

Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE PATENT TRIAL AND APPEAL BOARD

ACCESSIBE LTD.
Petitioner,

v.

AUDIOEYE, INC.
Patent Owner

Case No. IPR2022-01492
U.S. Patent No. 11,061,532
Issue Date: July 13, 2021

Title: MODULAR SYSTEMS AND METHODS FOR SELECTIVELY ENABLING CLOUD-
BASED ASSISTIVE TECHNOLOGIES

**DECLARATION OF JAMES FOGARTY, PH.D IN SUPPORT OF
PETITION FOR *INTER PARTES* REVIEW OF U.S. PATENT NO.
11,061,532**

TABLE OF CONTENTS

	Page
I. INTRODUCTION AND QUALIFICATIONS.....	1
A. INTRODUCTION.....	1
B. QUALIFICATIONS AND EXPERIENCE	2
C. MATERIALS CONSIDERED.....	7
II. LEGAL PRINCIPLES.....	12
A. PRIOR ART	12
B. CLAIM CONSTRUCTION	13
C. ANTICIPATION.....	15
D. OBVIOUSNESS	16
III. LEVEL OF ORDINARY SKILL IN THE ART.....	18
IV. BACKGROUND OF THE TECHNOLOGY.....	19
A. WEB PAGES AND DOCUMENT OBJECT MODELS (DOMS).....	19
B. WEB ACCESSIBILITY AND WEB ACCESSIBILITY GUIDELINES	24
C. AUTOMATED ACCESSIBILITY CHECKING AND REMEDATION TOOLS	30
V. THE '532 PATENT.....	45
A. OVERVIEW OF THE '532 PATENT	45
B. PROSECUTION HISTORY	50
C. THE CHALLENGED CLAIMS	52
VI. CLAIM CONSTRUCTION	55
VII. APPLICATION OF THE PRIOR ART TO ASSERTED CLAIMS	57
A. BRIEF SUMMARY OF PRIOR ART.....	59
1. REFERENCES CITED IN GROUNDS 1-6.....	59
A. SPRINGER [EX. 1005].....	59
B. MARKEL [EX. 1006]	72

TABLE OF CONTENTS

(continued)

	Page
C. TAKAGI [EX. 1007].....	73
D. BIGHAM [EX. 1008].....	77
E. HENDRY [EX. 1009].....	79
B. GROUND 1: CLAIMS 1-4, 6-14, 16-20 ARE UNPATENTABLE AS OBVIOUS OVER SPRINGER AND TAKAGI.....	83
1. MOTIVATION TO COMBINE	83
2. INDEPENDENT CLAIM 1	91
A. “A COMPUTER-IMPLEMENTED METHOD OF PROGRAMMATICALLY ASSIGNING A DESCRIPTIVE ATTRIBUTE TO AN UNTAGGED ELEMENT ON A WEB PAGE TO ENABLE AN AUDIBLE DESCRIPTION OF THE UNTAGGED ELEMENT, THE WEB PAGE HAVING AN ASSOCIATED DOCUMENT OBJECT MODEL (DOM), THE METHOD COMPRISING:” (CLAIM 1[PRE])	92
B. “DYNAMICALLY ANALYZING, BY A COMPUTER SYSTEM, CODE ASSOCIATED WITH THE WEB PAGE, THE CODE COMPRISING AT LEAST THE DOM OR HTML CODE;” (CLAIM 1[A])	95
C. “DETECTING, BY THE COMPUTER SYSTEM, ONE OR MORE COMPLIANCE ISSUES RELATING TO WEB ACCESSIBILITY STANDARDS IN THE CODE, WHEREIN AT LEAST ONE OF THE ONE OR MORE COMPLIANCE ISSUES COMPRISES THE UNTAGGED ELEMENT LACKING AN ADEQUATE DESCRIPTIVE ATTRIBUTE;” (CLAIM 1[B])	98

TABLE OF CONTENTS

(continued)

	Page
D. “APPLYING, BY THE COMPUTER SYSTEM, ONE OR MORE PRE-EXISTING REMEDIATION CODE TO THE ONE OR MORE COMPLIANCE ISSUES, WHEREIN THE ONE OR MORE PRE-EXISTING REMEDIATIONS IS GENERATED BY A REMOTE SERVER SYSTEM PERFORMING AT LEAST:” (CLAIM 1[C][1]).....	102
E. “ACCESSING THE CODE ASSOCIATED WITH THE WEB PAGE” (CLAIM 1[C][2]).....	112
F. “RECEIVING, BY THE REMOTE SERVER SYSTEM, VIA A REMEDIATION INTERFACE INPUT FROM A USER, THE INPUT COMPRISING A REMEDIATION ACTION TO MANUALLY REMEDIATE A NON-PROGRAMMATICALLY-FIXABLE COMPLIANCE ISSUE ASSOCIATED WITH THE WEB PAGE;” (CLAIM 1[C][3]).....	116
G. “GENERATING, BY THE REMOTE SERVER SYSTEM, THE ONE OR MORE PRE-EXISTING REMEDIATIONS BASED ON THE INPUT COMPRISING THE REMEDIATION ACTION TO MANUALLY REMEDIATE A NON-PROGRAMMATICALLY-FIXABLE COMPLIANCE ISSUE; ” (CLAIM 1[C][4])	123
H. “STORING, BY THE REMOTE SERVER SYSTEM, THE ONE OR MORE PRE-EXISTING REMEDIATIONS IN AN ELECTRONIC DATA STORAGE MEDIUM; AND” (CLAIM 1[C][5]).....	127

TABLE OF CONTENTS

(continued)

Page

I.	“ASSIGNING, BY THE COMPUTER SYSTEM, A DESCRIPTIVE ATTRIBUTE TO THE UNTAGGED ELEMENT IN THE CODE OF THE WEB PAGE BASED ON THE ONE OR MORE PRE-EXISTING REMEDIATIONS, THE DESCRIPTIVE ATTRIBUTE ASSIGNED TO THE UNTAGGED ELEMENT ADAPTED TO ENABLE AN ASSISTIVE TECHNOLOGY TO SPEAK THE DESCRIPTIVE ATTRIBUTE TO A USER.” (CLAIM 1[D])	130
3.	CLAIM 2	134
4.	CLAIM 3	135
5.	CLAIM 4	137
6.	CLAIM 6	139
7.	CLAIM 7	142
8.	CLAIM 8	143
9.	CLAIM 9	145
10.	CLAIM 10	148
11.	INDEPENDENT CLAIM 11	150
A.	“A NON-TRANSITORY ELECTRONIC STORAGE MEDIUM WITH COMPUTER CODE STORED THEREON, THE COMPUTER CODE CONFIGURED TO PROGRAMMATICALLY ASSIGN A DESCRIPTIVE ATTRIBUTE TO AN ELEMENT ON A WEB PAGE TO ENABLE AN AUDIBLE DESCRIPTION OF THE ELEMENT, THE WEB PAGE HAVING AN ASSOCIATED DOCUMENT OBJECT MODEL (DOM), AND THE COMPUTER CODE CONFIGURED TO PERFORM, WHEN EXECUTED BY A COMPUTER PROCESSOR, THE STEPS OF:” (CLAIM 11[PRE])	150

TABLE OF CONTENTS

(continued)

	Page
B. “DYNAMICALLY ANALYZING, BY A COMPUTER SYSTEM, CODE ASSOCIATED WITH THE WEB PAGE, THE CODE COMPRISING AT LEAST THE DOM OR HTML CODE” (CLAIM 11[A])	150
C. “DETECTING, BY THE COMPUTER SYSTEM, ONE OR MORE COMPLIANCE ISSUES RELATING TO WEB ACCESSIBILITY STANDARDS IN THE CODE, WHEREIN AT LEAST ONE OF THE ONE OR MORE COMPLIANCE ISSUES COMPRISES THE ELEMENT LACKING AN ADEQUATE DESCRIPTIVE ATTRIBUTE;” (CLAIM 11[B])	151
D. “APPLYING, BY THE COMPUTER SYSTEM, ONE OR MORE PRE-EXISTING REMEDIATIONS TO THE ONE OR MORE COMPLIANCE ISSUES, WHEREIN THE ONE OR MORE PRE-EXISTING REMEDIATIONS IS GENERATED BY A REMOTE SERVER SYSTEM PERFORMING AT LEAST:” (CLAIM 11[C][1])	151
E. “ACCESSING THE CODE ASSOCIATED WITH THE WEB PAGE;” (CLAIM 11[C][2])	151
F. “RECEIVING, BY THE REMOTE SERVER SYSTEM, VIA A REMEDIATION INTERFACE INPUT FROM A USER, THE INPUT COMPRISING A REMEDIATION ACTION TO MANUALLY REMEDIATE A NON-PROGRAMMATICALLY-FIXABLE COMPLIANCE ISSUE ASSOCIATED WITH THE WEB PAGE;” (CLAIM 11[C][3])	152

