

UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE PATENT TRIAL AND APPEAL BOARD

FOLEON INC.,

Petitioner

v.

TURTL SURF & IMMERSE LIMITED

Patent Owner

IPR2026-00404
Patent No. 11,379,651

**PETITION FOR *INTER PARTES* REVIEW
UNDER 35 U.S.C. § 312 AND 37 C.F.R. § 42.104**

TABLE OF CONTENTS

I.	Introduction.....	5
II.	Petition Requirements.....	6
	A. Grounds For Standing and Timeliness.....	6
	B. Payment of Fees	7
	C. Identification of Challenge and Relief Requested	7
III.	Technological Background.....	8
IV.	The '651 Patent.....	10
	A. Overview of the '651 Patent.....	10
	B. Prosecution History	16
V.	Level of Ordinary Skill in The Art	16
VI.	Claim Construction.....	17
VII.	Summary of the Prior Art	17
	A. Harrop.....	17
	B. Hailey	20
	C. Davis.....	22
	D. Lo.....	23
VIII.	GROUND FOR UNPATENTABILITY	25
	A. Ground 1: Claims 1-6, 8-10, and 13-20 are rendered obvious by Harrop and Hailey	25
	i. Reasons to Combine	25
	a. Analogous Art.....	25
	b. Motivation to Combine.....	26
	c. Reasonable Expectation of Success.....	27
	ii. Claim 1	28
	a. Limitation 1[PRE]: A method for automatically assembling an interactive document comprising:	28
	b. Limitation 1[A]: embedding a first set of rules into a first modular item of a first document, wherein the first document comprises a first set of modular items:.....	30

c.	Limitation 1[B]: receiving one or more parameters for personalizing the first document to generate a derivative document;	32
d.	Limitation 1[C1]: selecting to display or hide at least the first modular item from the first document based at least in part on a value of the one or more parameters and the first set of rules or.....	34
e.	Limitation 1[C2.A]: pull a second modular item from a second document based at least in part on the value of the one or more parameters and a second set of rules embedded with the second modular item,.....	37
f.	Limitation 1[C2.B]: wherein the second document comprises a second set of modular items, and wherein the first document and the second document are master documents that one or more of the first set of modular items and one or more of the second set of modular items are embedded with rules; and.....	38
g.	Limitation 1[D1]: generating the derivative document by assembling the second modular item with the first document or by displaying or hiding the at least first modular item from the first document, wherein.....	39
h.	Limitation 1[D2]: the derivative document comprises a plurality of modular items that are not embedded with rules.	41
iii.	Claim 2: The method of claim 1, wherein the first modular item is stored in a structured data that defines the first set of rules.	43
iv.	Claim 3: The method of claim 1, wherein the first set of rules define a link between the first modular item and another document or a rule for assembling the first modular item into a derivative document.....	44
v.	Claim 4: The method of claim 3, wherein the value of the one or more parameters is specified by a user via a user interface.....	45

vi.	Claim 5: The method of claim 1, wherein the user interface displays one or more personalization fields that are automatically generated based on the first set of rules.....	46
vii.	Claim 6: The method of claim 5, wherein at least one of the one or more personalization fields is generated based on the second set of rules associated with the second modular item..	47
viii.	Claim 8: The method of claim 6, wherein the value of the one or more parameters is specified by a user in a text file.....	48
ix.	Claim 9: The method of claim 1, wherein the first set of rules includes a mapping relationship that maps the first modular item to a style for rendering the first modular item...	49
x.	Claim 10: The method of claim 1, wherein the derivative document is displayed to a reader using the mapped style.....	50
xi.	Claim 13: The method of claim 1, further comprising storing the one or more parameters and a reference to the first document for updating the derivative document.....	51
xii.	Claim 14: The method of claim 1, further comprising converting the derivative document into a master document by storing the one or more parameters as a set of rules associated with the derivative document.	54
xiii.	Claim 15	56
a.	Limitation 15[PRE]	56
b.	Limitation 15[A1]	57
c.	Limitation 15[B]	57
d.	Limitation 15[C1]	57
e.	Limitation 15[C2]	57
f.	Limitation 15[C3]	58
g.	Limitation 15[D]	58
h.	Limitation 15[E]	58
xiv.	Claim 16: The method of claim 15, further comprising providing a first user interface for receiving a user input to generate the first set of rules.	58
xv.	Claim 17: The method of claim 16, wherein the first user interface displays a plurality of fields for defining the first set of rules.	59

xvi.	Claim 18: The method of claim 15, further comprising providing a second user interface for receiving user input to customize the one or more modular items.....	60
xvii.	Claim 19: The method of claim 15, wherein the first set of rules embedded with the at least one [modular] item of the first document includes a mapping relationship that maps the at least one item to the style.....	60
xviii.	Claim 20: The method of claim 15, further comprising providing a user interface for receiving the instruction for personalizing the first document.....	60
B.	Ground 2: Claims 11 and 12 are rendered obvious by Hailey and Harrop in further view of Davis	61
i.	Technology Background – User Tracking.....	61
ii.	Reasons to Combine	62
a.	Analogous Art.....	62
b.	Motivation to combine.....	62
c.	Reasonable Expectation of Success.....	64
iii.	Claim 11: The method of claim 10, further comprising collecting data about the reader's interaction with the derivative document to extract insight data.	65
iv.	Claim 12: The method of claim 11, further comprising receiving an instruction for personalizing the first modular item or the second modular item.....	66
C.	Ground 3: Claim 7 is rendered obvious by Harrop and Hailey in further view of Lo.....	66
i.	Reasons to Combine	66
a.	Analogous Art.....	66
b.	Motivation to combine.....	67
c.	Reasonable Expectation of Success.....	68
ii.	Claim 7: The method of claim 6, further comprising: upon receiving an input in a given personalization field, displaying at least a portion of the content from the first document that is related to the given personalization field.	68
IX.	Discretionary Denial Is Not Warranted.....	70
X.	Conclusion	71

Petitioner Mandatory Notices	72
------------------------------------	----

Petitioner's Exhibit List¹

Exhibit No.	Description
1001	U.S. Patent No. 11,379,651 B2 to Mason
1002	Prosecution History of U.S. 11,379,651
1003	Declaration of Ben Bederson
1004	<i>Curriculum Vitae</i> of Ben 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	[reserved]
1009	U.S. Patent Publication No. 2005/0094205 A1 to Lo et al.
1010	U.S. Patent No. 4,712,174 to Minkler
1011	U.S. Patent 5,956,687 to Wamsley et al. (“Wamsley”)
1012	“How Users Read on the Web”
1013	<i>Information on RFC 3986</i> , 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”

¹ Petitioner cites to exhibits' original page numbers. Unless otherwise noted, all emphasis in any quoted material has been added.

Exhibit No.	Description
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
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.. <i>The World-Wide Web</i> . 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”).
1023	<i>Foleon Inc. v. Turtl Surf & Immerse Limited</i> , IPR2025-01525, Paper 6 (PTAB Jan. 28, 2026).
1024	<i>Foleon Inc. v. Turtl Surf & Immerse Limited</i> , IPR2025-01525, EX2001 (PTAB Jan. 28, 2026).
1025	Prosecution History of U.S. Patent No. 11,361,144

Claim Appendix

Claim Limitation	Claim Language
1[PRE]	A method for automatically assembling an interactive document comprising:
1[A]	embedding a first set of rules into a first modular item of a first document, wherein the first document comprises a first set of modular items;
1[B]	receiving one or more parameters for personalizing the first document to generate a derivative document;
1[C1]	selecting to display or hide at least the first modular item from the first document based at least in part on a value of the one or more parameters and the first set of rules
1[C2.A]	pull a second modular item from a second document based at least in part on the value of the one or more parameters and a second set of rules embedded with the second modular item,
1[C2.B]	wherein the second document comprises a second set of modular items, and wherein the first document and the second document are master documents that one or more of the first set of modular items and one or more of the second set of modular items are embedded with rules; and
1[D1]	generating the derivative document by assembling the second modular item with the first document or by displaying or hiding the at least first modular item from the first document,
1[D2]	wherein the derivative document comprises a plurality of modular items that are not embedded with rules.
2	The method of claim 1, wherein the first modular item is stored in a structured data that defines the first set of rules.

3	The method of claim 1, wherein the first set of rules define a link between the first modular item and another document or a rule for assembling the first modular item into a derivative document.
4	The method of claim 3, wherein the value of the one or more parameters is specified by a user via a user interface.
5	The method of claim 1, wherein the user interface displays one or more personalization fields that are automatically generated based on the first set of rules.
6	The method of claim 5, wherein at least one of the one or more personalization fields is generated based on the second set of rules associated with the second modular item.
7	The method of claim 6, further comprising: upon receiving an input in a given personalization field, displaying at least a portion of the content from the first document that is related to the given personalization field.
8	The method of claim 6, wherein the value of the one or more parameters is specified by a user in a text file.
9	The method of claim 1, wherein the first set of rules includes a mapping relationship that maps the first modular item to a style for rendering the first modular item.
10	The method of claim 1, wherein the derivative document is displayed to a reader using the mapped style.

11	The method of claim 10, further comprising collecting data about the reader's interaction with the derivative document to extract insight data.
12	The method of claim 11, further comprising receiving an instruction for personalizing the first modular item or the second modular item.
13	The method of claim 1, further comprising storing the one or more parameters and a reference to the first document for updating the derivative document.
14	The method of claim 1, further comprising converting the derivative document into a master document by storing the one or more parameters as a set of rules associated with the derivative document.
15[pre]	A method for personalizing an interactive document comprising:
15[A.1]	(a) creating one or more modular items of a first document, wherein at least one modular item is embedded with a first set of rules defining a link between the at least one modular item and a second document or
15[A.2]	a first set of rules for assembling the at least one modular item into a derivative document
15[B]	(b) receiving an instruction for personalizing the first document;
15[C.1]	(c) assembling a plurality of modular items to generate the derivative document of the first document,
15[C.2]	wherein the plurality of modular items comprise the at least one item displayed or hidden from the given document based on the first set of rules and the instruction or

15[C.3]	the plurality of modular items are pulled from the first document and the second document based at least in part on the first set of rules embedded with the at least one modular item of the first document, wherein the first document and the second document are master documents that one or more modular items of the second document are embedded with rules; and
15[D]	(iii) rendering the derivative document using a style associated with the first document and
15[E]	wherein the plurality of modular items of the derivative document are not embedded with rules.
16	The method of claim 15, further comprising providing a first user interface for receiving a user input to generate the first set of rules.
17	The method of claim 16, wherein the first user interface displays a plurality of fields for defining the first set of rules.
18	The method of claim 15, further comprising providing a second user interface for receiving user input to customize the one or more modular items.
19	The method of claim 15, wherein the first set of rules embedded with the at least one item of the first document includes a mapping relationship that maps the at least one item to the style.
20	The method of claim 15, further comprising providing a user interface for receiving the instruction for personalizing the first document.

I. INTRODUCTION

Foleon Inc., (“Foleon” or “Petitioner”) respectfully requests the Board institute *inter partes* review of U.S. Patent No. 11,379,651 (“the ’651 Patent”), which is assigned to Turtl Surf & Immerse Limited (“Turtl” or “Patent Owner”). Petitioner has been accused of infringing the ’651 Patent and is proactively seeking early cancellation of claims 1-20 (“the Challenged Claims”) of the ’651 Patent to protect its product, market, and customers.

