogicSource__2" uri="/files/logic/LS2WD.lgc">
        <PartyMapping>
          <PartyMap From="Contractor" To="Employee">
            <PartyDetailMap From="Shortname" To="Firstname"/>
            <PartyDetailMap From="Fullname" To="Firstname"/>
          </PartyMap>
        </PartyMapping>
      </LogicSource>
    </LogicSources>
  ?>
  <?SpeedLegal
    <MetaInformation>
      <rdf:RDF xmlns:dc="http://purl.org/dc/elements/1.1/"
xmlns:rdf="http://www.w3.org/TR/WD-rdf-syntax-ns#">
```

```
<LetterHead>
  <Date/>
  <Salutation/>
</LetterHead>
<LetterBody>
  <p>
    <?SpeedLegal
      <Condition IDREF="PaidLeave" LogicSource="LogicSource_1"/>
    ?>
    <?SpeedLegal
      <PartyReference IDREF="Employee" LogicSource="LogicSource_1"
Style="Pronoun" Type="PartyDetail_1"/>
    ?> shall be eligible for paid leave.</p>
  <p>
```

EX1006, 48:15-38, 18:18-31. In my opinion, a POSITA would have understood that the above example shows embedded assembly rules (*i.e.*, the logic sources and their application) which instruct the processing engine how to assemble the appropriate text once the appropriate questions are answered. The first extract above shows two logic sources (LogicSource_1 and LogicSource_2) being embedded. Then the second extract above shows the application of the logic from LogicSource_1. It shows that the text "shall be eligible for paid leave" will only be included if the "PaidLeave" property is true. It also includes the employee's name in the generated text. Accordingly, a POSITA would have recognized from these extracts that Harrop teaches embedding an assembly rule (*i.e.*, logic sources and their application) in a modular item (*i.e.*, XML processing instruction module) in a master document (*i.e.*, XML source document).

4. <u>Claim 3: The system of claim 2, wherein the one or more modular items in the interactive document are not embedded with the assembly rule.</u>

140. In my opinion, a POSITA would have understood that the Hailey/ Harrop/Davis combination renders obvious all the limitations of claims 1 and 2, and further renders claim 3 obvious. *See* ¶¶106-139, *supra* (incorporated here).

141. Claim 3 further limits claim 2 such that "the assembly rule" which is embedded in the modular item in the master document is ***not embedded*** in the modular items in the interactive document. Although the '290 Patent describes the embedding of assembly rules in modular items in a master document, the '290 Patent provides no description or example of an interactive document that *does not* include the assembly rule. *See* EX1001, 3:51-53. Nevertheless, in my opinion, a POSITA would have been able to understand the plain and ordinary meaning of this limitation and would have found the claim to be obvious over the Hailey/Harrop/Davis combination.