TABLE OF CONTENTS

(continued)

	Page
G. “GENERATING, BY THE REMOTE SERVER SYSTEM, THE ONE OR MORE PRE-EXISTING REMEDIATIONS BASED ON THE INPUT COMPRISING THE REMEDIATION ACTION TO MANUALLY REMEDIATE A NON-PROGRAMMATICALLY-FIXABLE COMPLIANCE ISSUE;” (CLAIM 11[C][4])	152
H. “STORING, BY THE REMOTE SERVER SYSTEM, THE ONE OR MORE PRE-EXISTING REMEDIATIONS IN AN ELECTRONIC DATA STORAGE MEDIUM; AND” (CLAIM 11[C][5]).....	152
I. “ASSIGNING, BY THE COMPUTER SYSTEM, A DESCRIPTIVE ATTRIBUTE TO THE ELEMENT IN THE CODE OF THE WEB PAGE BASED ON THE ONE OR MORE PRE-EXISTING REMEDIATIONS, THE DESCRIPTIVE ATTRIBUTE ASSIGNED TO THE ELEMENT ADAPTED TO ENABLE AN ASSISTIVE TECHNOLOGY TO SPEAK THE DESCRIPTIVE ATTRIBUTE TO A USER.” (CLAIM 11[D])	152
12. CLAIM 12.....	153
13. CLAIM 13.....	153
14. CLAIM 14.....	153
15. CLAIM 16.....	154
16. CLAIM 17.....	154
17. CLAIM 18.....	154
18. CLAIM 19.....	154
19. CLAIM 20.....	155
C. GROUND 2: CLAIMS 1-4, 6-14, 16-20 ARE UNPATENTABLE AS OBVIOUS OVER SPRINGER AND TAKAGI IN VIEW OF HENDRY	155

TABLE OF CONTENTS

(continued)

	Page
1. MOTIVATION TO COMBINE	155
2. INDEPENDENT CLAIM 1	160
A. “A COMPUTER-IMPLEMENTED METHOD OF PROGRAMMATICALLY ASSIGNING A DESCRIPTIVE ATTRIBUTE TO AN UNTAGGED ELEMENT ON A WEB PAGE TO ENABLE AN AUDIBLE DESCRIPTION OF THE UNTAGGED ELEMENT, THE WEB PAGE HAVING AN ASSOCIATED DOCUMENT OBJECT MODEL (DOM), THE METHOD COMPRISING:” (CLAIM 1[PRE])	160
B. “DYNAMICALLY ANALYZING, BY A COMPUTER SYSTEM, CODE ASSOCIATED WITH THE WEB PAGE, THE CODE COMPRISING AT LEAST THE DOM OR HTML CODE;” (CLAIM 1[A])	163
C. “DETECTING, BY THE COMPUTER SYSTEM, ONE OR MORE COMPLIANCE ISSUES RELATING TO WEB ACCESSIBILITY STANDARDS IN THE CODE, WHEREIN AT LEAST ONE OF THE ONE OR MORE COMPLIANCE ISSUES COMPRISES THE UNTAGGED ELEMENT LACKING AN ADEQUATE DESCRIPTIVE ATTRIBUTE;” (CLAIM 1[B])	164
D. “APPLYING, BY THE COMPUTER SYSTEM, ONE OR MORE PRE-EXISTING REMEDiation CODE TO THE ONE OR MORE COMPLIANCE ISSUES, WHEREIN THE ONE OR MORE PRE-EXISTING REMEDiations IS GENERATED BY A REMOTE SERVER SYSTEM PERFORMING AT LEAST:” (CLAIM 1[C][1])	165

TABLE OF CONTENTS

(continued)

	Page
E. “ACCESSING THE CODE ASSOCIATED WITH THE WEB PAGE” (CLAIM 1[C][2]).....	167
F. “RECEIVING, BY THE REMOTE SERVER SYSTEM, VIA A REMEDIATION INTERFACE INPUT FROM A USER, THE INPUT COMPRISING A REMEDIATION ACTION TO MANUALLY REMEDIATE A NON-PROGRAMMATICALLY-FIXABLE COMPLIANCE ISSUE ASSOCIATED WITH THE WEB PAGE;” (CLAIM 1[C][3]).....	168
G. “GENERATING, BY THE REMOTE SERVER SYSTEM, THE ONE OR MORE PRE-EXISTING REMEDIATIONS BASED ON THE INPUT COMPRISING THE REMEDIATION ACTION TO MANUALLY REMEDIATE A NON-PROGRAMMATICALLY-FIXABLE COMPLIANCE ISSUE; ” (CLAIM 1[C][4])	170
H. “STORING, BY THE REMOTE SERVER SYSTEM, THE ONE OR MORE PRE-EXISTING REMEDIATIONS IN AN ELECTRONIC DATA STORAGE MEDIUM; AND” (CLAIM 1[C][5]).....	171
I. “ASSIGNING, BY THE COMPUTER SYSTEM, A DESCRIPTIVE ATTRIBUTE TO THE UNTAGGED ELEMENT IN THE CODE OF THE WEB PAGE BASED ON THE ONE OR MORE PRE-EXISTING REMEDIATIONS, THE DESCRIPTIVE ATTRIBUTE ASSIGNED TO THE UNTAGGED ELEMENT ADAPTED TO ENABLE AN ASSISTIVE TECHNOLOGY TO SPEAK THE DESCRIPTIVE ATTRIBUTE TO A USER.” (CLAIM 1[D])	172
3. CLAIM 2	174
4. CLAIM 3	175
5. CLAIM 4	176

TABLE OF CONTENTS

(continued)

	Page
6. CLAIM 6.....	177
7. CLAIM 7.....	177
8. CLAIM 8.....	178
9. CLAIM 9.....	179
10. CLAIM 10.....	180
11. INDEPENDENT CLAIM 11.....	180
A. “A NON-TRANSITORY ELECTRONIC STORAGE MEDIUM WITH COMPUTER CODE STORED THEREON, THE COMPUTER CODE CONFIGURED TO PROGRAMMATICALLY ASSIGN A DESCRIPTIVE ATTRIBUTE TO AN ELEMENT ON A WEB PAGE TO ENABLE AN AUDIBLE DESCRIPTION OF THE ELEMENT, THE WEB PAGE HAVING AN ASSOCIATED DOCUMENT OBJECT MODEL (DOM), AND THE COMPUTER CODE CONFIGURED TO PERFORM, WHEN EXECUTED BY A COMPUTER PROCESSOR, THE STEPS OF:” (CLAIM 11[PRE]).....	180
B. “DYNAMICALLY ANALYZING, BY A COMPUTER SYSTEM, CODE ASSOCIATED WITH THE WEB PAGE, THE CODE COMPRISING AT LEAST THE DOM OR HTML CODE” (CLAIM 11[A])	181
C. “DETECTING, BY THE COMPUTER SYSTEM, ONE OR MORE COMPLIANCE ISSUES RELATING TO WEB ACCESSIBILITY STANDARDS IN THE CODE, WHEREIN AT LEAST ONE OF THE ONE OR MORE COMPLIANCE ISSUES COMPRISES THE ELEMENT LACKING AN ADEQUATE DESCRIPTIVE ATTRIBUTE;” (CLAIM 11[B])	182

TABLE OF CONTENTS

(continued)