The ’651 Patent purports to describe a novel system and method for generating a document based on modular items, personalization parameters (*e.g.*, variables), and rules that govern the inclusion of one or more of the modular items. The independent claims also recite creating a reference for linking the modular items to the generated derivative document based on the personalization parameters. However, none of the features claimed by the ’651 Patent are novel or nonobvious over the prior art.

For example, Hailey and Harrop both describe document generation systems that pre-date the ’651 Patent by fifteen and seventeen years, respectively and, as explained in Ground 1 below, these references together render obvious the features of independent claims 1 and 15, as well as dependent claims 2-6, 8-10, 13-14, and 16-20. Ground 2 explains that the user tracking feature recited in dependent claims 11 and 12 is obvious in view of the combined teachings of Hailey, Harrop, and

Davis. Ground 3 explains that the real-time rendering recited in Claim 7 is obvious over the combination of Hailey, Harrop, and Lo.

In addition, Patent Owner has dedicated a substantial portion of the inventions described in the '651 Patent to the public domain. Specifically, in response to an IPR petition which relied on the same prior art presented here and which challenged the claims of U.S. Patent No. 12,118,290—which shares an identical specification with the '651 Patent—Patent Owner statutorily disclaimed all of the claims of the '290 patent. *See* EX1023, EX1024.

Accordingly, institution of this petition is requested and a Final Written Decision invalidating the Challenged Claims is appropriate.

II. PETITION REQUIREMENTS

A. Grounds For Standing and Timeliness

Petitioner certifies that the '651 Patent is eligible for IPR and that Petitioner is not barred or estopped from requesting IPR. 37 C.F.R. §42.104(a). The '651 Patent was filed on June 15, 2021, and issued on July 5, 2022. It became eligible for IPR on April 5, 2023. *See* 37 C.F.R. §42.102(a)(1). The '651 Patent has been in force for about four years and has been eligible for IPR for just over three years. Furthermore, Petitioner has only been aware of Patent Owner's allegations of infringement since receiving Patent Owner's November 1, 2024 letter alleging infringement, after which Petitioner promptly informed Patent Owner that the '651 Patent was invalid.

As such, Patent Owner cannot reasonably claim settled expectations in the validity of its patent claims.

B. Payment of Fees

Pursuant to 37 C.F.R. §§42.103(a) and 42.15(a), the required fees are submitted herewith. If additional fees are due during this proceeding, the Office is authorized to charge Deposit Account 50-1943.

C. Identification of Challenge and Relief Requested

Pursuant to 37 C.F.R. §§ 42.22(a) and 42.104(b), Petitioner requests a final written decision that the Challenged Claims are unpatentable in view of the analysis below. *See* §VIII, *infra* (incorporated here). The effective filing date of the '651 Patent, for purposes of this Petition only, is assumed to be Jul. 6, 2020 (the “Effective Filing Date”). *See* §IV, *infra* (incorporated here). The following is the prior art that anticipates and/or renders obvious the Challenged Claims.

Exhibit	Description	Operative Date
1005	U.S. Patent Publication No. 2005/0080814 A1 (“Hailey”)	April 14, 2005
1006	International Publication No. WO 03/075191 A1 t (“Harrop”)	September 12, 2003
1007	U.S. Patent No. 5,796,952 (“Davis”)	August 18, 1998
1009	U.S. Patent Publication No. 2005/0094205A1 (“Lo”)	May 5, 2005

Each of Hailey, Harrop, Davis, and Lo are prior art under §102(a)(1) because each reference is a printed patent publication prior to the earliest effective filing date of the '651 Patent and none of the exceptions of §102(b)(1) or §102(b)(2) apply.

Petitioner requests cancellation of the Challenged Claims on the following grounds:

Ground	Claims	Obviousness under §103
1	1-6, 8-10, 13-20	Obvious in view of Harrop and Hailey
2	11 and 12	Obvious in view of Harrop, Hailey, and Davis
3	7	Obvious in view of Harrop, Hailey, and Lo.

III. TECHNOLOGICAL BACKGROUND

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, Abstract. Even four decades ago, Minkler describes that “[i]t 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, 1:12-14. 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, 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, Abstract. EX1003, ¶54.

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, Abstract. Similarly, Hailey teaches that “[d]ocument production systems that insert customer-specific content into a pre-fabricated document are known.” EX1005, ¶3. Hailey’s system, among other things, “creat[es] [] documents by assembling document components and applying precedence and rules to create a desired document.” EX1005, ¶2; EX1003, ¶55.

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. EX1003, ¶56.

As recognized by Minkler, Wamsley, Hailey, Harrop, and others, it was known to use 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, 16:53-17:32. Wamsley teaches generating a demand letter with rules for which paragraphs and information to include in the letter. EX1011, 23:6-39. As discussed in §VII.A-B below, Harrop and Hailey also describe using assembly rules to determine what content to include in a generated document. EX1003, ¶57.

IV. THE '651 PATENT

The '651 Patent issued on July 5, 2022, from Patent Application No. 17/347,899, filed on June 15, 2021, by Patent Owner. *See* EX1001, 1. The '651 Patent claims priority to U.S. Provisional Application No. 63/048,512 which was filed on July 6, 2020. EX1001, 1-2 (field 60). Without conceding the issue, Petitioner treats all the Challenged Claims of the '651 Patent as having an effective filing date of July 6, 2020.

A. Overview of the '651 Patent

The '651 Patent is directed to document assembly systems and methods 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. EX1001, 1:33-39; 2:5-15. These assembled documents may take various forms including “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.*, 4:40-45. EX1003, ¶58.

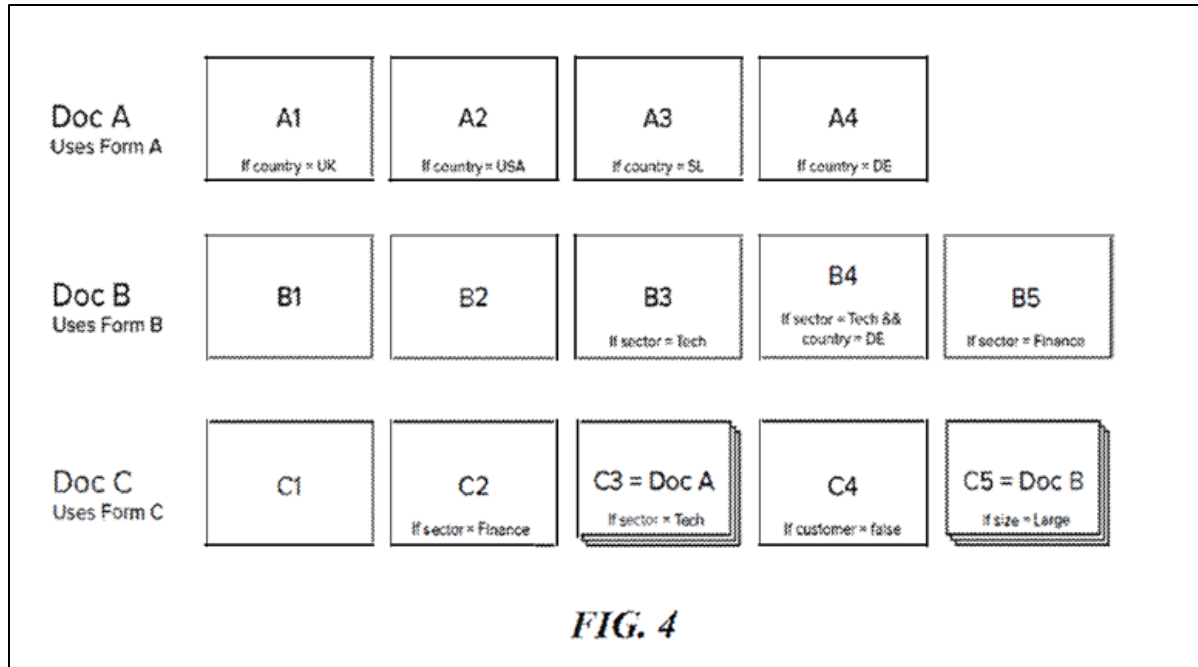
The patent describes these documents as being “interactive” documents and broadly defines “interaction.” In one example, the patent describes interaction when a user has “read, edited, shared, [and/or] saved” a document. EX1001, 23:12-27. In another example, interaction amounts to a user “brows[ing], filter[ing], or select[ing] a content item (e.g., page, video) for viewing.” *Id.*, 25:10-14. EX1003, ¶59-62.

Regarding the creation of personalized interactive documents, the 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, 8:16-21. 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, 15:47-51; EX1003, ¶63.

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, 5:51-56. The patent refers to these “conditions” defined in the assembly rules as “personalization parameters.” EX1001, 5:56-63, 20:1-6. 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, 16:44-47. A user may select personalization options in a personalization field provided to the user via a graphical user interface where the selections allow a document to be customized to a specific customer or reader. EX1001, 8:30-40. Thus, the personalization parameters are variables which influence which content are included in the generated document. EX1003, ¶64.

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, 3:47-49. EX1003, ¶65.



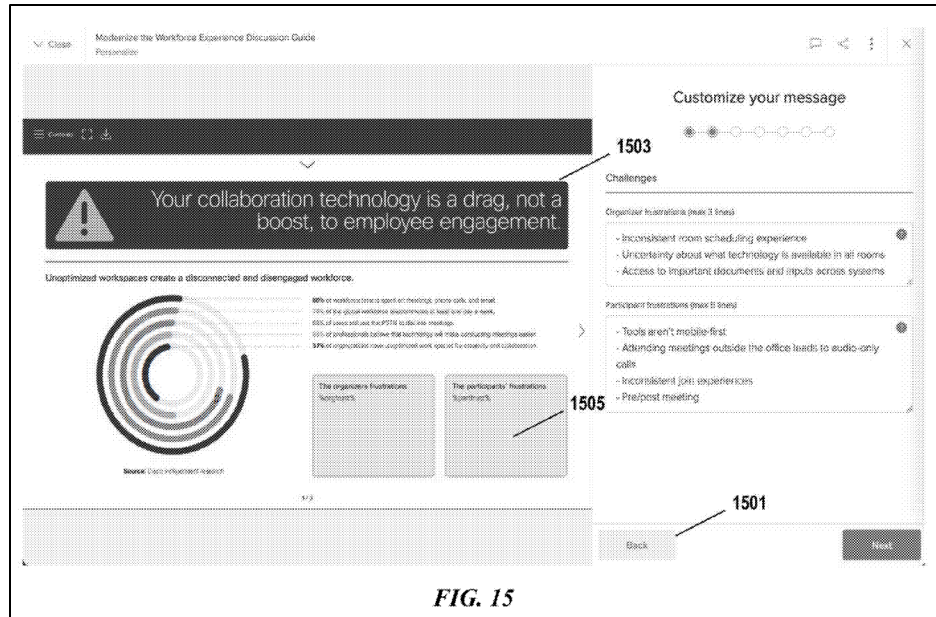
As seen above, page A1 includes the rule “if country = UK” such that page A1 would be included in the generated document if the user location is “UK.” EX1001, 19:1-4. Beyond the depiction of an “embedded” rule shown in figure 4 and other similar figures, the patent does not give an example (e.g., example code, file type, pointer, etc.) of how an assembly rule would be “embedded” in a content item. Yet, the ’651 Patent provides no explanation, example, or discussion of an interactive document either including or excluding these assembly rules. EX1003, ¶66.