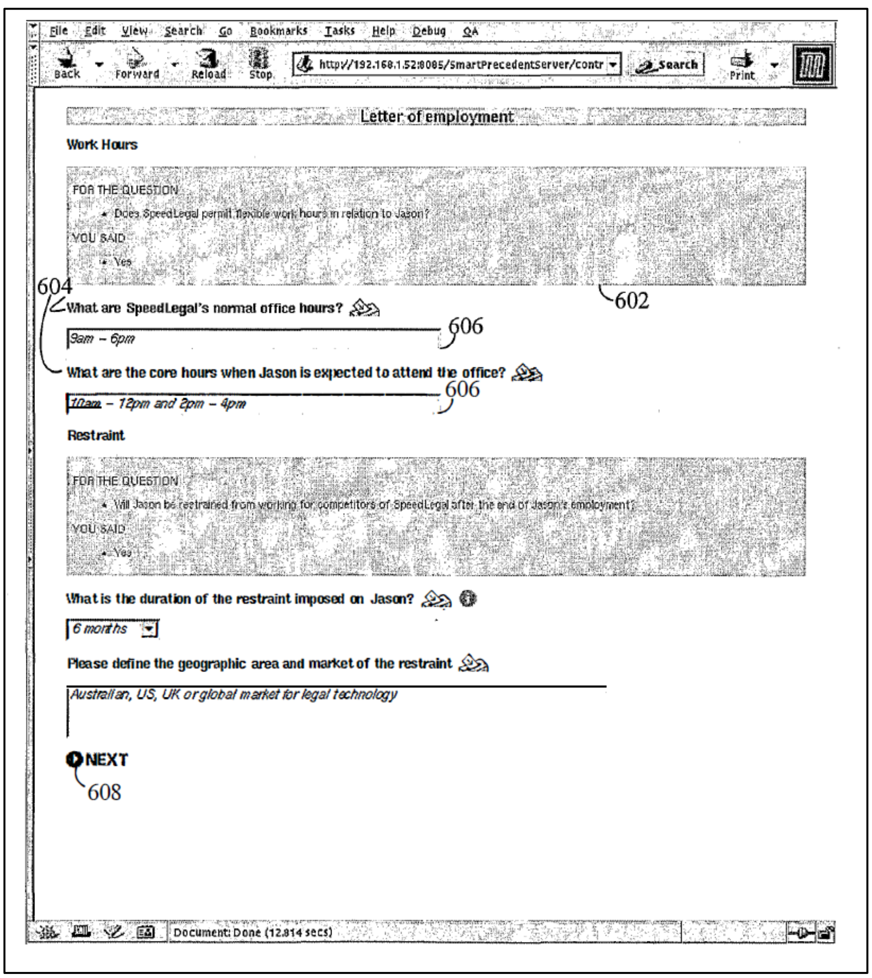
142. Hailey and Harrop both describe using source (master) documents and variables (personalization parameters) to generate output (interactive) documents in HTML and PDF documents. *See* ¶¶106-108, 114-121, *supra* (incorporated here); EX1005, ¶¶39, 40-41, 64, Figs. 1, 4.; EX1006, 8:1-9:23, 14:7-11, 14:21-15:6, Fig. 7. A POSITA would have understood that a generated HTML or PDF document does not always include the assembly rule used to assemble the generated document.

A POSITA would have considered it obvious to exclude the assembly rules after they are evaluated and only populate the generated document with the object or text content for several reasons. For example, a POSITA would have recognized that including assembly rules in a generated would unnecessarily complicate the generated document and limit the file types that the generated document use. Additionally, a POSITA would have understood that including the assembly rules in the generated document would introduce security risks. First of all, an end-user could manipulate the value of parameters in the generated document relying on the rules to modify the document in ways unintended by the owner of the master document. *See, e.g.*, EX1020. Additionally, URL parameters can expose sensitive information, which is undesirable. *See, e.g.*, EX1021. Moreover, a POSITA would have understood that the typical approach would be to use the assembly rules to decide what content is included in the final generated document and leave the rule itself in the source/master document. It is thus my opinion that a POSITA would have found it obvious to use assembly rules from modular items to select content and create an interactive document, and not import those assembly rules into the final, created document.

5. Claim 4: The system of claim 1, wherein the one or more operations further comprise providing a graphical user interface (GUI) for receiving the one or more personalization parameters.

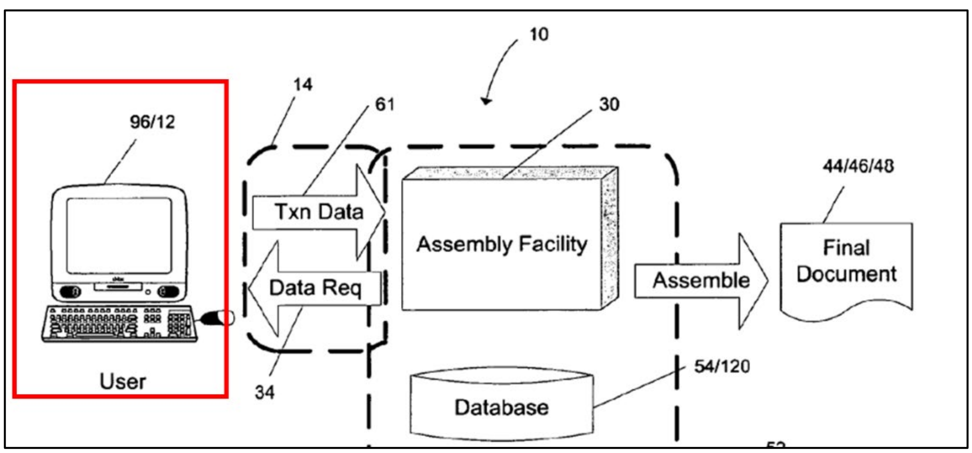
143. In my opinion, a POSITA would have understood that the Hailey/ Harrop/Davis combination renders obvious all the limitations of claim 1 and further renders claim 4 obvious. ¶¶106-137, *supra* (incorporated here).