Page

D.	“APPLYING, BY THE COMPUTER SYSTEM, ONE OR MORE PRE-EXISTING REMEDIATIONS TO THE ONE OR MORE COMPLIANCE ISSUES, WHEREIN THE ONE OR MORE PRE-EXISTING REMEDIATIONS IS GENERATED BY A REMOTE SERVER SYSTEM PERFORMING AT LEAST:” (CLAIM 11[C][1]).....	182
E.	“ACCESSING THE CODE ASSOCIATED WITH THE WEB PAGE;” (CLAIM 11[C][2]).....	182
F.	“RECEIVING, BY THE REMOTE SERVER SYSTEM, VIA A REMEDIATION INTERFACE INPUT FROM A USER, THE INPUT COMPRISING A REMEDIATION ACTION TO MANUALLY REMEDIATE A NON-PROGRAMMATICALLY-FIXABLE COMPLIANCE ISSUE ASSOCIATED WITH THE WEB PAGE;” (CLAIM 11[C][3]).....	183
G.	“GENERATING, BY THE REMOTE SERVER SYSTEM, THE ONE OR MORE PRE-EXISTING REMEDIATIONS BASED ON THE INPUT COMPRISING THE REMEDIATION ACTION TO MANUALLY REMEDIATE A NON-PROGRAMMATICALLY-FIXABLE COMPLIANCE ISSUE;” (CLAIM 11[C][4])	183
H.	“STORING, BY THE REMOTE SERVER SYSTEM, THE ONE OR MORE PRE-EXISTING REMEDIATIONS IN AN ELECTRONIC DATA STORAGE MEDIUM; AND” (CLAIM 11[C][5]).....	183

TABLE OF CONTENTS

(continued)

	Page
I. “ASSIGNING, BY THE COMPUTER SYSTEM, A DESCRIPTIVE ATTRIBUTE TO THE ELEMENT IN THE CODE OF THE WEB PAGE BASED ON THE ONE OR MORE PRE-EXISTING REMEDIATIONS, THE DESCRIPTIVE ATTRIBUTE ASSIGNED TO THE ELEMENT ADAPTED TO ENABLE AN ASSISTIVE TECHNOLOGY TO SPEAK THE DESCRIPTIVE ATTRIBUTE TO A USER.” (CLAIM 11[D])	184
12. CLAIM 12.....	184
13. CLAIM 13.....	184
14. CLAIM 14.....	185
15. CLAIM 16.....	185
16. CLAIM 17.....	185
17. CLAIM 18.....	186
18. CLAIM 19.....	186
19. CLAIM 20.....	186
D. GROUND 3: CLAIMS 1-4, 6-14, 16-20 ARE UNPATENTABLE AS OBVIOUS OVER SPRINGER AND BIGHAM IN VIEW OF HENDRY	187
1. MOTIVATION TO COMBINE	187
2. INDEPENDENT CLAIM 1	196
A. “A COMPUTER-IMPLEMENTED METHOD OF PROGRAMMATICALLY ASSIGNING A DESCRIPTIVE ATTRIBUTE TO AN UNTAGGED ELEMENT ON A WEB PAGE TO ENABLE AN AUDIBLE DESCRIPTION OF THE UNTAGGED ELEMENT, THE WEB PAGE HAVING AN ASSOCIATED DOCUMENT OBJECT MODEL (DOM), THE METHOD COMPRISING:” (CLAIM 1[PRE])	196

TABLE OF CONTENTS

(continued)

	Page
B. “DYNAMICALLY ANALYZING, BY A COMPUTER SYSTEM, CODE ASSOCIATED WITH THE WEB PAGE, THE CODE COMPRISING AT LEAST THE DOM OR HTML CODE;” (CLAIM 1[A]).....	197
C. “DETECTING, BY THE COMPUTER SYSTEM, ONE OR MORE COMPLIANCE ISSUES RELATING TO WEB ACCESSIBILITY STANDARDS IN THE CODE, WHEREIN AT LEAST ONE OF THE ONE OR MORE COMPLIANCE ISSUES COMPRISES THE UNTAGGED ELEMENT LACKING AN ADEQUATE DESCRIPTIVE ATTRIBUTE;” (CLAIM 1[B])	199
D. “APPLYING, BY THE COMPUTER SYSTEM, ONE OR MORE PRE-EXISTING REMEDIATION CODE TO THE ONE OR MORE COMPLIANCE ISSUES, WHEREIN THE ONE OR MORE PRE-EXISTING REMEDIATIONS IS GENERATED BY A REMOTE SERVER SYSTEM PERFORMING AT LEAST:” (CLAIM 1[C][1]).....	202
E. “ACCESSING THE CODE ASSOCIATED WITH THE WEB PAGE” (CLAIM 1[C][2]).....	203
F. “RECEIVING, BY THE REMOTE SERVER SYSTEM, VIA A REMEDIATION INTERFACE INPUT FROM A USER, THE INPUT COMPRISING A REMEDIATION ACTION TO MANUALLY REMEDIATE A NON-PROGRAMMATICALLY-FIXABLE COMPLIANCE ISSUE ASSOCIATED WITH THE WEB PAGE;” (CLAIM 1[C][3]).....	204

TABLE OF CONTENTS

(continued)

	Page
G. “GENERATING, BY THE REMOTE SERVER SYSTEM, THE ONE OR MORE PRE-EXISTING REMEDIATIONS BASED ON THE INPUT COMPRISING THE REMEDIATION ACTION TO MANUALLY REMEDIATE A NON-PROGRAMMATICALLY-FIXABLE COMPLIANCE ISSUE; ” (CLAIM 1[C][4])	206
H. “STORING, BY THE REMOTE SERVER SYSTEM, THE ONE OR MORE PRE-EXISTING REMEDIATIONS IN AN ELECTRONIC DATA STORAGE MEDIUM; AND” (CLAIM 1[C][5]).....	207
I. “ASSIGNING, BY THE COMPUTER SYSTEM, A DESCRIPTIVE ATTRIBUTE TO THE UNTAGGED ELEMENT IN THE CODE OF THE WEB PAGE BASED ON THE ONE OR MORE PRE-EXISTING REMEDIATIONS, THE DESCRIPTIVE ATTRIBUTE ASSIGNED TO THE UNTAGGED ELEMENT ADAPTED TO ENABLE AN ASSISTIVE TECHNOLOGY TO SPEAK THE DESCRIPTIVE ATTRIBUTE TO A USER.” (CLAIM 1[D])	208
3. CLAIM 2	210
4. CLAIM 3	210
5. CLAIM 4	213
6. CLAIM 6	213
7. CLAIM 7	216
8. CLAIM 8	217
9. CLAIM 9	219
10. CLAIM 10	220
11. INDEPENDENT CLAIM 11	220

TABLE OF CONTENTS

(continued)

Page

A.	“A NON-TRANSITORY ELECTRONIC STORAGE MEDIUM WITH COMPUTER CODE STORED THEREON, THE COMPUTER CODE CONFIGURED TO PROGRAMMATICALLY ASSIGN A DESCRIPTIVE ATTRIBUTE TO AN ELEMENT ON A WEB PAGE TO ENABLE AN AUDIBLE DESCRIPTION OF THE ELEMENT, THE WEB PAGE HAVING AN ASSOCIATED DOCUMENT OBJECT MODEL (DOM), AND THE COMPUTER CODE CONFIGURED TO PERFORM, WHEN EXECUTED BY A COMPUTER PROCESSOR, THE STEPS OF:” (CLAIM 11[PRE]).....	220
B.	“DYNAMICALLY ANALYZING, BY A COMPUTER SYSTEM, CODE ASSOCIATED WITH THE WEB PAGE, THE CODE COMPRISING AT LEAST THE DOM OR HTML CODE” (CLAIM 11[A])	221
C.	“DETECTING, BY THE COMPUTER SYSTEM, ONE OR MORE COMPLIANCE ISSUES RELATING TO WEB ACCESSIBILITY STANDARDS IN THE CODE, WHEREIN AT LEAST ONE OF THE ONE OR MORE COMPLIANCE ISSUES COMPRISES THE ELEMENT LACKING AN ADEQUATE DESCRIPTIVE ATTRIBUTE;” (CLAIM 11[B])	222
D.	“APPLYING, BY THE COMPUTER SYSTEM, ONE OR MORE PRE-EXISTING REMEDIATIONS TO THE ONE OR MORE COMPLIANCE ISSUES, WHEREIN THE ONE OR MORE PRE-EXISTING REMEDIATIONS IS GENERATED BY A REMOTE SERVER SYSTEM PERFORMING AT LEAST:” (CLAIM 11[C][1])	222