The patent describes a “content renderer for rend[er]ing the assembled document.” EX1001, 22:52-54. The function of content renderer is described 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, 22:52-63. However, the patent does not provide any example of what a “content renderer” may be, other than describing the intended function. EX1003, ¶67.

Another aspect of the ’651 Patent is the creation of a “link or connection between the source document and the derivative document.” EX1001, 6:7-9. 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, 19:33-38. 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, 19:38-41. Aside from a reference to the “stored link” and the “link or connection,” the patent offers no elaboration of what form the “link or connection” takes (*e.g.*, whether it is 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.” EX1003, ¶70.

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.” EX1003, ¶68.



EX1001, Fig. 15, 22:39-41. 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, 22:44-47. EX1003, ¶68-69.

Lastly, the ’651 Patent describes that “the system may be capable of converting a personalized document into a master document” by “break[ing] the connection between the personalized document and its original master document.” The patent describes that “breaking the connection” prevents updates to the original master document from affecting the converted derivative documents. EX1001, 20:12-27. The patent describes that an aspect of converting the derivative document into a master document may involve “hardcoding” stored fields as assembly rules. EX1001, 20:23-27. EX1003, ¶71.

B. Prosecution History

The prosecution history of the '651 Patent is relatively uneventful. The original claims were filed on June 15, 2021. EX1002, 46-48. On December 22, 2021 the Examiner rejected the claims for provisional statutory double patenting over then-pending U.S. 11,361,144 (“the '144 patent”) and obviousness based on a combination of references under 35 U.S.C. §103. *Id.*, 115-132. In response, on April 18, 2022 Patent Owner concurrently submitted a terminal disclaimer over the '144 patent, cancelled all original filed claims, and presented new claims which were substantially identical to the then-pending claims of the '144 patent—which were allowed by the same Examiner just 11 days prior on April 7, 2022. EX1002, 172-179; 1025, 187-195. Considering the timing of the allowance of the '144 patent application and the subsequent filing of near-identical claims in the '651 application, the Examiner likely allowed the '651 application based on the similarity and based on the filing of the terminal disclaimer over the '144 patent claims.

V. LEVEL OF ORDINARY SKILL IN THE ART

A person of ordinary skill in the art (“POSITA”) in the field of the '651 Patent, as of its earliest possible filing date of July 6, 2020, would have been someone knowledgeable about and familiar with electronic document assembly systems and methods pertinent to the '651 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. EX1003, ¶¶22-26.

VI. CLAIM CONSTRUCTION

Claim terms in an IPR are construed according to their “ordinary and customary meaning” to those of skill in the art. 37 C.F.R. § 42.100(b); *Phillips v. AWH Corp.*, 415 F.3d 1303 (Fed. Cir. 2005) (en banc). Petitioner submits that, for the purposes of this proceeding and the grounds presented herein, no claim term requires express construction. *Nidec Motor Corp. v. Zhongshan Broad Ocean Motor Co.*, 868 F.3d 1013, 1017 (Fed. Cir. 2017)

VII. SUMMARY OF THE PRIOR ART

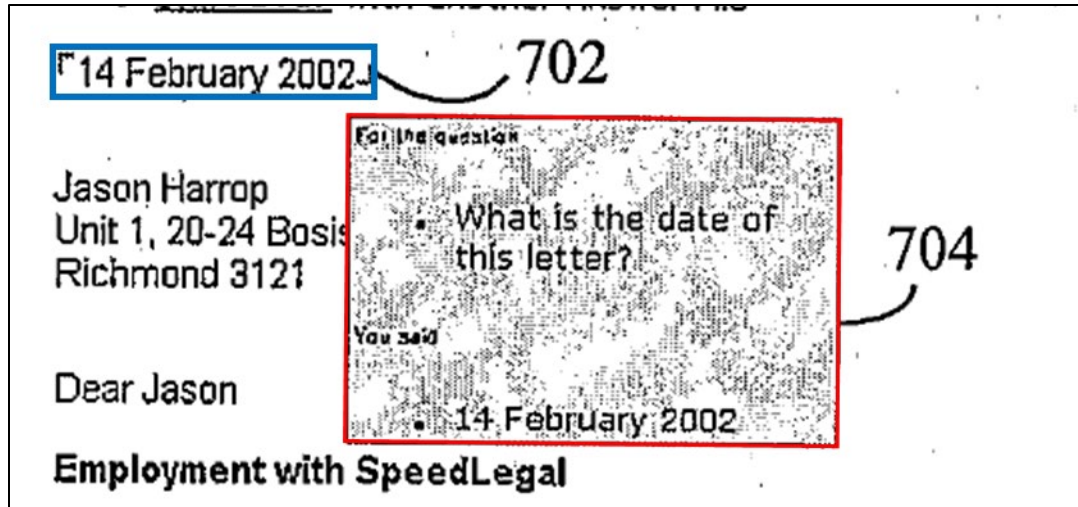
A. Harrop

Just like the '651 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, 8:1-20; EX1003, ¶72.

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

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, 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, 20:4-12. The logic sources and content associated with logic sources together are a modular item in the XML source document. EX1003, ¶74.

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, 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, 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, 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, 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. EX1006, 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, 68:15-19; EX1003, ¶75.

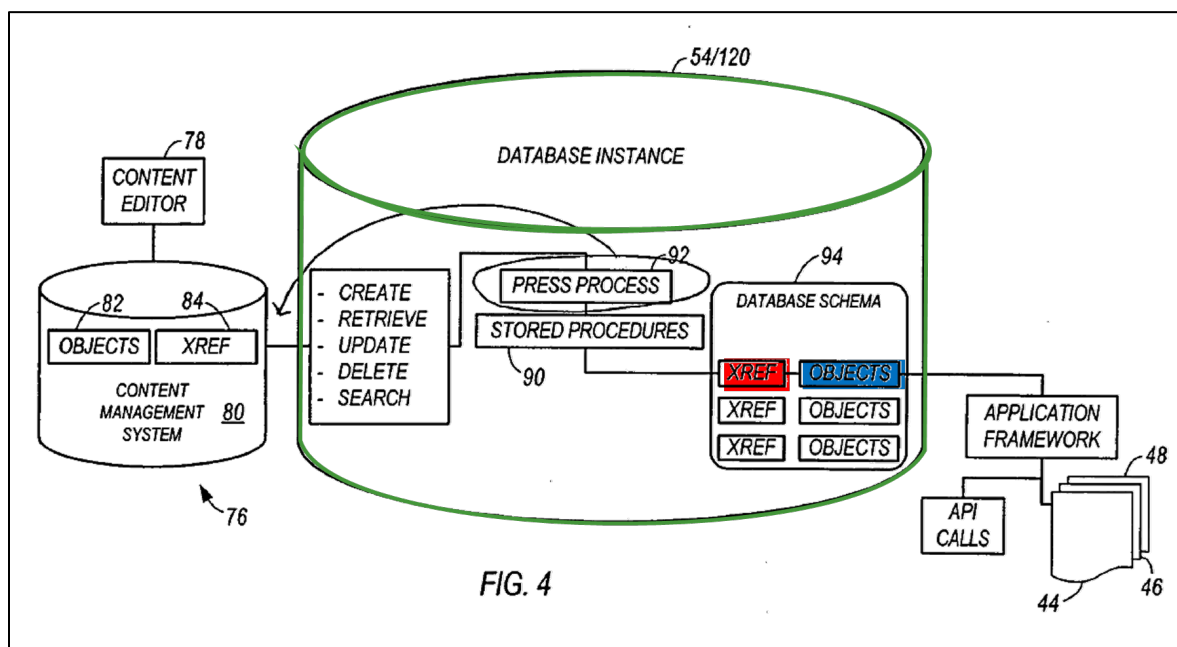
After the processing engine 202 has resolved all the processing-instructions logic, the rendering engine 204 renders the instance document into a desired format, such as HTML. EX1006, 9:30-10:7. Additionally, Harrop’s system can generate a document “in HTML (without controls), XML, or in rich text format (RTF).” EX1006, 15:20-23; EX1003, ¶76.

B. Hailey

Just like the ’651 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; EX1003, ¶77.

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).



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; EX1003, ¶¶78-79.

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, ¶9; EX1003, ¶80.

C. Davis

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.” EX1007, 4:3-8. 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; EX1003, ¶81.

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, to track a user's interaction with the Web page.” *Id.*, 8:12-16. 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. Davis teaches that the server comprises a memory and a processor. *See* EX1007, 7:30-37. EX1003, ¶82.

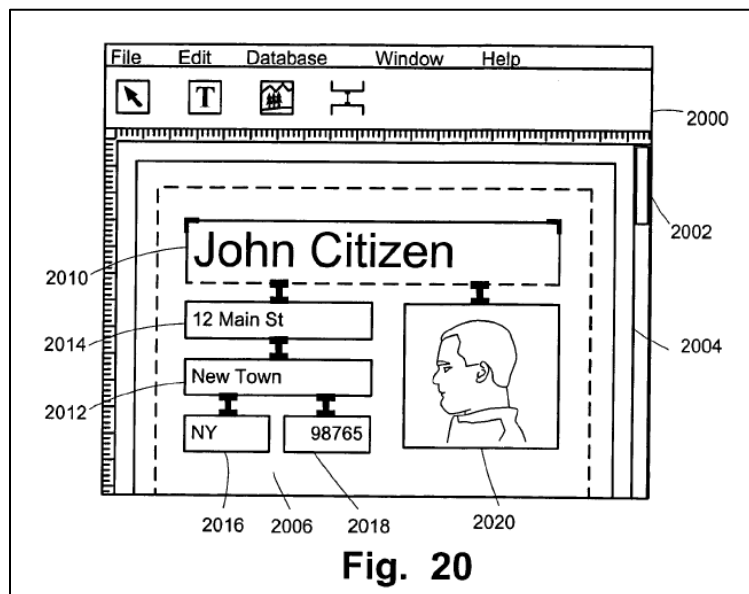
Davis teaches that “[t]he tracked information may be used to assemble resources geared toward the user's interests.” EX1007, 14:47-48. For example, “[b]ased upon the historic user profiles created in the server database, downloading of information to the same client on a subsequent visit to the same 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). EX1003, ¶83.

D. Lo

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, ¶¶6, 84. Lo’s customized documents are “are created by inserting variable data into a document template.” *Id.*, ¶7; EX1003, ¶84.

Further, Lo teaches using “live proofing” to “display[] variable data merged with the variable document template interactively and on-demand.” EX1009, ¶428. 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. EX1003, ¶85.



As seen in Figure 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. EX1003, ¶¶85-86.

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; EX1003, ¶87.