144. Harrop teaches that a web browser (*i.e.*, a GUI) displays interview questions to solicit answers from a user and generate an instance document. EX1006, 14:21-31, Fig. 6 (reproduced below).



A POSITA would have understood that web browser is, or at least includes, a graphical user interface (GUI).

145. In addition, Hailey teaches that an origination platform 12 is used to solicit data from a user, which data (personalization parameter) is used to select the appropriate content for the generated document. *See* ¶¶83-86, 114-121, *supra* (incorporated here); EX1005, ¶¶39, 40-41, 64, Fig. 1. As shown in figure 4A (annotated below), Hailey describes that a computer is used to input data into the origination platform 96/12. EX1005, Fig. 4A, ¶66 ("A user 96 interacting with the origination platform 12 enters transaction data 61 (which may include, for example, a transaction identifier.").



146. Accordingly, it is my opinion that a POSITA would have understood that that the Hailey/Harrop/Davis combination renders claim 4 obvious

6. Claim 7: The system of claim 1, wherein the master document is stored in a pre-defined document schema.

147. In my opinion, a POSITA would have understood that the Hailey/ Harrop/Davis combination renders obvious all the limitations of claim 1 and further renders claim 7 obvious. ¶¶106-137, *supra* (incorporated here).

148. Hailey teaches storing objects (content) using an XML schema. EX1005, ¶¶45-58, 87, Fig. 18. Similarly, Harrop teaches that source documents are in an XML format which can be validated against an XML schema. EX1006, 24:26-28:43. An XML schema is a pre-defined document schema. Accordingly, a POSITA would have understood that both Hailey and Harrop teach storing the master document in a pre-defined document schema (*i.e.*, an XML schema).

7. Claim 8: The system of claim 7, wherein the master document is stored as structured data in a document database.

149. In my opinion, a POSITA would have understood that the Hailey/ Harrop/Davis combination renders obvious all of the limitations of claims 1 and 7, and further renders claim 8 obvious. ¶¶106-137, 147-148, *supra* (incorporated here).

150. Hailey teaches storing content (*i.e.*, objects) in a knowledge base (shown as a runtime database 120 in FIGS. 8 and 4) from which the content (*i.e.* objects) for a particular document can be extracted. EX1005, ¶95. Figure 4 (reproduced below) demonstrates that this data is stored in a structured manner.

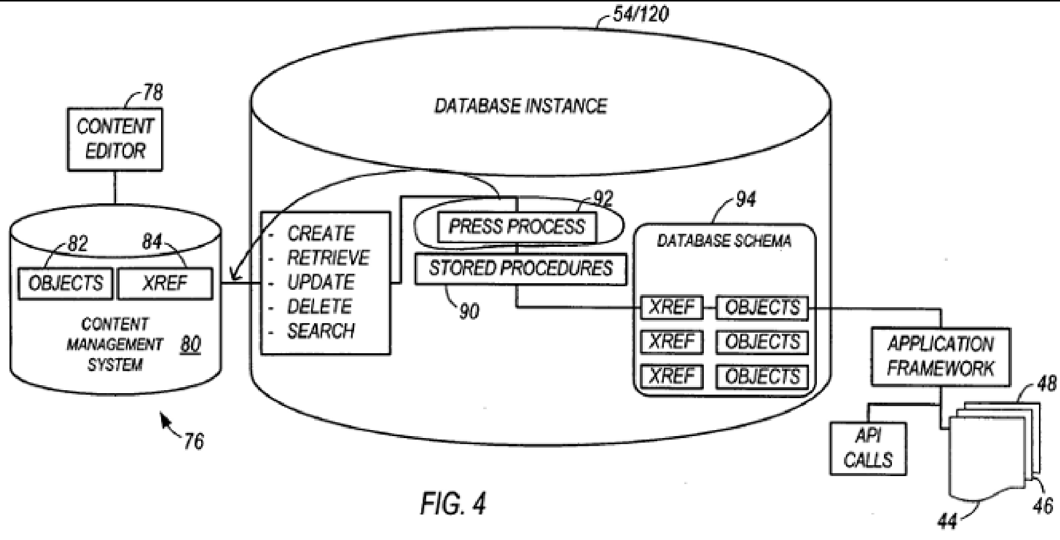


FIG. 4

Similarly, Harrop teaches that the source documents 208 XML documents (*i.e.*, structured data files) in data files 110, which are stored in a hard drive (*i.e.*, database). EX1006, 7:21-31.