TABLE OF CONTENTS

(continued)

	Page
E. “ACCESSING THE CODE ASSOCIATED WITH THE WEB PAGE;” (CLAIM 11[C][2]).....	222
F. “RECEIVING, BY THE REMOTE SERVER SYSTEM, VIA A REMEDIATION INTERFACE INPUT FROM A USER, THE INPUT COMPRISING A REMEDIATION ACTION TO MANUALLY REMEDIATE A NON-PROGRAMMATICALLY-FIXABLE COMPLIANCE ISSUE ASSOCIATED WITH THE WEB PAGE;” (CLAIM 11[C][3]).....	223
G. “GENERATING, BY THE REMOTE SERVER SYSTEM, THE ONE OR MORE PRE-EXISTING REMEDIATIONS BASED ON THE INPUT COMPRISING THE REMEDIATION ACTION TO MANUALLY REMEDIATE A NON-PROGRAMMATICALLY-FIXABLE COMPLIANCE ISSUE;” (CLAIM 11[C][4])	223
H. “STORING, BY THE REMOTE SERVER SYSTEM, THE ONE OR MORE PRE-EXISTING REMEDIATIONS IN AN ELECTRONIC DATA STORAGE MEDIUM; AND” (CLAIM 11[G])	223
I. “ASSIGNING, BY THE COMPUTER SYSTEM, A DESCRIPTIVE ATTRIBUTE TO THE ELEMENT IN THE CODE OF THE WEB PAGE BASED ON THE ONE OR MORE PRE-EXISTING REMEDIATIONS, THE DESCRIPTIVE ATTRIBUTE ASSIGNED TO THE ELEMENT ADAPTED TO ENABLE AN ASSISTIVE TECHNOLOGY TO SPEAK THE DESCRIPTIVE ATTRIBUTE TO A USER.” (CLAIM 11[D])	224
12. CLAIM 12.....	224
13. CLAIM 13.....	224
14. CLAIM 14.....	225

TABLE OF CONTENTS

(continued)

	Page
15. CLAIM 16.....	225
16. CLAIM 17.....	225
17. CLAIM 18.....	226
18. CLAIM 19.....	226
19. CLAIM 20.....	226
E. GROUND 4: CLAIMS 1-4, 6-14, 16-20 ARE UNPATENTABLE AS OBVIOUS OVER MARKEL AND TAKAGI.....	227
1. MOTIVATION TO COMBINE	227
2. INDEPENDENT CLAIM 1	234
A. “A COMPUTER-IMPLEMENTED METHOD OF PROGRAMMATICALLY ASSIGNING A DESCRIPTIVE ATTRIBUTE TO AN UNTAGGED ELEMENT ON A WEB PAGE TO ENABLE AN AUDIBLE DESCRIPTION OF THE UNTAGGED ELEMENT, THE WEB PAGE HAVING AN ASSOCIATED DOCUMENT OBJECT MODEL (DOM), THE METHOD COMPRISING:” (CLAIM 1[PRE])	234
B. “DYNAMICALLY ANALYZING, BY A COMPUTER SYSTEM, CODE ASSOCIATED WITH THE WEB PAGE, THE CODE COMPRISING AT LEAST THE DOM OR HTML CODE;” (CLAIM 1[A])	237
C. “DETECTING, BY THE COMPUTER SYSTEM, ONE OR MORE COMPLIANCE ISSUES RELATING TO WEB ACCESSIBILITY STANDARDS IN THE CODE, WHEREIN AT LEAST ONE OF THE ONE OR MORE COMPLIANCE ISSUES COMPRISES THE UNTAGGED ELEMENT LACKING AN ADEQUATE DESCRIPTIVE ATTRIBUTE;” (CLAIM 1[B])	240

TABLE OF CONTENTS

(continued)

	Page
D. “APPLYING, BY THE COMPUTER SYSTEM, ONE OR MORE PRE-EXISTING REMEDIATION CODE TO THE ONE OR MORE COMPLIANCE ISSUES, WHEREIN THE ONE OR MORE PRE-EXISTING REMEDIATIONS IS GENERATED BY A REMOTE SERVER SYSTEM PERFORMING AT LEAST:” (CLAIM 1[C][1]).....	242
E. “ACCESSING THE CODE ASSOCIATED WITH THE WEB PAGE” (CLAIM 1[C][2]).....	244
F. “RECEIVING, BY THE REMOTE SERVER SYSTEM, VIA A REMEDIATION INTERFACE INPUT FROM A USER, THE INPUT COMPRISING A REMEDIATION ACTION TO MANUALLY REMEDIATE A NON-PROGRAMMATICALLY-FIXABLE COMPLIANCE ISSUE ASSOCIATED WITH THE WEB PAGE;” (CLAIM 1[C][3]).....	246
G. “GENERATING, BY THE REMOTE SERVER SYSTEM, THE ONE OR MORE PRE-EXISTING REMEDIATIONS BASED ON THE INPUT COMPRISING THE REMEDIATION ACTION TO MANUALLY REMEDIATE A NON-PROGRAMMATICALLY-FIXABLE COMPLIANCE ISSUE; ” (CLAIM 1[C][4])	248
H. “STORING, BY THE REMOTE SERVER SYSTEM, THE ONE OR MORE PRE-EXISTING REMEDIATIONS IN AN ELECTRONIC DATA STORAGE MEDIUM; AND” (CLAIM 1[C][5]).....	250

TABLE OF CONTENTS

(continued)

Page

I.	“ASSIGNING, BY THE COMPUTER SYSTEM, A DESCRIPTIVE ATTRIBUTE TO THE UNTAGGED ELEMENT IN THE CODE OF THE WEB PAGE BASED ON THE ONE OR MORE PRE-EXISTING REMEDIATIONS, THE DESCRIPTIVE ATTRIBUTE ASSIGNED TO THE UNTAGGED ELEMENT ADAPTED TO ENABLE AN ASSISTIVE TECHNOLOGY TO SPEAK THE DESCRIPTIVE ATTRIBUTE TO A USER.” (CLAIM 1[D])	251
3.	CLAIM 2	254
4.	CLAIM 3	254
5.	CLAIM 4	255
6.	CLAIM 6	256
7.	CLAIM 7	257
8.	CLAIM 8	259
9.	CLAIM 9	259
10.	CLAIM 10	260
11.	INDEPENDENT CLAIM 11	261
A.	“A NON-TRANSITORY ELECTRONIC STORAGE MEDIUM WITH COMPUTER CODE STORED THEREON, THE COMPUTER CODE CONFIGURED TO PROGRAMMATICALLY ASSIGN A DESCRIPTIVE ATTRIBUTE TO AN ELEMENT ON A WEB PAGE TO ENABLE AN AUDIBLE DESCRIPTION OF THE ELEMENT, THE WEB PAGE HAVING AN ASSOCIATED DOCUMENT OBJECT MODEL (DOM), AND THE COMPUTER CODE CONFIGURED TO PERFORM, WHEN EXECUTED BY A COMPUTER PROCESSOR, THE STEPS OF:” (CLAIM 11[PRE])	262

TABLE OF CONTENTS

(continued)

	Page
B. “DYNAMICALLY ANALYZING, BY A COMPUTER SYSTEM, CODE ASSOCIATED WITH THE WEB PAGE, THE CODE COMPRISING AT LEAST THE DOM OR HTML CODE” (CLAIM 11[A])	262
C. “DETECTING, BY THE COMPUTER SYSTEM, ONE OR MORE COMPLIANCE ISSUES RELATING TO WEB ACCESSIBILITY STANDARDS IN THE CODE, WHEREIN AT LEAST ONE OF THE ONE OR MORE COMPLIANCE ISSUES COMPRISES THE ELEMENT LACKING AN ADEQUATE DESCRIPTIVE ATTRIBUTE;” (CLAIM 11[B])	263
D. “APPLYING, BY THE COMPUTER SYSTEM, ONE OR MORE PRE-EXISTING REMEDIATIONS TO THE ONE OR MORE COMPLIANCE ISSUES, WHEREIN THE ONE OR MORE PRE-EXISTING REMEDIATIONS IS GENERATED BY A REMOTE SERVER SYSTEM PERFORMING AT LEAST:” (CLAIM 11[C][1])	263
E. “ACCESSING THE CODE ASSOCIATED WITH THE WEB PAGE;” (CLAIM 11[C][2])	263
F. “RECEIVING, BY THE REMOTE SERVER SYSTEM, VIA A REMEDIATION INTERFACE INPUT FROM A USER, THE INPUT COMPRISING A REMEDIATION ACTION TO MANUALLY REMEDIATE A NON-PROGRAMMATICALLY-FIXABLE COMPLIANCE ISSUE ASSOCIATED WITH THE WEB PAGE;” (CLAIM 11[C][3])	264