VIII. GROUNDS FOR UNPATENTABILITY

A. Ground 1: Claims 1-6, 8-10, and 13-20 are rendered obvious by Harrop and Hailey

As explained in this section, claims 1-6, 8-10, and 13-20 are obvious based on the combined teachings of Hailey and Harrop. As discussed below, a POSITA would find it obvious to combine the teachings of Hailey and Harrop to arrive at the claimed invention. The combined teachings of Hailey and Harrop are referred to below as the “Hailey-Harrop combination.” EX1003, ¶88.

i. Reasons to Combine

A POSITA would have been motivated to combine Harrop and Hailey to arrive at the features of claims 1-6, 8-10, and 13-20 for numerous reasons presented below. EX1003, ¶89.

a. Analogous Art

Harrop and Hailey are in the same field of endeavor as the ’651 Patent, namely document assembly and creation. *See* EX1001, Abstract; 1:31-51; EX1003, ¶90. 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, 1:4-23. Further, the '651 Patent is classified in International Patent Class G06F, Electronic Digital Data Processing and generally describes document creation and assembly systems and methods for use on the Internet. *See* EX1001 at (51), 4:40-58. Harrop and Hailey are each classified in International Patent Class G06F and 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, 1 (field (51)), ¶42 (output file in HTML format); EX1006, 1 (field (51)), 6:28-7:6 (document assembly system can be accessed by a user over the Internet), 15:18-28 (output document in HTML format); EX1003, ¶90.

b. Motivation to Combine

A POSITA would recognize 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, a POSITA would be 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. EX1003, ¶¶91-92.

c. Reasonable Expectation of Success

The references include similar arrangements of well-understood components and use predictable technology in routine and conventional ways. EX1003, ¶93.

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 onMouseOver event...." EX1006, 68:15-19. Similarly, Hailey teaches reviewing a generated (assembled) document and, if information is missing (e.g., a loan amount), querying the assembly engine 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. 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 '651 Patent. EX1003, ¶94.

ii. **Claim 1**

a. **Limitation 1[PRE]: A method for automatically assembling an interactive document comprising:**

To the extent the preamble is limiting, and without conceding the issue, the Hailey-Harrop combination renders the elements of the preamble obvious by teaching methods for automatically assembling an interactive document. EX1003, ¶95-99.

For example, Hailey and Harrop each teach methods for creating interactive 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 document is then displayed by the web browser, as shown in FIG. 7."). A POSITA would understand 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 the same interactions may be present in a PDF document as well. EX1003, ¶96. Accordingly, a POSITA would understand that Harrop and Hailey both teach methods of creating interactive documents. *Id.*

As discussed above, the '651 Patent describes "interaction" as including user actions such as reading, editing, saving, browsing, filtering, and/or selecting video or web page. EX1001, 6:21-36, 25:10-15; EX1003, ¶97. Accordingly, Harrop and Hailey teach this feature at least in as much detail as the '651 Patent. *Lockwood v. Am. Airlines, Inc.*, 107 F.3d 1565, 1570 (Fed. Cir. 1997) (patent owner cannot demand a greater level of detail from the prior art than the challenged patent teaches).

To the extent Patent Owner argues that the user answers and selections used in the document assembly processes of Hailey and Harrop mean that the prior art systems do not teach "automatically assembling" a document, such arguments would be inconsistent with the '651 Patent specification. Specifically, '651 Patent's document assembly system explicitly contemplates that "the value of the one or more parameters is specified by a user via a user interface" during the document assembly process. EX1001, 2:5-6. Thus, the '651 Patent does not use the term

“automatically” to exclude user action or intervention. Accordingly, a POSITA would have understood Hailey and Harrop as teaching “automatically assembling” interactive documents as claimed. EX1003, ¶98.

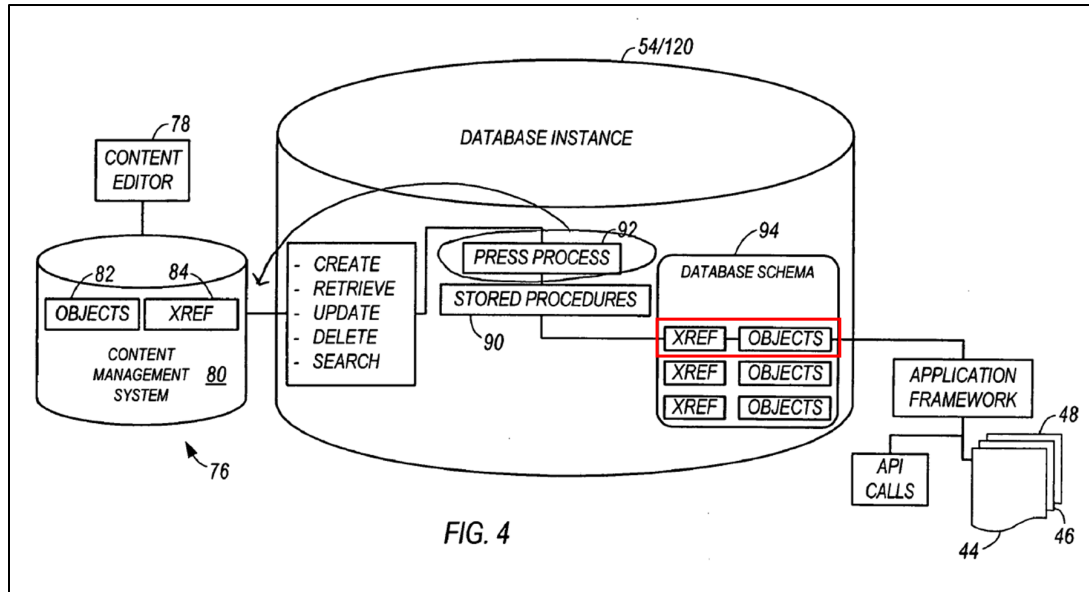
Thus, Hailey and Harrop each teach methods of creating interactive documents and the Hailey-Harrop combination renders limitation 1[PRE] obvious. EX1003, ¶99.

b. Limitation 1[A]: embedding a first set of rules into a first modular item of a first document, wherein the first document comprises a first set of modular items:

The Hailey-Harrop combination renders this limitation obvious because each of Hailey and Harrop teaches this limitation. EX1003, ¶¶100-104.

Hailey teaches providing content (modular items) stored in a knowledge base (first document) and to create output files (interactive documents) based on various input values (personalization parameters). *See* EX1005, ¶39. Hailey teaches that the items of content are stored in database 54/120 as groups of “objects” and associated embedded rules (“XREF”). EX1005, ¶64, Fig. 4. Similarly, the ’651 Patent describes that the first (or master) document may be “stored as structured data ... in the document database.” *See, e.g.*, EX1001, 1:66-67; 13:34-14:22; 15:33-46; 24:33-35. A POSITA would understand that Hailey’s knowledge base containing the content is the “first document” as claimed and that the various items of content are pulled from this database (i.e., the knowledge base). EX1003, ¶101. Hailey teaches that

these embedded rules (XREF) are created via a content editor 78. EX1005, ¶45. “[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 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.”); EX1003, ¶101.



Similarly, Harrop teaches generating an instance document (interactive document) based on logic sources (modular items) that include reusable text options which are pulled from one or more XML source documents (first 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; EX1003, ¶102. The logic sources “are themselves

structured XML documents that define questions, conditions, and reusable text objects.” EX1006, 8:28-29. A POSITA would understand that these logic sources are modular items defining rules and various text options for a derivative document. EX1003, ¶102-103.

Thus, both Hailey and Harrop teach embedding a first set of rules (XREF rules / Logic Sources) into a first modular item (Xref-content groups / Logic Sources) of a first document (database / XML source documents), wherein the first document comprises a first set of modular items. EX1003, ¶104. Accordingly, Hailey and Harrop each teach Limitation 1[A] and the Hailey-Harrop combination renders this limitation obvious. *Id.*

c. **Limitation 1[B]: receiving one or more parameters for personalizing the first document to generate a derivative document;**

The Hailey-Harrop combination renders this limitation obvious because each of Hailey and Harrop teach this limitation. EX1003, ¶105-109. As explained in §IV.A, *supra* (incorporated here), the ’651 Patent describes parameters for personalization as variables or options which a user can select to personalize a generated document. EX1001, 8:28-33. A POSITA would understand that this description aligns with the plain and ordinary meaning of “parameters for personalization.” EX1003, ¶105.

Hailey teaches that the content (*i.e.* objects and XREF) are pulled from the database (first document) based on various data (*i.e.* personalization parameters) input into the system.

[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)

EX1005, ¶40. As an 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 (“if the amount of a loan has not been specified in the input data ..., a question module 32 queries the origination platform 12 to request needed information or data 34.”), Fig. 1. Accordingly, the origination platform receives data (*i.e.*, personalization parameters) which influence, at least in part, which content is assembled into the final output file. EX1003, ¶106.

Similarly, Harrop teaches receiving values of variables (e.g., based on information specific to the parties involved and the circumstances of their relationships) to generate an instance document. EX1006, 8:1-20, 8:22-9:23;

EX1003, ¶107. In one example, Harrop teaches receiving 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. The answers provided by a user are an example of receiving a "personalization parameter for generating a derivative document." EX1003, ¶108.

Thus, as Dr. Bederson confirms, both Hailey and Harrop teach receiving personalization parameters which are used in generating a derivative document and, therefore, Hailey and Harrop each teach Limitation 1[B] and the Hailey-Harrop combination renders this limitation obvious. EX1003, ¶109.

d. **Limitation 1[C1]: selecting to display or hide at least the first modular item from the first document based at least in part on a value of the one or more parameters and the first set of rules or**

The Hailey-Harrop combination renders this limitation obvious because each of Hailey and Harrop teach this limitation. EX1003, ¶¶110-115.

Hailey teaches selecting modular items (*i.e.* objects and XREF) from the database (first document) based on the evaluation of various data input (*i.e.* personalization parameters) and displaying the selected modular item in the generated documents. EX1005, ¶40 ("The document generation module 22 validates

incoming data..., applies precedence..., *applies rules to the content...*, and *performs document assembly.*”). As discussed above, Hailey describes an example of 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, Fig. 1. Accordingly, the data (location, loan, amount) input into the origination platform are personalization parameters which influence whether certain content is assembled into the final output file. A POSITA would understand that Harrop uses the input data to evaluate whether an embedded rule is met and, depending on how the rule is met, selects which content from the modular item to include in the generated document. EX1003, ¶111.

Similarly, Harrop teaches selecting text options from logic sources in XML source documents based on received values of variables. EX1006, 8:1-20, 8:22-9:23; EX1003, ¶112. 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. A POSITA would understand that

“determin[ing] content for the resulting instance document” and “determining whether an associated portion of text is to be included” are descriptions of applying the values of variables (i.e., personalization parameters) to rules to determine which content is to be displayed (included) or hidden (excluded). Thus, whether a portion of text is to be included in the output document is based on how user’s responses (i.e., personalization parameters) meet a rule associated with the variables. EX1003, ¶112.

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?”; where 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. EX1003, ¶113.

Thus, 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 logic sources (modular item) to resolve an associated logic rule and the appropriate a portion of text is pulled from the XML source document (i.e.,

master document) and the text is selected and displayed in the generated instance document (i.e., interactive document) based on how the values (i.e. answer) meets the logic rule. EX1003, ¶114.