Accordingly, it is my opinion that a POSITA would have understood that the Hailey/Harrop/Davis combination renders claim 8 obvious.

8. <u>Claim 9: The system of claim 8, wherein the structured data further stores an assembly rule associated with the one or more modular items.</u>

151. In my opinion, a POSITA would have understood that the Hailey/ Harrop/Davis combination renders obvious all of the limitations of claims 1, 7, and 8, and further renders claim 9 obvious. ¶¶106-137, 147-148, 149-150, *supra* (incorporated here).

152. As discussed regarding limitation 1[B1], Hailey teaches that content (modular items) are stored in the database as groups of XREF (rules) and objects.

See ¶¶101-108, *supra* (incorporated here); EX1005,¶¶39, 40-41, 64. Figure 4 of Hailey (annotated below), depicts the content (modular items) includes the assembly rules (XREF) and objects and, therefore, the XREF assembly rules are associated with each modular item (content). EX1005, ¶64, Fig. 4.

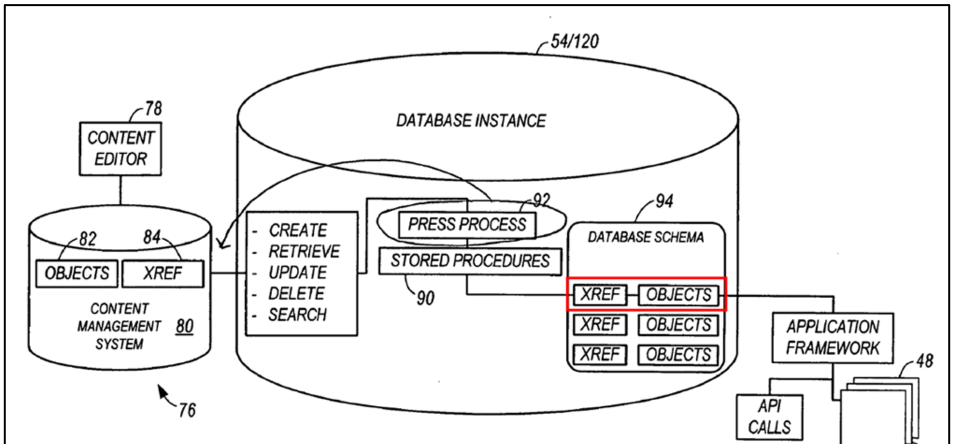


FIG. 4

153. Accordingly, it is my opinion that a POSITA would have that the Hailey/Harrop/Davis combination renders claim 9 obvious.

9. <u>Claim 10</u>

154. Claim 10 is directed to a computer-implemented method, reciting limitations that are substantially identical to the recitations in claim 1. Thus, for at least the same reasons as claim 1, it is my opinion that a POSITA would have found that the Hailey/Harrop/Davis combination renders claim 10 obvious. ¶¶97-137, *supra* (incorporated here).

a. Limitation 10[PRE]: A computer-implemented method for creating an interactive document, comprising:

155. As discussed regarding limitations 1[pre] and 1[A], it is my opinion that a POSITA would have understood that the Hailey/Harrop/Davis combination renders this limitation obvious. *See* ¶¶106-113, *supra* (incorporated here).

b. Limitation 10[A1]: (a) generating an interactive document comprising one or more modular items, wherein the one or more modular items are pulled from a master document based at least in part on one or more personalization parameters,

156. As discussed regarding limitations 1[B1], it is my opinion that a POSITA would have understood that the Hailey/Harrop/Davis combination renders this limitation obvious. *See* ¶¶114-121, *supra* (incorporated here).

c. Limitation 10[A2]: and wherein the interactive document comprises a tracking feature to collect data indicative of user interaction with the interactive document;

157. As discussed regarding limitations 1[B2], it is my opinion that a POSITA would have understood that the Hailey/Harrop/Davis combination renders this limitation obvious. *See* ¶¶122-124, *supra* (incorporated here).