TABLE OF CONTENTS

(continued)

	Page
G. “GENERATING, BY THE REMOTE SERVER SYSTEM, THE ONE OR MORE PRE-EXISTING REMEDIATIONS BASED ON THE INPUT COMPRISING THE REMEDIATION ACTION TO MANUALLY REMEDIATE A NON-PROGRAMMATICALLY-FIXABLE COMPLIANCE ISSUE;” (CLAIM 11[C][4])	264
H. “STORING, BY THE REMOTE SERVER SYSTEM, THE ONE OR MORE PRE-EXISTING REMEDIATIONS IN AN ELECTRONIC DATA STORAGE MEDIUM; AND” (CLAIM 11[C][5]).....	264
I. “ASSIGNING, BY THE COMPUTER SYSTEM, A DESCRIPTIVE ATTRIBUTE TO THE ELEMENT IN THE CODE OF THE WEB PAGE BASED ON THE ONE OR MORE PRE-EXISTING REMEDIATIONS, THE DESCRIPTIVE ATTRIBUTE ASSIGNED TO THE ELEMENT ADAPTED TO ENABLE AN ASSISTIVE TECHNOLOGY TO SPEAK THE DESCRIPTIVE ATTRIBUTE TO A USER.” (CLAIM 11[D])	264
12. CLAIM 12.....	265
13. CLAIM 13.....	265
14. CLAIM 14.....	265
15. CLAIM 16.....	266
16. CLAIM 17.....	266
17. CLAIM 18.....	266
18. CLAIM 19.....	266
19. CLAIM 20.....	267
F. GROUND 5: CLAIMS 1-4, 6-14, 16-20 ARE UNPATENTABLE AS OBVIOUS OVER MARKEL AND TAKAGI IN VIEW OF HENDRY	267

-xx-

TABLE OF CONTENTS

(continued)

	Page
1. MOTIVATION TO COMBINE	267
2. INDEPENDENT CLAIM 1	273
A. “A COMPUTER-IMPLEMENTED METHOD OF PROGRAMMATICALLY ASSIGNING A DESCRIPTIVE ATTRIBUTE TO AN UNTAGGED ELEMENT ON A WEB PAGE TO ENABLE AN AUDIBLE DESCRIPTION OF THE UNTAGGED ELEMENT, THE WEB PAGE HAVING AN ASSOCIATED DOCUMENT OBJECT MODEL (DOM), THE METHOD COMPRISING:” (CLAIM 1[PRE])	273
B. “DYNAMICALLY ANALYZING, BY A COMPUTER SYSTEM, CODE ASSOCIATED WITH THE WEB PAGE, THE CODE COMPRISING AT LEAST THE DOM OR HTML CODE;” (CLAIM 1[A])	273
C. “DETECTING, BY THE COMPUTER SYSTEM, ONE OR MORE COMPLIANCE ISSUES RELATING TO WEB ACCESSIBILITY STANDARDS IN THE CODE, WHEREIN AT LEAST ONE OF THE ONE OR MORE COMPLIANCE ISSUES COMPRISES THE UNTAGGED ELEMENT LACKING AN ADEQUATE DESCRIPTIVE ATTRIBUTE;” (CLAIM 1[B])	273
D. “APPLYING, BY THE COMPUTER SYSTEM, ONE OR MORE PRE-EXISTING REMEDiation CODE TO THE ONE OR MORE COMPLIANCE ISSUES, WHEREIN THE ONE OR MORE PRE-EXISTING REMEDiations IS GENERATED BY A REMOTE SERVER SYSTEM PERFORMING AT LEAST:” (CLAIM 1[C][1])	274

TABLE OF CONTENTS

(continued)

	Page
E. “ACCESSING THE CODE ASSOCIATED WITH THE WEB PAGE” (CLAIM 1[C][2]).....	274
F. “RECEIVING, BY THE REMOTE SERVER SYSTEM, VIA A REMEDIATION INTERFACE INPUT FROM A USER, THE INPUT COMPRISING A REMEDIATION ACTION TO MANUALLY REMEDIATE A NON-PROGRAMMATICALLY-FIXABLE COMPLIANCE ISSUE ASSOCIATED WITH THE WEB PAGE;” (CLAIM 1[C][3]).....	274
G. “GENERATING, BY THE REMOTE SERVER SYSTEM, THE ONE OR MORE PRE-EXISTING REMEDIATIONS BASED ON THE INPUT COMPRISING THE REMEDIATION ACTION TO MANUALLY REMEDIATE A NON-PROGRAMMATICALLY-FIXABLE COMPLIANCE ISSUE; ” (CLAIM 1[C][4])	275
H. “STORING, BY THE REMOTE SERVER SYSTEM, THE ONE OR MORE PRE-EXISTING REMEDIATIONS IN AN ELECTRONIC DATA STORAGE MEDIUM; AND” (CLAIM 1[C][5]).....	275
I. “ASSIGNING, BY THE COMPUTER SYSTEM, A DESCRIPTIVE ATTRIBUTE TO THE UNTAGGED ELEMENT IN THE CODE OF THE WEB PAGE BASED ON THE ONE OR MORE PRE-EXISTING REMEDIATIONS, THE DESCRIPTIVE ATTRIBUTE ASSIGNED TO THE UNTAGGED ELEMENT ADAPTED TO ENABLE AN ASSISTIVE TECHNOLOGY TO SPEAK THE DESCRIPTIVE ATTRIBUTE TO A USER.” (CLAIM 1[D])	275
3. CLAIM 2	276
4. CLAIM 3	276
5. CLAIM 4	276

TABLE OF CONTENTS

(continued)

	Page
6. CLAIM 6.....	277
7. CLAIM 7.....	277
8. CLAIM 8.....	277
9. CLAIM 9.....	278
10. CLAIM 10.....	278
11. INDEPENDENT CLAIM 11	278
A. “A NON-TRANSITORY ELECTRONIC STORAGE MEDIUM WITH COMPUTER CODE STORED THEREON, THE COMPUTER CODE CONFIGURED TO PROGRAMMATICALLY ASSIGN A DESCRIPTIVE ATTRIBUTE TO AN ELEMENT ON A WEB PAGE TO ENABLE AN AUDIBLE DESCRIPTION OF THE ELEMENT, THE WEB PAGE HAVING AN ASSOCIATED DOCUMENT OBJECT MODEL (DOM), AND THE COMPUTER CODE CONFIGURED TO PERFORM, WHEN EXECUTED BY A COMPUTER PROCESSOR, THE STEPS OF:” (CLAIM 11[PRE]).....	279
B. “DYNAMICALLY ANALYZING, BY A COMPUTER SYSTEM, CODE ASSOCIATED WITH THE WEB PAGE, THE CODE COMPRISING AT LEAST THE DOM OR HTML CODE” (CLAIM 11[A])	279
C. “DETECTING, BY THE COMPUTER SYSTEM, ONE OR MORE COMPLIANCE ISSUES RELATING TO WEB ACCESSIBILITY STANDARDS IN THE CODE, WHEREIN AT LEAST ONE OF THE ONE OR MORE COMPLIANCE ISSUES COMPRISES THE ELEMENT LACKING AN ADEQUATE DESCRIPTIVE ATTRIBUTE;” (CLAIM 11[B])	280

TABLE OF CONTENTS

(continued)