Thus, as Dr. Bederson confirms, both Hailey and Harrop teach selecting one or more modular items based on how one or more personalization parameters (information related to the parties / variables holding answers as values) meets an associated rule. EX1003, ¶115. Thus, Hailey and Harrop each teach Limitation 1[C1] and the Hailey–Harrop combination renders this limitation obvious. *Id.*

- e. **Limitation 1[C2.A]: pull a second modular item from a second document based at least in part on the value of the one or more parameters and a second set of rules embedded with the second modular item,**

The Hailey-Harrop combination renders this limitation obvious because Harrop teaches this limitation. EX1003, ¶¶116-117.

Harrop teaches using links to incorporate a module from a second source into a first XML document:

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.

EX1006, 55:22-24. For each module link, Harrop’s editor 112 includes, *inter alia*, the “logic sources required by the module to the target document LogicSources” and “party mapping (if possible) so that the module can be imported into the target

document.” EX1006, 56:2-3. Thus, Harrop teaches that a second set of rules (the logic sources and party mapping) are embedded in the second modular item (the linked module). A POSITA would understand Harrop as teaching that the linked modules are evaluated and included in the generated document according to a similar process as modules that are directly included in a source document—that is, input data is evaluated and a module’s content is included if the proper conditions are met. *See* EX1006, 8:1-20, 8:22-9:23; EX1003, ¶117. Accordingly, Harrop teaches pulling a second modular item (content in a referenced module) from a second document (the database or source document where the referenced module resides) depending on the value of a personalization parameter. EX1003, ¶117.

f. **Limitation 1[C2.B]: wherein the second document comprises a second set of modular items, and wherein the first document and the second document are master documents that one or more of the first set of modular items and one or more of the second set of modular items are embedded with rules; and**

The Hailey-Harrop combination renders this limitation obvious because each of Harrop teaches this limitation. EX1003, ¶¶118-120.

Harrop teaches that the modular items reside in target documents directly when a user imports the text and all associated logic of the module into a target documents or modules reside elsewhere in data structures and are incorporated into a target document via reference. EX1006, 55:6-20, 55:22-56:13; *see* §VIII.A.ii.d, *supra* (incorporated here). As discussed above, Harrop teaches source XML

documents 208 which contain logic sources (i.e., modules) having “questions, conditions, and reusable text objects.” EX1006, 8:22-30; *see* §VIII.A.ii.c, *supra* (incorporated here). Similarly, the ’651 Patent describes that the master document may be “stored as structured data ... in the document database.” *See, e.g.*, EX1001, 1:66-67, 13:34-14:22, 15:33-46, 24:33-35. A POSITA would understand that in Harrop’s system, the modules, which include reusable text and logic sources, are contained in XML source documents. EX1003, ¶119.

Further, Harrop teaches that the modules comprise: “(i) the XML element or text the user has selected; (ii) external-items (that is, items in the document but outside the text selected by the user that are referred to (generally by ID) by that text or by a reference from that reference and so on); and *(iii) a LogicSources element declaring logic required by (i) and (ii).*” EX1006, 54:23-28 (emphasis added). A POSITA would understand that the included “LogicSources” are rules which are embedded in each module. EX1003, ¶120.

g. Limitation 1[D1]: generating the derivative document by assembling the second modular item with the first document or by displaying or hiding the at least first modular item from the first document, wherein

The Hailey-Harrop combination renders this limitation obvious because each of Hailey and Harrop teach this limitation. EX1003, ¶121-124.

Hailey teaches generating a derivative document by assembling content (modular items) based on various input values (personalization parameters).

EX1005, ¶39. “The assembly module 30 retrieves data or content and rules from the knowledge base 54 and assembles the content to create documents (such as documents 44, 46, and 48).” EX1005, ¶66. A POSITA would understand that when a content is assembled in the generated document, the content is “displaying” as required by limitation 1[D1] since the generated document is ultimately displayed with included/assembled content. EX1003, ¶122. Where a second modular item is selected, as discussed in §VIII.A.ii.e, *supra* (incorporated here) regarding limitation 1[C2.A] above, the second modular item will be assembled and displayed as part of the generated document. EX1005, ¶66; EX1003, ¶122.

Similarly, Harrop teaches generating an instance document (interactive document) based on logic sources or modules (“reusable XML chunks”) 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; EX1003, ¶123. Harrop teaches the modular content included in the generated document is displayed to a user in HTML or PDF format. EX1003, ¶123. Thus, whether the first modular item or second modular item is assembled in the generated document, the included modular item will be displayed. *Id.*

Thus, Hailey and Harrop each teach Limitation 1[D1] and the Hailey-Harrop combination renders this limitation obvious. EX1003, ¶124.

h. Limitation 1[D2]: the derivative document comprises a plurality of modular items that are not embedded with rules.

The Hailey-Harrop combination renders this limitation obvious because each of Hailey and Harrop teaches this limitation. EX1003, ¶¶125-127.

Notably, the '651 Patent describes the embedding of rules in modular items in a master document, but the '651 Patent provides no description or example of an interactive document that *does not* include the assembly rule. *See* EX1001, 3:47-49; EX1003, ¶126. Notwithstanding a lack of supporting disclosure for this limitation, a POSITA would have been able to understand the plain and ordinary meaning of this limitation based on the claim language itself and would have found the claim to be obvious over the Hailey-Harrop combination. *Id.*

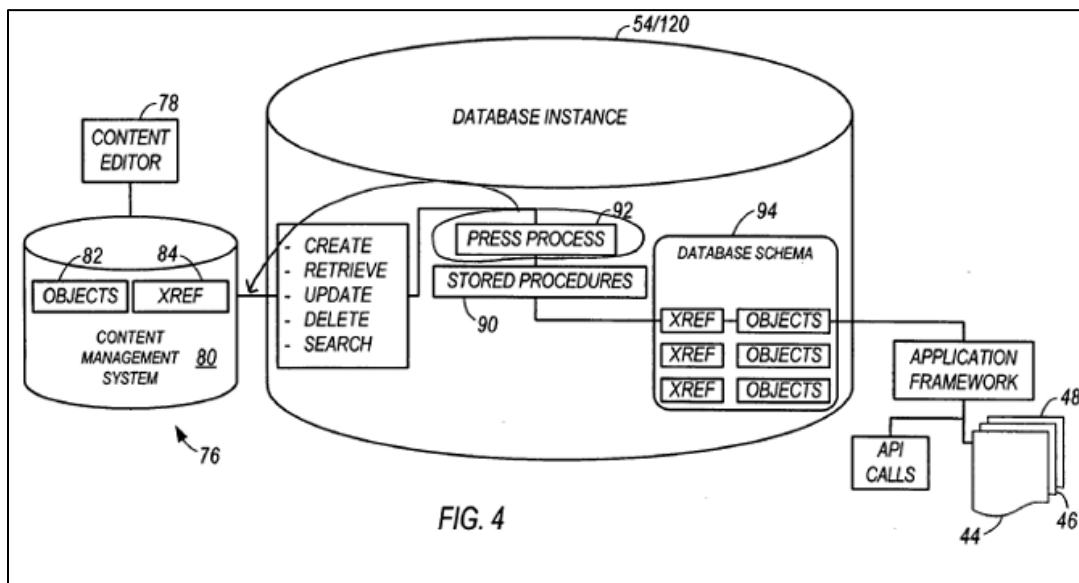
Hailey and Harrop both describe using source (master) documents and variables (personalization parameters) to select modular content and generate output (derivative) documents as HTML and PDF documents. *See* §VIII.A.ii.c, *supra* (incorporated here); EX1005, ¶¶39-41, 64, Figs. 1, 4; EX1006, 8:1-9:23, 14:7-11, 14:21-15:6, Fig. 7. The selected modular content that is included in the generated document forms the “second set of modular items” as claimed. Further, a POSITA would understand that a generated HTML or PDF document would include the

selected modular content based on personalization parameters but would not include the assembly rules used to select the content and assemble the generated document. EX1003, ¶¶126-127. As Dr. Bederson explains, it would be obvious to *exclude* the assembly rules from the modular content items after they are evaluated and only populate the generated document with the object or text content from the modular items for several reasons. *Id.* For example, a POSITA would recognize that including assembly rules in a generated document would unnecessarily complicate the document, including limiting the file types that the generated document can use. Additionally, a POSITA would understand that including the assembly rules in the generated document would introduce a security risk because an end-user could then potentially 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, EX1021; EX1003, ¶127. Moreover, a POSITA would understand 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. EX1003, ¶127. In other words, 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, generated document. *Id.*

iii. **Claim 2: The method of claim 1, wherein the first modular item is stored in a structured data that defines the first set of rules.**

In addition to all the limitations of claim 1, the Hailey-Harrop combination further renders claim 2 obvious. *See* §VIII.A.ii, *supra* (incorporated here); EX1003, ¶128-131.

Hailey teaches storing content (*e.g.*, objects) in a knowledge base (shown as a runtime database 120 in FIGS. 8 and 4) from which the content (*e.g.* objects) for a particular document can be extracted. EX1005, ¶95. Figure 4 (reproduced below) demonstrates that this data is stored in a structured manner. EX1003, ¶129.



Similarly, Harrop teaches that the source documents 208 are 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; EX1003, ¶130.

Accordingly, the Hailey-Harrop combination renders claim 2 obvious.

EX1003, ¶131.

- iv. **Claim 3: The method of claim 1, wherein the first set of rules define a link between the first modular item and another document or a rule for assembling the first modular item into a derivative document.**

In addition to all the limitations of claim 1, the Hailey-Harrop combination further renders claim 3 obvious. *See* §VIII.A.ii, *supra* (incorporated here); EX1003, ¶¶132-135.

As discussed in §VIII.A.ii.b, *supra* (incorporated here) regarding limitation 1[A], Hailey teaches that content (modular items) are stored in the database as groups of XREF (rules) and objects. *See* EX1005, ¶¶39-41, 64; EX1003, ¶133. “The assembly module 30 retrieves data or content and rules from the knowledge base 54 and assembles the content to create documents (such as documents 44, 46, and 48).” EX1005, ¶66. These XREF rules define a rule for assembling the modular items into a derivative document. EX1003, ¶133.

In addition, Harrop teaches generating an instance document (interactive document) based on logic sources or modules (“reusable XML chunks”) 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; EX1003, ¶134. The logic sources define

rules “used to determine values for variables required by a source document or other logic source referring to that logic source.” EX1006, 9:11-12. The application of variables “determin[es] whether an associated portion of text is to be included in a generated instance document.” *Id.*, 9:19-23. EX1003, ¶134.