d. Limitation 10[B]: (b) creating a link between the interactive document and the master document, wherein the link comprises the one or more personalization parameters;

158. As discussed regarding limitations 1[C], it is my opinion that a POSITA would have understood that the Hailey/Harrop/Davis combination renders this limitation obvious. *See* ¶¶125-130, *supra* (incorporated here).

e. Limitation 10[C1]: (c) rendering the interactive document with aid of a content renderer,

159. As discussed regarding limitations 1[D1], it is my opinion that a POSITA would have understood that the Hailey/Harrop/Davis combination renders this limitation obvious. *See* ¶¶131-133, *supra* (incorporated here).

f. Limitation 10[C2]: wherein the content renderer is configured to use the link to apply a change to the one or more modular items based on a change to the master document.

160. As discussed regarding limitations 1[D2], it is my opinion that a POSITA would have understood that the Hailey/Harrop/Davis combination renders this limitation obvious. *See* ¶¶134-137, *supra* (incorporated here).

10. Claim 11: The computer-implemented method of claim 10, wherein the one or more modular items from the master document are embedded with an assembly rule.

161. As discussed regarding claim 2, it is my opinion that a POSITA would have understood that the Hailey/Harrop/Davis combination renders this claim obvious. *See* ¶¶138-139, *supra* (incorporated here).

11. <u>Claim 12: The computer-implemented method of claim 11, wherein the one or more modular items in the interactive document are not embedded with the assembly rule.</u>

162. As discussed regarding claim 3, it is my opinion that a POSITA would have understood that the Hailey/Harrop/Davis combination renders this claim obvious. *See* ¶¶140-142, *supra* (incorporated here).

12. <u>Claim 13: The computer-implemented method of claim 10, wherein the one or more operations further comprise providing a graphical user interface (GUI) for receiving the one or more personalization parameters.</u>

163. As discussed regarding claim 4, it is my opinion that a POSITA would have understood that the Hailey/Harrop/Davis combination renders this claim obvious. *See* ¶¶143-146, *supra* (incorporated here).

13. <u>Claim 16: The computer-implemented method of claim 10, wherein the master document is stored in a pre-defined document schema.</u>

164. As discussed regarding claim 7, it is my opinion that a POSITA would have understood that the Hailey/Harrop/Davis combination renders this claim obvious. *See* ¶¶147-148, *supra* (incorporated here).

14. <u>Claim 17: The computer-implemented method of claim 16, wherein the master document is stored as structured data in a document database.</u>

165. As discussed regarding claim 8, it is my opinion that a POSITA would have understood that the Hailey/Harrop/Davis combination renders this claim obvious. *See* ¶¶149-150, *supra* (incorporated here).

15. Claim 18: The computer-implemented method of claim 17, wherein the structured data further stores an assembly rule associated with the one or more modular items.

166. As discussed regarding claim 9, it is my opinion that a POSITA would have understood that the Hailey/Harrop/Davis combination renders this claim obvious. *See* ¶¶151-152, *supra* (incorporated here).

B. *[Alternate] Ground 2: Claims 1-5, 7-14, and 16-18 are rendered obvious by Hailey, Harrop, Davis, and McKinlay*

167. This Ground is included as an alternative to the opinions I presented regarding limitations 1[C] above. All of my other opinions regarding the other limitations of claim 1 and all other claims follow the same rationale presented above. ¶¶97-137, *supra* (incorporated here). Accordingly, as discussed in more detail below, this ground focuses on the addition of McKinlay which teaches a link that comprises personalization parameters, as required by limitation 1[C].

i. Reasons to Combine

1. Analogous Art

168. In my opinion a POSITA would have found that McKinlay is analogous art to Haiely, Harrop, Davis, and the ’290 Patent. McKinlay teaches a system that initiates a download of a software component based on custom parameters (*e.g.*, download authorization and preferences). EX1008, abstract. Similar to Harrop’s pop up preview, McKinlay teaches a user interface (web browser) and uses URL parameters and JavaScript to generate a pop-up window with specific information.

EX1008, ¶86. Thus, Harrop and McKinlay both teach similar to solutions (*e.g.* showing a pop-up window) to similar problems (*e.g.*, displaying specific information to a user). Accordingly, a POSITA would have considered McKinlay to be analogous art to Hailey, Harrop, Davis, and the ’290 Patent. In particular, a POSITA would have recognized McKinlay’s teaching regarding creating a pop-up with specific information is analogous to Harrop’s teaching of hypertext anchors that generate pop-up windows with the associated question and answer.