	Page
D. “APPLYING, BY THE COMPUTER SYSTEM, ONE OR MORE PRE-EXISTING REMEDIATIONS TO THE ONE OR MORE COMPLIANCE ISSUES, WHEREIN THE ONE OR MORE PRE-EXISTING REMEDIATIONS IS GENERATED BY A REMOTE SERVER SYSTEM PERFORMING AT LEAST:” (CLAIM 11[C][1]).....	280
E. “ACCESSING THE CODE ASSOCIATED WITH THE WEB PAGE;” (CLAIM 11[C][2]).....	281
F. “RECEIVING, BY THE REMOTE SERVER SYSTEM, VIA A REMEDIATION INTERFACE INPUT FROM A USER, THE INPUT COMPRISING A REMEDIATION ACTION TO MANUALLY REMEDIATE A NON-PROGRAMMATICALLY-FIXABLE COMPLIANCE ISSUE ASSOCIATED WITH THE WEB PAGE;” (CLAIM 11[C][3]).....	281
G. “GENERATING, BY THE REMOTE SERVER SYSTEM, THE ONE OR MORE PRE-EXISTING REMEDIATIONS BASED ON THE INPUT COMPRISING THE REMEDIATION ACTION TO MANUALLY REMEDIATE A NON-PROGRAMMATICALLY-FIXABLE COMPLIANCE ISSUE;” (CLAIM 11[C][4])	281
H. “STORING, BY THE REMOTE SERVER SYSTEM, THE ONE OR MORE PRE-EXISTING REMEDIATIONS IN AN ELECTRONIC DATA STORAGE MEDIUM; AND” (CLAIM 11[C][5]).....	282

TABLE OF CONTENTS

(continued)

	Page
I. “ASSIGNING, BY THE COMPUTER SYSTEM, A DESCRIPTIVE ATTRIBUTE TO THE ELEMENT IN THE CODE OF THE WEB PAGE BASED ON THE ONE OR MORE PRE-EXISTING REMEDIATIONS, THE DESCRIPTIVE ATTRIBUTE ASSIGNED TO THE ELEMENT ADAPTED TO ENABLE AN ASSISTIVE TECHNOLOGY TO SPEAK THE DESCRIPTIVE ATTRIBUTE TO A USER.” (CLAIM 11[D])	282
12. CLAIM 12.....	282
13. CLAIM 13.....	283
14. CLAIM 14.....	283
15. CLAIM 16.....	283
16. CLAIM 17.....	283
17. CLAIM 18.....	284
18. CLAIM 19.....	284
19. CLAIM 20.....	284
G. GROUND 6: CLAIMS 1-4, 6-14, 16-20 ARE UNPATENTABLE AS OBVIOUS OVER MARKEL AND BIGHAM IN VIEW OF HENDRY	285
1. MOTIVATION TO COMBINE	285
2. INDEPENDENT CLAIM 1	293
A. “A COMPUTER-IMPLEMENTED METHOD OF PROGRAMMATICALLY ASSIGNING A DESCRIPTIVE ATTRIBUTE TO AN UNTAGGED ELEMENT ON A WEB PAGE TO ENABLE AN AUDIBLE DESCRIPTION OF THE UNTAGGED ELEMENT, THE WEB PAGE HAVING AN ASSOCIATED DOCUMENT OBJECT MODEL (DOM), THE METHOD COMPRISING:” (CLAIM 1[PRE])	293

TABLE OF CONTENTS

(continued)

	Page
B. “DYNAMICALLY ANALYZING, BY A COMPUTER SYSTEM, CODE ASSOCIATED WITH THE WEB PAGE, THE CODE COMPRISING AT LEAST THE DOM OR HTML CODE;” (CLAIM 1[A]).....	294
C. “DETECTING, BY THE COMPUTER SYSTEM, ONE OR MORE COMPLIANCE ISSUES RELATING TO WEB ACCESSIBILITY STANDARDS IN THE CODE, WHEREIN AT LEAST ONE OF THE ONE OR MORE COMPLIANCE ISSUES COMPRISES THE UNTAGGED ELEMENT LACKING AN ADEQUATE DESCRIPTIVE ATTRIBUTE;” (CLAIM 1[B])	294
D. “APPLYING, BY THE COMPUTER SYSTEM, ONE OR MORE PRE-EXISTING REMEDIATION CODE TO THE ONE OR MORE COMPLIANCE ISSUES, WHEREIN THE ONE OR MORE PRE-EXISTING REMEDIATIONS IS GENERATED BY A REMOTE SERVER SYSTEM PERFORMING AT LEAST:” (CLAIM 1[C][1]).....	295
E. “ACCESSING THE CODE ASSOCIATED WITH THE WEB PAGE” (CLAIM 1[C][2]).....	295
F. “RECEIVING, BY THE REMOTE SERVER SYSTEM, VIA A REMEDIATION INTERFACE INPUT FROM A USER, THE INPUT COMPRISING A REMEDIATION ACTION TO MANUALLY REMEDIATE A NON-PROGRAMMATICALLY-FIXABLE COMPLIANCE ISSUE ASSOCIATED WITH THE WEB PAGE;” (CLAIM 1[C][3]).....	295

TABLE OF CONTENTS

(continued)

	Page
G. “GENERATING, BY THE REMOTE SERVER SYSTEM, THE ONE OR MORE PRE-EXISTING REMEDIATIONS BASED ON THE INPUT COMPRISING THE REMEDIATION ACTION TO MANUALLY REMEDIATE A NON-PROGRAMMATICALLY-FIXABLE COMPLIANCE ISSUE; ” (CLAIM 1[C][4])	296
H. “STORING, BY THE REMOTE SERVER SYSTEM, THE ONE OR MORE PRE-EXISTING REMEDIATIONS IN AN ELECTRONIC DATA STORAGE MEDIUM; AND” (CLAIM 1[C][5]).....	296
I. “ASSIGNING, BY THE COMPUTER SYSTEM, A DESCRIPTIVE ATTRIBUTE TO THE UNTAGGED ELEMENT IN THE CODE OF THE WEB PAGE BASED ON THE ONE OR MORE PRE-EXISTING REMEDIATIONS, THE DESCRIPTIVE ATTRIBUTE ASSIGNED TO THE UNTAGGED ELEMENT ADAPTED TO ENABLE AN ASSISTIVE TECHNOLOGY TO SPEAK THE DESCRIPTIVE ATTRIBUTE TO A USER.” (CLAIM 1[D])	297
3. CLAIM 2	297
4. CLAIM 3	298
5. CLAIM 4	300
6. CLAIM 6	301
7. CLAIM 7	301
8. CLAIM 8	302
9. CLAIM 9	302
10. CLAIM 10	303
11. INDEPENDENT CLAIM 11	303

TABLE OF CONTENTS

(continued)

	Page
A. “A NON-TRANSITORY ELECTRONIC STORAGE MEDIUM WITH COMPUTER CODE STORED THEREON, THE COMPUTER CODE CONFIGURED TO PROGRAMMATICALLY ASSIGN A DESCRIPTIVE ATTRIBUTE TO AN ELEMENT ON A WEB PAGE TO ENABLE AN AUDIBLE DESCRIPTION OF THE ELEMENT, THE WEB PAGE HAVING AN ASSOCIATED DOCUMENT OBJECT MODEL (DOM), AND THE COMPUTER CODE CONFIGURED TO PERFORM, WHEN EXECUTED BY A COMPUTER PROCESSOR, THE STEPS OF:” (CLAIM 11[PRE]).....	303
B. “DYNAMICALLY ANALYZING, BY A COMPUTER SYSTEM, CODE ASSOCIATED WITH THE WEB PAGE, THE CODE COMPRISING AT LEAST THE DOM OR HTML CODE” (CLAIM 11[A])	304
C. “DETECTING, BY THE COMPUTER SYSTEM, ONE OR MORE COMPLIANCE ISSUES RELATING TO WEB ACCESSIBILITY STANDARDS IN THE CODE, WHEREIN AT LEAST ONE OF THE ONE OR MORE COMPLIANCE ISSUES COMPRISES THE ELEMENT LACKING AN ADEQUATE DESCRIPTIVE ATTRIBUTE;” (CLAIM 11[B])	304
D. “APPLYING, BY THE COMPUTER SYSTEM, ONE OR MORE PRE-EXISTING REMEDIATIONS TO THE ONE OR MORE COMPLIANCE ISSUES, WHEREIN THE ONE OR MORE PRE-EXISTING REMEDIATIONS IS GENERATED BY A REMOTE SERVER SYSTEM PERFORMING AT LEAST:” (CLAIM 11[C][1])	305

-xxviii

-

TABLE OF CONTENTS

(continued)