Thus, both Hailey and Harrop teach a first set of rules which define a rule for assembling a first modular item into a derivative document. EX1003, ¶135.

v. **Claim 4: The method of claim 3, wherein the value of the one or more parameters is specified by a user via a user interface.**

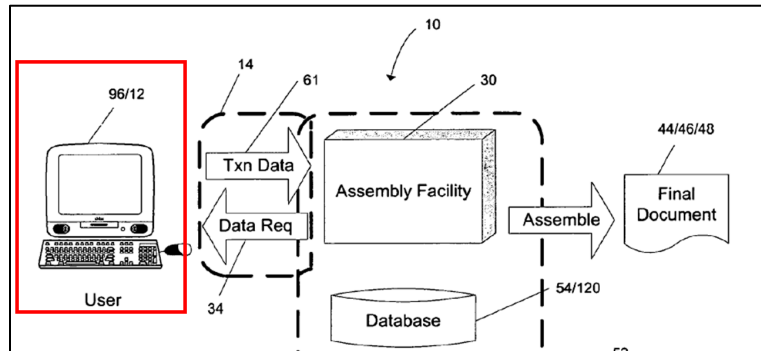
In addition to all the limitations of claims 1 and 3, the Hailey-Harrop combination further renders claim 4 obvious. §§VIII.A.ii, VIII.A.iv, *supra* (incorporated here). EX1003, ¶136-139.

Harrop teaches that a web browser displays interview questions to solicit answers from a user and generate an instance document. EX1006, 14:21-31, Fig. 6. A web browser is, or at least includes, a graphical user interface (GUI) which allows a user to input or select answers and the user’s answers to the displayed interview questions are parameters specified by a user. EX1003, ¶137.

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. EX1005, ¶¶39-41, 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”).

EX1003, ¶138.



Accordingly, Hailey teaches that a user inputs “transaction data” into the computer via an included a user interface, which necessarily exists on a computer with a monitor which receives user input. EX1003, ¶139.

vi. **Claim 5: The method of claim 1, wherein the user interface displays one or more personalization fields that are automatically generated based on the first set of rules.**

In addition to all the limitations of claim 1, the Hailey-Harrop combination further renders claim 5 obvious. *See* §VIII.A.ii, *supra* (incorporated here). EX1003, ¶140-141.

Harrop teaches displaying interview questions to a user to solicit answers to resolve logic rules. EX1006, 9:6-9 (“When a user of the system is used as a data source, the logic is resolved by performing one or more question and answer interview rounds that determine which components and text from the source documents 208, referenced logic sources 210 and user responses are included in the resulting instance document.”). The displayed interview questions are

“personalization fields” into which the user inputs or selects answers. EX1003, ¶141; *cf.* EX1001, 8:30-33 (“a user may select a parameter from a plurality of options for a personalization field to customize a document that is to be delivered to a specific reader/customer.”). Harrop further teaches generating the interview questions based on the variables that need to be resolved for the given XML source document:

Variables that can be resolved without user input are resolved by the processing engine 202 from other data sources. *The processing engine 202 provides the questions to which it requires responses to the Java controller servlet 218.* It is the responsibility of that servlet 218 to return the responses to the processing engine 202. *Where the questions are put to a user via their web browser, the controller servlet 218 can put all the questions to the user in a single HTML form, ...*

EX1006, 14:13-19. Thus, Harrop teaches displaying questions (personalization fields) to a user via a GUI (web browser) which are automatically generated based on the modules / logic sources in the XML source document.

vii. **Claim 6: The method of claim 5, wherein at least one of the one or more personalization fields is generated based on the second set of rules associated with the second modular item.**

In addition to all the limitations of claims 1 and 5, the Hailey-Harrop combination further renders claim 6 obvious. *See* §§VIII.A.ii, VIII.A.vi, *supra* (incorporated here); EX1003, ¶142-143.

As discussed in §VIII.A.ii.d, *supra* (incorporated here) regarding limitation 1[C2.A], Harrop teaches that modules (“reusable XML chunks”) may be incorporated by reference into a target document and that the referenced module

includes logic sources for resolving the required text. EX1006, 55:22-56:13. Harrop further teaches that logic sources define the question using “<Question>” tags and provides an example code:

```
<UserTextQuestion Columns="20" ID="LetterSubject" Rows="1" Type="Text">  
  <Question>What is the subject of this letter?</Question>  
  <Topic>Subject of Letter</Topic>  
</UserTextQuestion>
```

EX1006, 35:32-36:3 (annotated). A POSITA would understand that when a reference to a module is included in a target document, Harrop’s processing engine will include the question embedded in the referenced module as part of the interview questions presented to the user to resolve the logic of referenced module. EX1003, ¶143. Thus, the question (i.e., personalization field) presented to the user will be based on the second set of rules (logic sources in the referenced module) associated with the second modular item (referenced module). *Id.*

viii. Claim 8: The method of claim 6, wherein the value of the one or more parameters is specified by a user in a text file.

In addition to all the limitations of claims 1, 5 and 6, the Hailey-Harrop combination further renders claim 8 obvious. *See* §§VIII.A.ii, VIII.A.vi, VIII.A.vii, *supra* (incorporated here); EX1003, ¶144-146.

Harrop teaches that user response data may be stored in “HTML form” and re-applied to a source document to avoid unnecessary repetition of previous answers:

It is useful to be able to store the response data, so that if a user wishes to modify some of the answers they have given in an earlier session, they can start by applying the stored response data to the source document, rather than having to answer all the questions again themselves. *When an answer file containing stored response data is loaded, the questions and answers are displayed to the user in an HTML form.* After the user has modified the responses as appropriate, and submitted the form (by selecting a "next" button), those interview items are provided to the evaluator module 402.

EX1006, 16:4-10 ; *see also* EX1006, 7:22-23 (“The data files 110 may also include stored response data 215”), 9:18-19 (“ A given set of responses to a set of questions can be stored as stored response data 215 on the hard disk 104 of the document assembly system.”). A POSITA would understand that Harrop’s system allows for the user to edit answers via text input and the “HTML form” taught by Harrop is a text file that contains the user’s answers. EX1003, ¶145.

Further, to the extent that Patent Owner argues that an “HTML form” file is not a “text file,” a POSITA would have found it obvious to store user response data in a text file since text files have been known and used to electronic store data since nearly the invention of the computer. EX1003, ¶146.

ix. **Claim 9: The method of claim 1, wherein the first set of rules includes a mapping relationship that maps the first modular item to a style for rendering the first modular item.**

In addition to all the limitations of claim 1, the Hailey-Harrop combination further renders claim 9 obvious. *See* §VIII.A.ii, *supra* (incorporated here); EX1003, ¶147-148.

Hailey teaches using document-specific style attributes to specify formatting styles of the generated derivative document:

The document element includes a number of attributes 153. In the example shown, the attributes include a source attribute,... *a document-specific style attribute*.... The attributes 153 may be used for a variety of purposes including... to indicate specific margin requirements of the state or other government entity; to specify formatting styles....

EX1005, ¶79. Hailey further teaches that the rules 56 are used to select and generate a document, a question set, *or a style sheet*.” EX1005, ¶43. Hailey teaches that these “style sheets or other formatting mechanisms are applied” to the output document. EX1005, ¶42. Accordingly, Hailey teaches rules that include a mapping relationship to a style (i.e., a style sheet) that is used to render the generated document in a certain format or style. EX1003, ¶148.

x. **Claim 10: The method of claim 1, wherein the derivative document is displayed to a reader using the mapped style.**

In addition to all the limitations of claim 1, the Hailey-Harrop combination further renders claim 10 obvious. *See* §VIII.A.ii, *supra* (incorporated here); EX1003, ¶149. As discussed in §VIII.A.ix, *supra* (incorporated here) regarding claim 9 above, Hailey teaches that the XREF rules may include formatting or a style sheet which are applied to the output/generated document. As Dr. Bederson explains, style sheets are known and used to apply formatting, color, and other aesthetic choices to a generated document. EX1003, ¶149. Hailey also teaches that the output document

is displayed to the user as an HTML, pdf, or other file. EX1005, ¶42. Accordingly, the output file will be rendered using the style sheets or formatting rules defined by the XREF rules. EX1003, ¶149.

xi. Claim 13: The method of claim 1, further comprising storing the one or more parameters and a reference to the first document for updating the derivative document.

In addition to all the limitations of claim 1, the Hailey-Harrop combination further renders claim 13 obvious. *See* §VIII.A.ii, *supra* (incorporated here). The Hailey-Harrop combination teaches this limitation. EX1003, ¶¶150-156.

Claim 13 recites storing “one or more parameters and a reference to the first document for updating the derivative document.” The ’651 Patent describes that a “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, 19:33-38. The ’651 Patent does not describe any form or data structure for the “link or connection.” EX1003, ¶151.

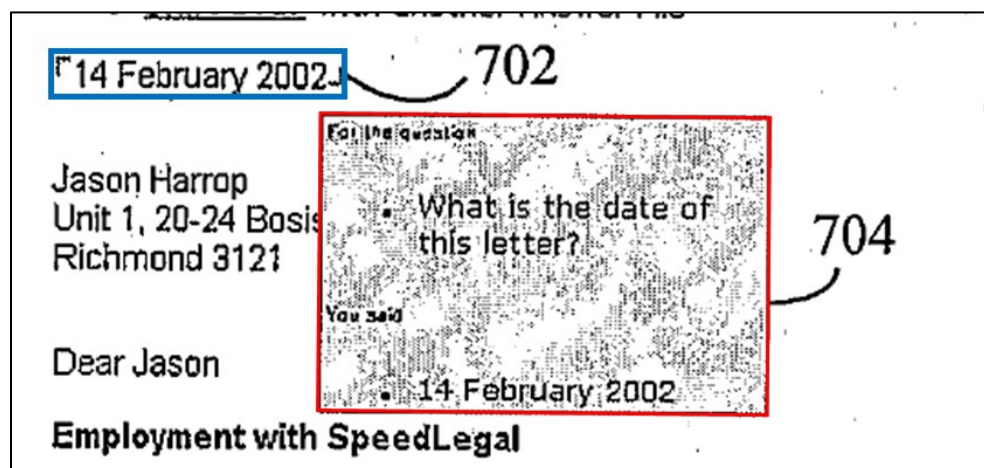
First, Harrop expressly teaches storing a user’s answers for reference in creating derivative documents based on the same XML source document or based on other source documents:

Controls 712 are also provided for storing responses on the system as stored response data 215 and for applying stored responses to the source

document they came from, or to another source document that uses some of the same logic sources, and for processing groups of related documents.

EX1006, 15:30-16:8. Thus, Harrop teaches storing the one or more parameters (user's answers). EX1003, ¶152.

Harrop teaches creating a hypertext anchor (a reference) that links the parameters (i.e., user responses) to the derivative 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 (i.e., the parameters) that resulted in that particular text being included in the rendered derivative 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). EX1003, ¶153.

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. The processing engine 202 along with java controller servlet 218 and rendering engine 204 are components of document assembly modules 108, which, in turn, exist on server 102. *See* EX1006, 6:28-29, 7:1-3, 7:8-9. Thus, invoking the processing engine means that the hypertext anchor provides a link from the user’s device (e.g., web browser) to the XML source document (i.e., the first document) located on server. EX1003, ¶154.