1. <u>Motivation to combine</u>

169. A POSITA would have been motivated to implement Harrop’s pop-up window and hypertext anchor as a URL link with URL parameters and JavaScript coding, as taught by McKinlay. Viewing Harrop’s teaching of the hypertext anchor, a POSITA would have known that the hypertext anchors, pop up window, and resetting the associated questions could be implemented in several different ways including URL parameters and JavaScript. Indeed, Harrop teaches using “JavaScript and the anchor elements HTML” but does not expressly describe using URL parameters in the anchor element’s HTML code. McKinlay, however, describes generating a pop-up using JavaScript and URL parameters and includes example code to implement those features. EX1008, ¶86. A POSITA would have found it obvious to combine the teachings of McKinlay and Harrop to implement the hypertext anchors taught by Harrop using URL parameters and JavaScript, as taught

by McKinlay, to display to the user the questions and answers that led to the inclusion of the linked text and allow resetting of the question, as taught by Harrop.

1. Reasonable Expectation of Success

170. A POSITA would have found it routine and predictable to implement Harrop's hypertext anchor, pop up, and question-resetting using McKinlay's URL parameters and JavaScript. McKinlay provides explicit examples of how to implement pop ups with specific information and a POSITA would have readily understood that similar programming could be used to implement Harrop's described features. Implementing code similar to McKinlay's in Harrop's system would have merely required including known technologies (*i.e.*, URL and JavaScript coding) in known systems (Harrop's generated instance document and system). Accordingly, in view of the clear teachings of both Harrop and McKinlay, it is my opinion that a POSITA would have found had a reasonable expectation of success in combining the teachings of McKinlay with Hailey, Harrop, and/or Davis.

i. Claim 1

1. Limitations 1[pre], 1[A], 1[B1], 1[B2], 1[D1], and 1[D2]

171. As discussed above in regard to Ground 1, it is my opinion that a POSITA would have understood that the Hailey/Harrop/Davis combination renders obvious all of the features recited in limitations 1[PRE], 1[A], 1[B1], 1[B2], 1[D1], and 1[D2]. It is also my opinion that a POSITA would have understood that

combination of Hailey, Harrop, Davis, and McKinlay render the same limitations obvious for the same reasons presented above. *See* ¶¶106-124, 131-137, *supra* (all incorporated here).

a. <u>Limitation 1[C]: (b) creating a link between the interactive document and the master document, wherein the link comprises the one or more personalization parameters;</u>

172. As discussed above regarding Ground 1, it is my opinion that a POSITA would have understood that the Hailey/Harrop/Davis combination renders obvious all the features recited in limitations 1[C]. ¶¶125-130, *supra* (incorporated here).

173. In addition, or in the alternative to the reasons stated above, and to the extent that the claimed "link" must be a specific data structure or specific data reference that "comprises the one or more parameters," a POSITA would have found it obvious to implement Harrop's hyperlink anchors as a URL hyperlink containing web parameters that specify the answers (personalization parameters) associated with the text included in the generated instance document, as taught by McKinlay.

174. URL web parameters were well known both by the time the '290 Patent was filed and, in 2006, when Harrop was published. In my opinion, a POSITA would have known that URL web parameters can be used to embed particularized information into a URL link. A POSITA would have understood, upon reading Harrop's disclosure regarding hypertext anchors, that an obvious way to implement

the hypertext anchors would be to embed a URL containing parameters that includes the question and answer associated with the hyperlinked text so that, on mouse over, the pop up is created and displayed as intended by Harrop.

175. Indeed, McKinlay teaches an example of using a URL that includes a defined parameter "yic=YIC_HIC_Site_Home" combined with the JavaScript function "window.open()" to define a link that opens a pop-up window displaying the webpage that includes the customized defined parameter. EX1008, ¶¶86-94. Similarly, Harrop itself contemplates using "JavaScript and the anchor elements HTML" to implement its hypertext anchor and pop-up window. EX1006, 68:14-18.

176. In my opinion, a POSITA would have found it obvious to implement the hypertext anchor and pop-up window taught by Harrop using a URL anchor that includes personalization parameters (*i.e.*, web parameters), as taught by McKinlay. *See* ¶¶168-170, *supra* (incorporated here).