	Page
E. “ACCESSING THE CODE ASSOCIATED WITH THE WEB PAGE;” (CLAIM 11[C][2]).....	305
F. “RECEIVING, BY THE REMOTE SERVER SYSTEM, VIA A REMEDIATION INTERFACE INPUT FROM A USER, THE INPUT COMPRISING A REMEDIATION ACTION TO MANUALLY REMEDIATE A NON-PROGRAMMATICALLY-FIXABLE COMPLIANCE ISSUE ASSOCIATED WITH THE WEB PAGE;” (CLAIM 11[C][3]).....	305
G. “GENERATING, BY THE REMOTE SERVER SYSTEM, THE ONE OR MORE PRE-EXISTING REMEDIATIONS BASED ON THE INPUT COMPRISING THE REMEDIATION ACTION TO MANUALLY REMEDIATE A NON-PROGRAMMATICALLY-FIXABLE COMPLIANCE ISSUE;” (CLAIM 11[C][4])	306
H. “STORING, BY THE REMOTE SERVER SYSTEM, THE ONE OR MORE PRE-EXISTING REMEDIATIONS IN AN ELECTRONIC DATA STORAGE MEDIUM; AND” (CLAIM 11[C][5]).....	306
I. “ASSIGNING, BY THE COMPUTER SYSTEM, A DESCRIPTIVE ATTRIBUTE TO THE ELEMENT IN THE CODE OF THE WEB PAGE BASED ON THE ONE OR MORE PRE-EXISTING REMEDIATIONS, THE DESCRIPTIVE ATTRIBUTE ASSIGNED TO THE ELEMENT ADAPTED TO ENABLE AN ASSISTIVE TECHNOLOGY TO SPEAK THE DESCRIPTIVE ATTRIBUTE TO A USER.” (CLAIM 11[D])	306
12. CLAIM 12.....	307
13. CLAIM 13.....	307
14. CLAIM 14.....	307

-xxix-

TABLE OF CONTENTS
(continued)

	Page
15. CLAIM 16.....	308
16. CLAIM 17.....	308
17. CLAIM 18.....	309
18. CLAIM 19.....	309
19. CLAIM 20.....	309
H. NO SECONDARY CONSIDERATIONS SUPPORT THE PATENTABILITY OF CLAIMS 1-5, 7-13, 16-20, AND 22-28 OF THE '532 PATENT	310
VIII. CONCLUSION.....	310

Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

I, James Fogarty, Ph.D., declare as follows:

I. INTRODUCTION AND QUALIFICATIONS

A. Introduction

1. I have been retained by counsel for accessiBe, Ltd. (“accessiBe” or “Petitioner”) as a technical expert witness in connection with the petition for *inter partes* review (“IPR”) of U.S. Patent No. 11,061,532 (“Ex. 1001” or “’532 patent”). I further understand that the ’532 patent is a continuation of application No. 17/147,323, filed on January 12, 2021, which itself is a continuation of application No. 16/430,210, which was filed on June 03, 2019, which itself is a continuation of application No. 15/074,818, which was filed on March 18, 2016, now U.S. Patent No. 10,444,934. For purposes of my analysis herein, I have used the March 18, 2016 date as the relevant time period.

2. I have been asked by accessiBe to offer opinions regarding the ’532 patent, including the patentability of the claims in view of certain prior art references and the knowledge of a person of ordinary skill in the art. This Declaration sets forth the opinions I have reached to date regarding these matters.

3. In preparing this Declaration, I have reviewed the ’532 patent, its prosecution history, and each of the documents I reference herein. In reaching my opinions, I have relied upon my experience in the field and have also considered

Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

the viewpoint of a person of ordinary skill in the art at the time of the '532 patent's priority date. As explained below, I am familiar with the level of skill of a person of ordinary skill in the art regarding the technology at issue as of that time frame.

4. I am being compensated for my time working on this matter at my standard hourly rate of \$400 per hour plus expenses. I do not have any personal or financial stake or interest in the outcome of the present proceeding, and the compensation is not dependent on the outcome of this IPR and in no way affects the substance of my statements in this declaration.

B. Qualifications and Experience

5. My qualifications for forming the opinions set forth in this Declaration are summarized here and explained in more detail in my curriculum vitae, which is attached as Exhibit 1003.

6. I am currently a Professor of Computer Science & Engineering at the University of Washington ("UW") and an Associate Director of the Center for Research and Education on Accessible Technology and Experiences (UW CREATE), a leading UW cross-campus interdisciplinary center whose aim is to make technology accessible and to make the world accessible through technology, which was launched with an initial \$2.5M gift from Microsoft.

Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

7. I am also Director of DUB, the UW's a cross-campus initiative advancing all aspects of Human-Computer Interaction and Design ("HCI") research and education, one of the world's top HCI communities. I also served on the founding executive committee for UW's interdisciplinary Master of HCI and Design, on the founding Executive Council for the UW's Population Health Initiative, and in the founding of UW Medicine's Digital Health Advisory Committee.

8. My broad research interests are in Human-Computer Interaction and User Interface Software and Technology, often with a focus on the role of software frameworks and tools in developing, deploying, and evaluating new approaches to the human obstacles surrounding everyday intelligent technologies.

9. My prior research includes new applications of low-cost and unobtrusive personal and environmental sensing, of non-expert machine learning, and of tools for runtime modification of graphical interfaces. My work with machine learning includes research on context awareness, web image search, social network friend groups, and gesture recognition. Our research helped define common approaches to everyday interaction with machine learning (e.g., "Show Similar Images" functionality in Microsoft's web image search) and common approaches to machine learning in software developer tools.

Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

10. Much of my work has had direct applications to the field of accessibility. For example, my work on runtime interpretation and modification of interfaces explores how to enable new approaches to deploying interaction techniques in already existing technology ecosystems (e.g., enabling third-party modification of interaction with Android apps, for goals such as improving accessibility using a screen reader).

11. My research group has been leading the largest-known open analyses of the accessibility of Android apps, providing new understanding of the current state of mobile accessibility and new insights into factors in the ecosystem that contribute to accessibility failures. Our analyses of common failure scenarios has directly led to Google improvements in the accessibility ecosystem (e.g., corrections to Android documentation code snippets that were previously inaccessible and therefore creating many accessibility failures as developers used those inaccessible snippets in their own apps) and motivated additional research (e.g., longitudinal analyses of accessibility, including the introduction of new accessibility failures and the repair of previously-existing failures in developers correctly applying accessibility metadata). Some of this work was published in connection with the ASSETS 2017, ASSETS 2018, and CHI 2022 conferences and in the journal TACCESS in 2020.

Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

12. My group has also been developing new techniques for runtime repair and enhancement for mobile accessibility, providing the ability to support third-party runtime enhancements within Android's security model and without requiring modification to existing accessibility services or apps. We have applied these approaches to accessibility repair (e.g., techniques to allow social annotation of apps with missing screen reader data), and to enable entirely new forms of tactile accessibility enhancements. These techniques provide a basis for both improving current accessibility and exploring new forms of future accessibility enhancements. Some of this work has been published in connection with the CHI 2017, ASSETS 2018, and UIST 2018 conferences.

13. My group also developed some of the earliest techniques for pixel-based recognition of interface elements in support of runtime enhancements, including enhancements to improve accessibility. Runtime modification research prior to our own had generally relied upon complete existing accessibility data, upon brittle and inflexible techniques for injecting code directly into existing applications, or upon extremely limited representations that could not support many desired enhancements. We showed techniques for direct recognition of the pixel level appearance of entire interface hierarchies to support runtime enhancements, including implementation of a target-aware pointing technique for

Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

improving accessibility for people with motor impairments. Some of this work was published in connection with the CHI 2010, CHI 2011, CHI 2012, CHI 2014, and UIST 2014 conferences.

14. Our work has maintained strong visibility and impact across multiple major technology organizations. My students have consistently been recruited and hired for HCI and accessibility internships and full-time positions at Apple, Google, and Microsoft. I have presented our accessibility research to Microsoft groups focused on accessibility and applications of artificial intelligence and machine learning in accessibility, including to Microsoft's Chief Accessibility Officer. I have presented our accessibility research to a Google workshop on Machine Learning for Accessibility, including to Google's Director of Accessibility. Our work directly informed Apple's 2020 deployment of Screen Recognition functionality applying on-device recognition to improve the accessibility of image elements, functionality that was implemented in part by students hired directly from my research group based on our accessibility research.