In addition, Harrop describes that “[i]f, at step 512, the user selects the hypertext anchor 702, 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, 15:10-13. Once a user enters new responses and any additionally required responses are also provided, “[a] new HTML rendering of the instance XML document is then generated at step 510 and displayed by the web browser” which will contain different content depending on the user’s new answers or responses. EX1006, 15:8-16. Step 510 shown in Figure 5 refers to the newly

rendered document as “generate HTML version of instance document.” EX1006, Fig. 5. Further, Harrop teaches that the referred web browser operates on the user’s computing system 114 which is separate from the server 102. EX1006, 7:11-13, Fig. 1 (showing the computing system 114 being separate from the server 102). Harrop also teaches that rendering engine 204, which exists on the server 102, is responsible for generating the HTML version of the instance document. EX1006, 10:4-5 (“Initially, the rendering engine 204 generates an HTML rendering of the instance document that includes controls for further processing of the document.”). Accordingly, a POSITA would understand that Harrop teaches updating the instance document using first a link from the derivative document to the XML source document and then using a link back to the derivative document to include updated content based on a user’s changed responses. EX1003, ¶155

Accordingly, the Hailey-Harrop combination teaches storing the one or more parameters and a reference to the first document for updating the derivative document as claimed. EX1003, ¶156.

xii. Claim 14: The method of claim 1, further comprising converting the derivative document into a master document by storing the one or more parameters as a set of rules associated with the derivative document.

In addition to all the limitations of claim 1, the Hailey-Harrop combination renders claim 14 obvious. *See* §VIII.A.ii, *supra* (incorporated here); EX1003, ¶157-159.

Claim 14 requires a derivative document to be converted to a master document by “storing the one or more *parameters* as a set of rules associated with the derivative document.” However, aside from parroting the claim language, the ’651 Patent provides no description or example where a set of parameters are stored “as a set of rules” nor does the patent provide any example where storing parameters “as a set of rules” results in the conversion of a derivative document to a master document. *See generally*, EX1001. Regardless, to the extent that the claim requires converting the derivative document by storing the personalization parameters as a set of rules, the Hailey-Harrop combination renders this claim obvious because Harrop teaches this limitation. EX1003, ¶158.

Harrop teaches that the user can select options to decide how the instance document will be further converted to an output format. Specifically, Harrop teaches that the user can select to generate an output document “in HTML (without controls), XML, or in rich text format” and that the user can select whether the output document includes “the instance document text, editorial notes... and/or the interview questions and answers on which the document was based.” EX1006, 15:18-28. Further, Harrop describes that the “controls” mentioned above “are provided for storing responses on the system as stored response data [] and for applying stored responses to the source document they came from, or to another source document that uses some of the same logic sources, and for processing groups

of related documents.” EX1006, 15:30-16:2. Harrop thus teaches converting a derivative document into an XML format that includes controls, the document content, user notes, and the interview questions and answers (parameters) associated with the derivative document. EX1003, ¶159. A POSITA would understand that Harrop’s outputting a derivative document to XML format with controls, content, notes, and questions/answers would result in a master document because the XML document with controls can be applied to another source document to create derivative documents, as taught by Harrop. *Id.*; *see also* EX1006, 15:30-16:14.

xiii. Claim 15

Claim 15 is directed to a computer-implemented method, reciting limitations that are substantially identical to the recitations in claim 1. Thus, claim 15, for at least the same reasons as claim 1, is unpatentable over Hailey and Harrop. VIII.A.ii, *supra* (incorporated here); EX1003, ¶160.

a. Limitation 15[PRE]

As discussed in §§VIII.A.ii.a, VIII.A.ii.c, *supra* (incorporated here) regarding limitations 1[pre] and 1[B], the Hailey-Harrop combination renders this limitation obvious. *See* EX1003, ¶161.

b. Limitation 15[A1]

As discussed in §§VIII.A.ii.b, VIII.A.ii.d, VIII.A.ii.e, *supra* (incorporated here) regarding limitations 1[A], 1[C1], and 1[C2.A] the Hailey-Harrop combination renders this limitation obvious. *See* EX1003, ¶162.

c. Limitation 15[B]

As discussed in §VIII.A.ii.c, *supra* (incorporated here) regarding limitations 1[B], the Hailey-Harrop combination renders this limitation obvious. *See* EX1003, ¶163.

d. Limitation 15[C1]

As discussed in §§VIII.A.ii.d, VIII.A.ii.g, *supra* (incorporated here) regarding limitations 1[C1] and 1[D1], the Hailey-Harrop combination renders this limitation obvious. *See* EX1003, ¶164.

e. Limitation 15[C2]

As discussed in §§VIII.A.ii.d-f, *supra* (incorporated here) regarding limitations 1[C1], 1[C2.A], and 1[C2.B], the Hailey-Harrop combination renders this limitation obvious. *See* EX1003, ¶165.

f. Limitation 15[C3]

As discussed in §VIII.A.ii.e-f, *supra* (incorporated here) regarding limitations 1[C2.A] and 1[C2.B], the Hailey-Harrop combination renders this limitation obvious. *See* EX1003, ¶166.

g. Limitation 15[D]

As discussed in §§VIII.A.ix-x, , *supra* (incorporated here) regarding claims 9 and 10, the Hailey-Harrop combination renders this limitation obvious. *See* EX1003, ¶167.

h. Limitation 15[E]

As discussed in §VIII.A.ii.h, *supra* (incorporated here) regarding limitations 1[D2], the Hailey-Harrop combination renders this limitation obvious. *See* EX1003, ¶168.

xiv. Claim 16: The method of claim 15, further comprising providing a first user interface for receiving a user input to generate the first set of rules.

In addition to all the limitations of claim 15, the Hailey-Harrop combination further renders claim 16 obvious. *See* §VIII.A.xiii, *supra* (incorporated here); EX1003, ¶169.

Hailey teaches that a GUI is provided to the user which allows the user to modify documents via the editing module 20. *See* EX1005, ¶39 (“a graphical user interface (“GUI”) (not shown) [may be] presented to a user such that the user may

choose to modify the list of documents in the editing module 20.”). Hailey describes the editing module as “editing module or interface 20.” Thus, Hailey teaches a user interface for receiving user input to generate the first set of rules. EX1003, ¶170.

Harrop also teaches “an editor 112 for creating and editing the data files 110.” EX1006, 7:3-4. “The editor 112 includes a user interface module 302.... the user interface 302 and XML editor 304 modules allow a user of the document assembly system to add, modify, or delete XML elements, attributes, comments, processing instructions, and text.” EX1006, 10:30-11:3. Thus, Harrop teaches a user interface for receiving a user input to generate the first set of rules. EX1003, ¶171.

xv. **Claim 17: The method of claim 16, wherein the first user interface displays a plurality of fields for defining the first set of rules.**

In addition to all the limitations of claims 15 and 16, the Hailey-Harrop combination further renders claim 17 obvious. *See* §VIII.A.xiv, *supra* (incorporated here); EX1003, ¶172.

As discussed in §VIII.A.xiv, *supra* (incorporated here), Harrop teaches an editor 112 which includes a user interface which allows a user to modify XML source documents. EX1006, 11:1-3. Further, the XML editor module 304, which is part of the editor 112, “provides standard XML editor functions *plus the ability to add and manipulate condition references* reusable text, user text, notes, and meta-information processing instructions around and within document components.”

EX1005, 11:3-6. The editing of “condition references” means that fields are present in the editor 112 which allow the user to define the “first set of rules” (the condition references in the XML elements). EX1003, ¶173.

- xvi. **Claim 18: The method of claim 15, further comprising providing a second user interface for receiving user input to customize the one or more modular items.**

In addition to all the limitations of claim 15 and as discussed in §VIII.A.v, *supra* (incorporated here) regarding claim 4, the Hailey-Harrop combination renders claim 18 obvious. *See* EX1003, ¶174.

- xvii. **Claim 19: The method of claim 15, wherein the first set of rules embedded with the at least one [modular] item of the first document includes a mapping relationship that maps the at least one item to the style.²**

In addition to all of the limitations regarding claim 15 and as discussed in §VIII.A.ix and §VIII.A.x, *supra* (incorporated here) regarding claims 9 and 10, the Hailey-Harrop combination renders this claim obvious. *See* EX1003, ¶175.

- xviii. **Claim 20: The method of claim 15, further comprising providing a user interface for receiving the instruction for personalizing the first document.**

In addition to all the limitations regarding claim 15 and as discussed in §§VIII.A.ii.c and VIII.A.xiv, *supra* (both incorporated here) regarding limitation

² The word “modular” is not present in the claim but it appears that the claim may have intended to refer to the “at least one modular item.”

1[B] and claim 16, the Hailey-Harrop combination renders this claim obvious. *See* EX1003, ¶176.

B. Ground 2: Claims 11 and 12 are rendered obvious by Hailey and Harrop in further view of Davis

i. Technology Background – User Tracking

Since (practically) the beginning of the web, user tracking has been known and implemented ubiquitously. A POSITA would have known of several commonly used ways to track how people use the web. At the simplest level, a POSITA would know of and how to use “web logs” which are logs of every URL request to a web server that are kept by essentially all web server systems. These logs can 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. For example, Google Analytics has provided analysis of web logs since at least 1995. EX1017; EX1003, ¶¶177-178.

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. EX1003, ¶179.

ii. Reasons to Combine

a. Analogous Art

A POSITA would consider Harrop, Hailey, Davis and the '651 Patent to be analogous art. As discussed in §VIII.A.i.a *supra* (incorporated here), Harrop, Hailey, and the '651 Patent are analogous art. Davis is also analogous art to Hailey, Harrop, and the '651 Patent. For example, Davis teaches generating a document (HTML web page) based on personal information (e.g., a user profile). EX1007, 14:46-52. Further, Davis is classified in International Patent Class G06F and also relates to the application of document creation systems and methods for use on the Internet, including standard hyper-text markup language (“HTML”) documents, just like the '651 Patent. *See* EX1005, 1 (field (51)), ¶42 (output file in HTML format; EX1007, 1 (field (51)), 4:1-10, 4:33-36, 5:14-18 (collecting user tracking information and using it to generate a customized HTML document). EX1003, ¶180.

b. Motivation to combine

Davis teaches embedding a tracking features in a derivative document to collect data indicative of user interaction with the document. EX1007, 5:14-18; *see* §VII.C, *supra* (incorporated here). Davis teaches embedding a tracking program in

an HTML document to gain information regarding user preferences, collect information regarding specific user interactions and create user profiles based on the tracked information. *See* EX1007, 4:3-8 (“a tracking program embedded in the resource and executable by a client.”), 4:19-23 (“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 (“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. Davis teaches using these tracked interactions to assemble a customized resource (*e.g.*, HTML webpage) based on a user’s profile which includes the tracked information. *Id.*, 4:33-36. EX1003, ¶181.

Accordingly, 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 understand 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, a POSITA would have known that user tracking information can be used to understand what works well and what does not for their users, thereby improving system designs and ensuring the system works as intended. EX1003, ¶182.

c. Reasonable Expectation of Success

The references include similar arrangements of well-understood components and use predictable technology in routine and conventional ways. EX1003, ¶183.

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 or assembling documents in an HTML format. Further, Davis teaches that his tracking feature can be "embedded in an HTML document." EX1007, 5:14-15. 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). EX1003, ¶184.

iii. **Claim 11: The method of claim 10, further comprising collecting data about the reader's interaction with the derivative document to extract insight data.**

In addition to all the limitations of claims 1 and 10, the Hailey/Harrop/Davis combination further renders claim 11 obvious. EX1003, ¶¶185-187.