177. Thus, it is my opinion that a POSITA would have understood that limitation 1[C] is obvious over the Harrop/Hailey/Davis combination and, in the alternative, the Harrop/Hailey/Davis/McKinlay combination.

i. <u>Claims 2-4, 7-14, and 16-18</u>

178. As discussed above regarding Ground 1, it is my opinion that a POSITA would have understood that the Hailey/Harrop/Davis combination renders obvious all the features recited in claims 2-4, 7-14, and 16-18. *See* ¶¶138-166, *supra* (all

incorporated here). For the same reasons discussed above, it is also my opinion that a POSITA would have understood that the Hailey/Harrop/Davis/McKinlay combination renders obvious all the features recited in these same claims. More specifically, a POSITA would have recognized the addition of McKinlay's teachings would not alter any of the features of Hailey, Harrop, or Davis discussed regarding claims 2-4, 7-14, and 16-18.

C. *Ground 3A: Claims 5-6 and 14-15 are rendered obvious by Harrop, Hailey, and Davis (and, alternatively McKinlay), in further view of Lo*

1. <u>Reasons to Combine</u>

a. <u>Analogous Art</u>

179. It is my opinion that a POSITA would have found that Lo is analogous art to the Harrop, Hailey, and Davis, as well as the '290 Patent. Lo is in the same field of endeavor, namely document assembly and creation. EX1009, ¶2 ("methods and processes for creation, editing, and of documents composed of text and graphics and, in printing particular, to the creation, editing and printing of variable data documents."). Accordingly, it is my opinion that a POSITA would have considered Lo to be analogous art to Hailey, Harrop, Davis, and the '290 Patent.

b. <u>Motivation to combine</u>

180. In my opinion, a POSITA would have been motivated to combine the teachings of Lo with the teachings of Hailey, Harrop, and Davis. Lo teaches that

"[l]ive proofing is advantageous over a separate merging process because errors in the design of the template are likely to be found earlier through interactive navigation." EX1009, ¶381. Similarly, a POSITA would have recognized that Hailey and Harrop each appreciate the importance of previewing and/or catching errors before a final output document is generated. For example, Harrop describes document previews a pop-up windows to assist a user in adjusting the final document. EX1006, 15:8-16, 67:28-68:18. Likewise, Hailey's system includes a mechanism to allow a user a "change or replace paragraphs or sections, add content, etc." in a generated document. EX1005, ¶69.

181. Accordingly, it is my opinion that a POSITA would have been motivated to combine Lo's teaching of live-proofing previews with the teachings of Hailey, Harrop, and/or Davis to enable a user to find errors in document design at an earlier stage, as taught by Lo.

c. <u>**Reasonable Expectation of Success**</u>

182. It is also my opinion that a POSITA would have had a reasonable expectation of success in modifying the systems of Hailey, Harrop, and/or Davis to include the live-proofing previews taught by Lo. The implementation of previews into a GUI would have required nothing more than incorporating known concepts (*i.e.*, previews in a GUI) into a known systems (*i.e.*, the interfaces taught by Harrop, Hailey, and Davis). In particular, Hailey and Harrop contemplate or teach user

interfaces through which a user inputs data or answers questions relative to the variables that influence which content is included in the generated documents. *See* ¶¶143-146, 163, *supra* (incorporated here). In my opinion, a POSITA would have found it routine to modify the already-existing user interfaces of Hailey, Harrop, and/or Davis to include Lo's teachings regarding the live-proofing preview feature.

2. **Claim 5: The system of claim 4, wherein the GUI is configured to display a preview of the interactive document upon receiving a user input indicative of at least one of the one or more personalization parameters.**

183. In my opinion, a POSITA would have understood that the Hailey/ Harrop/Davis (and/or with McKinlay) combination renders claim 4 obvious. Claim 5 is rendered obvious over Hailey, Harrop, Davis, and, in further view of Lo. *See* ¶¶143-146, 178, *supra* (incorporated here);

184. Lo teaches "live proofing" a template document to "display[] variable data merged with the variable document template interactively and on-demand." EX1009, ¶428. Figure 20 of Lo shows "what part of [a] record would look like if printed" including "extra borders 2004 and colors 2006…to make visible the constraints defined within the variable document template." EX1009, ¶¶380, 428-429, Fig. 20.