Davis teaches a tracking feature that collects data about the reader's (user's) interaction with the derivative document to extract insight data. 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 documents (*e.g.*, HTML documents). EX1007, 4:33-36 (“...[A]nother 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.”). Davis further teaches that the acquired data is “used to build historical profiles of individual users, to serve out highly targeted information based upon user profiles, as well as to extract information about how much data was downloaded by a respective client, and how long or how often specific files were displayed or in use by the client.” EX1003, ¶186.

As discussed above, 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 Claim 11. EX1003, ¶187.

iv. **Claim 12: The method of claim 11, further comprising receiving an instruction for personalizing the first modular item or the second modular item.**

In addition to all the limitations of claims 1, 10, and 11, the Hailey/Harrop/Davis combination further renders claim 12 obvious. *See* §§VIII.A.ii, VIII.A.x, VIII.B.iii, *supra* (incorporated here); EX1003, ¶188. Further, as discussed in §VIII.A.xv, *supra* (incorporated here) regarding claim 17 above, Harrop teaches an editor 112 which receives instructions (input) from a user to personalize the modular items in the source documents. Therefore, the Hailey/Harrop/Davis combination renders claim 12 obvious. EX1003, ¶188.

C. **Ground 3: Claim 7 is rendered obvious by Harrop and Hailey in further view of Lo**

The features of claim 7 are taught by Lo and rendered obvious when combined with the teachings of Hailey and Harrop. EX1003, ¶189.

i. **Reasons to Combine**

a. **Analogous Art**

Lo is analogous art to Harrop, Hailey, and Davis, as well as the '651 Patent. Lo is in the same field of endeavor, namely document assembly and creation.

EX1009, ¶2 (“methods and processes for creation, editing, and printing of documents composed of text and graphics and, in printing particular, to the creation, editing and printing of variable data documents.”). As with the ’651 Patent, Lo is classified in International Patent Class G06F, Electronic Digital Data Processing. EX1009, 1 (field (52)). Accordingly, a POSITA would have considered Lo to be analogous art to Hailey, Harrop, Davis, and the ’651 Patent. EX1003, ¶190.

b. Motivation to combine

A POSITA would have been motivated to combine the teachings of Lo with the teachings of Haiely and Harrop. 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 (i.e., 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; EX1003, ¶191.

A POSITA would have been motivated to combine Lo's teaching of live-proofing previews with the teachings of Hailey and/or Harrop to enable a user to find errors in document design at an earlier stage, as taught by Lo. EX1003, ¶192.

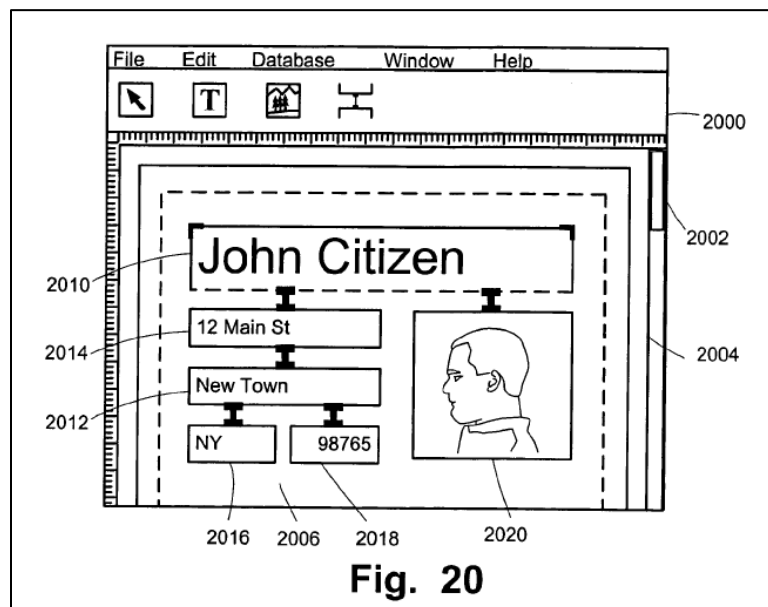
c. Reasonable Expectation of Success

A POSITA would have had a reasonable expectation of success in modifying the systems of Hailey and/or Harrop 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 known systems (*i.e.*, the interfaces taught by Hailey and Harrop). In particular, Hailey and Harrop 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* §VIII.A.v, *supra* (incorporated here). A POSITA would have found it routine to modify the already-existing user interfaces of Hailey and/or Harrop to include Lo's teachings regarding the live-proofing preview feature. EX1003, ¶193.

ii. Claim 7: The method of claim 6, further comprising: upon receiving an input in a given personalization field, displaying at least a portion of the content from the first document that is related to the given personalization field.

In addition to all the limitations of claims 1, 5, and 6, claim 7 is rendered obvious over Hailey and Harrop in further view of Lo. *See* §§VIII.A.ii, VIII.A.vi, VIII.A.vii *supra* (incorporated here); EX1003, ¶194-197.

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.



Thus, Lo teaches displaying a portion of the content from the first document that is related to a personalization field (Lo’s “variable data”) in real-time as shown in figure 20. EX1003, ¶¶195-196.

For the reasons discussed above, a POSITA would have found it obvious to include Lo’s live-proofing feature in the Hailey-Harrop combined system to provide the user with a preview of how the generated interactive document would change depending on changes to input variables. *See* §VIII.C.i, *supra* (incorporated here);

EX1003, ¶197. Accordingly, the Hailey/Harrop/Lo combination renders claim 7 obvious. *Id.*

IX. DISCRETIONARY DENIAL IS NOT WARRANTED

“Early challenges to patents favor robust, predictable patent rights and weigh against discretionary denial.” *Imperative Care, Inc. v. Inari Med., Inc.*, IPR2025-00289, Paper 9 at 2 (PTAB June 12, 2025). The ’651 Patent was issued less than five years ago, on July 5, 2022. Patent Owner sent Petitioner a notice of alleged infringement of the ’651 Patent and a demand to cease and desist the use and sale of the accused product on November 1, 2024. Petitioner promptly informed Patent Owner of Petitioner’s belief that the ’651 Patent was invalid. On October 20, 2025, Petitioner filed a Petition for *inter partes* review challenging all claims of a related patent, U.S. patent 12,118,290 which shares the same specification and is directed to the same general invention as the ’651 Patent. On January 28, 2026, in response to the petition challenging the ’290 patent, Patent Owner statutorily disclaimed all of the invention(s) claimed by the ’290 patent.

Considering Patent Owner’s notice of invalidity theories and an IPR challenge against a related patent with the same specification, Patent Owner cannot claim any settled expectations in the ’651 Patent. *See Cambridge Indus. USA Inc. v. Applied Optoelectronics, Inc.*, IPR2025-00434, Paper 11 at 2-3 (PTAB June 26, 2025) (denying requests for discretionary denial on three petitions challenging patents that

“have not been in force for a significant period of time (issued in 2020, 2019, and 2019),” finding that “Patent Owner has not developed strong settled expectations that favor discretionary denial as to at least those patents”). Petitioner, therefore, is proactively challenging the ’651 Patent to protect the market for its products and its customers from improper infringement allegations by PO that threaten Petitioner’s compelling economic and business interests.

Further, none of the references relied on in this petition were before the Office during prosecution. Had the Examiner reviewed the petition and evidence cited herein, the ’651 Patent would not have been issued. In other words, it would have been a material error by the Office to issue the ’651 Patent had the Examiner been aware of the combined teachings of Harrop and Hailey. Thus, the circumstances surrounding this Petition weigh strongly against discretionary denial.

X. CONCLUSION

Petitioner has established a reasonable likelihood that the Challenged Claims are unpatentable.

Respectfully submitted,

Dated: July 16, 2026

By: /Ryan N. Miller/
Ryan North Miller
Reg. No. 68,262

Counsel for Petitioner, Foleon Inc.

PETITIONER MANDATORY NOTICES

Real Party-in-Interest

Pursuant to 37 C.F.R. § 42.8(b)(1), Petitioner certifies that Foleon, Inc., Newion Investments Capital Early Stage Fund B.V., Newion II B.V., and CC&P IM B.V. are real parties-in interest.

Related Matters

Pursuant to 37 C.F.R. § 42.8(b)(2), to the best knowledge of the Petitioner, the '651 Patent is not involved in any pending cases.

Lead and Back-up Counsel and Service Information

Lead Counsel	Backup Counsel
Ryan N. Miller USPTO Registration 68,262 FOX ROTHSCHILD LLP 2001 Market St., Suite 1700 Philadelphia, PA 19103-3222 Tel: 215.299.2000 Fax: 215.299.2150 rmiller@foxrothschild.com	Chetan Chandra USPTO Registration 77,571 FOX ROTHSCHILD LLP 2020 K Street N.W., Suite 500 Washington, DC 20006 Tel: 202.461.3106 Fax: 202.461.3102 cchandra@foxrothschild.com
	Austin Moll USPTO Registration 80,749 FOX ROTHSCHILD LLP 777 S. Flagler Dr., Suite 1700 West Tower West Palm Beach, FL 33401 Tel: 561.804.4423 Fax: 561.835.9602 amoll@foxrothschild.com

Please address all correspondence to lead and back-up counsel. Petitioner
consents to service in this proceeding by email at

ipdocket@foxrothschild.com,
rmiller@foxrothschild.com,
cchandra@foxrothschild.com,
amoll@foxrothschild.com.

Respectfully submitted,

Dated: July 16, 2026

By: /Ryan N. Miller /
Ryan North Miller
Reg. No. 68,262

Counsel for Petitioner, Foleon, Inc.

CERTIFICATE OF WORD COUNT

Pursuant to 37 C.F.R. § 42.24(d), Petitioner hereby certifies, in accordance with and in reliance on the word count provided by the word-processing system used to prepare this Petition, that the number of words in this paper is 13,970. Pursuant to 37 C.F.R. § 42.24(a)(1), this word count excludes the table of contents, table of authorities, mandatory notices under § 42.8, certificate of service, certificate of word count, appendix of exhibits, and any claim listing.

Dated: July 16, 2026

/Ryan N. Miller/
Ryan N. Miller
Reg. No. 68,262
Counsel for Petitioner
Foleon, Inc.

CERTIFICATE OF SERVICE

Pursuant to 37 C.F.R. §§ 42.6(e) and 42.205(b), the undersigned certifies that on the 16th day of July, 2026, a complete and entire copy of this Petition for Inter Partes Review and all supporting exhibits was provided via Federal Express to:

WILSON SONSINI GOODRICH & ROSATI
650 PAGE MILL ROAD
PALO ALTO, CA 94304-1050
UNITED STATES

Courtesy Copies emailed to:

Matthew A. Argenti
USPTO Reg. No. 61,836
WILSON SONSINI
GOODRICH & ROSATI
650 Page Mill Road
Palo Alto, CA 94304
Tel.: 650-354-4154
Fax: 650-493-6811
Email: margenti@wsgr.com

Michael T. Rosato
USPTO Reg. No. 52,182
WILSON SONSINI GOODRICH
& ROSATI
701 Fifth Avenue, Suite 5100
Seattle, WA 98104-7036
Tel.: 206-883-2529
Fax: 206-883-2699
Email: mrosato@wsgr.com

Dated: July 16, 2026

/Ryan N. Miller/
Ryan N. Miller
Reg. No. 68,262
Counsel for Petitioner
Foleon, Inc.