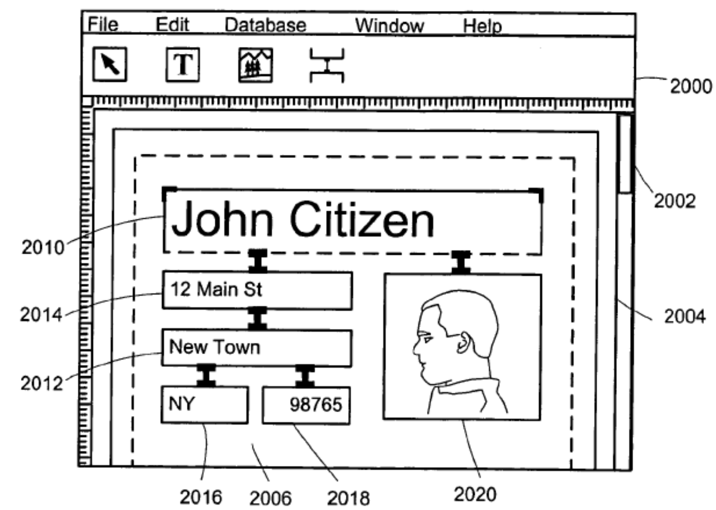


Fig. 20

185. Thus, it is my opinion that a POSITA would have recognized that Lo teaches displaying a preview of an interactive document (as shown in figure 20) where the preview reflects a user input regarding a personalization parameter (*e.g.*, Lo's "variable data").

186. In my opinion, a POSITA would have found it obvious to include Lo's teaching of live-proofing feature in the Hailey/Harrop/Davis combined system to provide the user with a preview of how the generated interactive document would change depending on changes to input variables. *See* ¶¶179-182, *supra* (incorporated here). Accordingly, the Hailey/Harrop/Davis combination as modified by Lo renders claim 5 obvious.

3. Claim 6: The system of claim 5, wherein the preview comprises automatically navigating to at least one of the one or more modular items corresponding to the user input.

187. In my opinion, a POSITA would have understood that the Hailey/ Harrop/Davis (and/or McKinlay) combination in view of Lo renders claim 5 obvious. A POSITA would have also understood that same combination renders claim 6 obvious. *See* ¶¶183-186, *supra* (incorporated here).

188. As discussed above in reference to claim 5, Figure 20 of Lo shows a preview that automatically depicts the portion of the interactive document that is modified based on the changed input variable. EX1009, Fig. 20, ¶428; ¶¶183-186, *supra*. Specifically, Lo teaches that the preview of figure 20 shows "what ***part*** of that ***record*** would look like." EX1009, ¶428. Indeed, Lo describes its live proofing as "interactive navigation." EX1009, ¶381. Accordingly, it is my opinion that a POSITA would have understood that Lo's live-proofing preview automatically navigates to the "part of [the document]" that corresponds to the part of the record data modified by the user.

4. Claim 14: The computer-implemented method of claim 13, wherein the GUI is configured to display a preview of the interactive document upon receiving a user input indicative of at least one of the one or more personalization parameters.

189. For the same reasons discussed regarding claims 10, 13, 5, and 6 above, it is my opinion that a POSITA would have understood that the Hailey/Harrop/Davis

combination in view of Lo renders claim 14 obvious. *See* ¶¶154-160, 163, 183-188, *supra* (incorporated here).

5. Claim 15: The computer-implemented method of claim 14, wherein the preview comprises automatically navigating to at least one of the one or more modular items corresponding to the user input.

190. For the same reasons discussed in regard to claims 10, 13, 5, 6, and 14 above, it is my opinion that a POSITA would have understood that the Hailey/ Harrop/Davis combination in view of Lo renders claim 15 obvious. *See* ¶¶154-160, 163, 183-189, *supra* (incorporated here).

IX. CONCLUSION

191. For these reasons, it is my opinion that claims 1-18 of the ’290 Patent are not novel and, within my understanding as an external technical expert, appear to be unpatentable.

192. I understand that this 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 understand that I may be subject to cross-examination concerning this declaration, and I will appear if required, at the time allotted for cross-examination.

193. I hereby declare that all of the 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 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. I reserve the option to modify my conclusions in the event that authentic contradictory information should become known at a future time.

Executed on this 16th day of September, 2025.

____

Dr. Benjamin B. Bederson, Ph.D.