15. I have consistently published in the field of accessibility and related underlying software techniques since at least 2009, including our CHI 2009 development of a target-agnostic pointing enhancement for people with motor impairments and the above-described techniques for interface enhancements. Our

Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

interface research has included innovative interface enhancements in traditional desktop interfaces, in web-based interfaces, and in mobile interfaces, so I am knowledgeable in these multiple interface contexts, their commonalities, and their differences. I have similarly included accessibility and related underlying software techniques in my graduate and undergraduate teaching since at least 2008. I have also served on and chaired several committees for leading research conferences in human-computer interaction and related software. My work has been directly supported by the National Science Foundation, by the Agency for Healthcare Research and Quality and by the National Institutes of Health, and by unrestricted gifts from multiple leading technology organizations including Adobe, Google, Intel, and Microsoft.

C. Materials Considered

16. The analysis that I provide in this Declaration is based on my education and experience in the field of software development including for accessibility applications, as well as the documents I have considered, including the '532 patent (Ex. 1001) and its prosecution history (Ex. 1004). The '532 patent states on its face that it issued from an application filed on February 5, 2021 and claims priority to U.S. Patent No. 10,444,934, which was filed on March 18, 2016. For the purposes of this Declaration, I have assumed March 18, 2016 as the

Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

effective filing date for the '532 patent. I have cited to the following documents in my analysis below:

EXHIBIT NO.	DESCRIPTION
1001	U.S. Patent No. 11,061,532 to Sean D. Bradley, <i>et al.</i> , (filed Feb. 5, 2021, issued July 13, 2021)
1004	Prosecution History of U.S. Patent No. 11,061,532 (excerpts).
1005	U.S. Patent Application Pub. No. 2004/0148568 A1 (“Springer”)
1006	U.S. Patent Application Pub. No. 20020156799 A1 (“Markel”)
1007	Hironobu Takagi et al., <i>Collaborative Web Accessibility Improvement: Challenges and Possibilities</i> , 11th International ACM SIGACCESS Conference on Computers and Accessibility, ASSETS 2009, Pittsburgh, Pennsylvania, USA, (Oct. 25-28, 2009), available at https://dl.acm.org/doi/10.1145/1168987.1169018 (“Takagi”)
1008	Jeffrey Bigham et al., <i>WebInSight: Making Web Images Accessible</i> , 8th Int. ACM SIGACCESS Conf. on Computers and Accessibility, ASSETS 2006, Portland, Oregon, USA (Oct. 23-25, 2006), available at https://dl.acm.org/doi/10.1145/1168987.1169018 (DOI:10.1145/1168987.1169018) (“Bigham”)
1009	U.S. Patent Application Pub. No. 2013/0104029 (“Hendry”)
1010	Web Content Accessibility Guidelines (WCAG) v1.0, W3C (May 5, 1999) (WCAG v1.0)

Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

EXHIBIT NO.	DESCRIPTION
1011	Techniques for Web Content Accessibility Guidelines 1.0, W3C Note (Nov. 6, 2000) (“WCAG Techniques v1.0”)
1012	HTML Techniques for Web Content Accessibility Guidelines 1.0, W3C Note (Nov. 6, 2000) (“WCAG HTML Techniques v1.0”)
1013	Web Content Accessibility Guidelines (WCAG) v2.0 (Dec. 11, 2008) (“WCAG 2.0”)
1014	Document Object Model (DOM) Level 1 Specification v1.0, W3C Recommendation (Oct. 1, 1998) (“W3C DOM”)
1015	U.S. Patent Application Pub. No. 2010/0205523 (“Lehota”)
1016	U.S. Patent Application Pub. No. 2002/0124025 (“Janakiraman I”)
1017	Joint Claim Construction Statement, <i>Audioeye, Inc. v. AccessiBe Ltd.</i> , Case No. 6:20-CV-00726-ADA (W.D. Tex. Mar. 31, 2022), ECF No. 38
1018	Jeffrey Bigham, Increasing Web Accessibility by Automatically Judging Alternative Text Quality, 12th International Conference on Intelligent User Interfaces, IUI 2007, Honolulu, Hawaii, USA, (Jan. 28-31, 2007), available at https://dl.acm.org/doi/10.1145/1216295.1216364 (“Bigham 2007”)
1019	U.S. Patent No. 10,394,579 (“Nandakumar”)
1020	History of the Web Content Accessibility Guidelines (WCAG), Bureau of Internet Accessibility (May 6, 2019), available at https://www.boia.org/blog/history-of-the-web-content-accessibility-guidelines-wcag

Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

EXHIBIT NO.	DESCRIPTION
1021	Declaration of Sylvia Hall-Ellis
1022	Web Accessibility Evaluation Tools List (last visited May 10, 2021), available at https://www.w3.org/TR/mobile-accessibility-mapping/
1023	Introduction to Web Accessibility (last visited August 31, 2021), available at https://www.w3.org/WAI/fundamentals/accessibility-intro/
1024	Amended Claim Construction Order, <i>AudioEye, Inc. v. AccessiBe Ltd.</i> , No. 6-20-cv-00997-ADA (W.D. Tex. Nov. 23, 2021), ECF No. 69
1025	<i>Not Used</i>
1026	HTML hidden Attribute (Jan. 15, 2021), originally available at http://www.w3schools.com/tags/att_global_hidden.asp , currently available at https://archive.org/web/
1027	HTML <input> name Attribute (Jan. 15, 2021), originally available at https://www.w3schools.com/tags/att_input_name.asp , currently available at https://archive.org/web/
1028	HTML <input> placeholder Attribute (Jan. 15, 2021), originally available at https://www.w3schools.com/tags/att_input_placeholder.asp , currently available at https://archive.org/web/
1029	HTML <p> Tag (Jan. 15, 2015), originally available at https://www.w3schools.com/tags/tag_p.asp , currently available at https://archive.org/web/

Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

1030	HTML Input Types (Jan. 15, 2015), originally available at https://www.w3schools.com/html/html_form_input_types.asp , currently available at https://archive.org/web/
1031	The ALT-server (July 1997 – Jan. 1998), originally available at https://www.w3.org/WAI/altserv.htm , currently available at https://archive.org/web/ (“The ALT-server”)
1032	Chieko Asakawa et al., <i>Annotation-Based Transcoding for Nonvisual Web Access</i> , ACM Conference on Assistive Technologies, ASSETS 2000, Arlington, Virginia, USA, (Nov. 13-15, 2000), available at https://dl.acm.org/doi/10.1145/354324.354588 (“Asakawa”)
1033	Webvisum-1, July 7, 2008, originally available at http://www.webvisum.com , currently available at https://archive.org/web/ (“Webvisum-1”)
1034	Webvisum-2 (July 3, 2008), originally available at http://www.marcozehe.de/2008/07/03/review-of-the-webvisum-firefox-extension/ , currently available at http://web.archive.org/web/20080902031833/http://www.marcozehe.de/2008/07/03/review-of-the-webvisum-firefox-extension/ (Webvisum-2”)
1035	U.S. Patent No. 8,935,364 (“Teraguchi”)
1036	Philippe Le Hégarret et al., <i>What is the Document Object Model?</i> (Apr. 9, 2002), originally available at https://www.w3.org/TR/2002/WD-DOM-Level-3-Core-20020409/introduction.html , currently available at https://archive.org/web/ (“Hégarret”)

Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

1037	XML Processing (2004), available at https://web.mit.edu/java_v1.5.0_22/distrib/share/docs/guide/xml/jaxp/index.html (“XML Processing”)
1038	Package org.w3c.dom (2004), originally available at https://web.mit.edu/java_v1.5.0_22/distrib/share/docs/api/org/w3c/dom/package-summary.html (“DOM interface”)
1039	Introduction to XML and XML With Java (Nov. 8, 2005), originally available at http://totheriver.com/learn/xml/xmltutorial.html currently available at: https://archive.org/web/ (“XML with JAVA”)

II. LEGAL PRINCIPLES

17. I am not an attorney. For purposes of this declaration, I have been informed by counsel for ACCESSIBe about certain aspects of the law that are relevant to my analysis and opinions, as set forth below.

A. Prior Art

18. I understand that the prior art to the ’532 patent includes patents and printed publications in the relevant art that predate the ’532 patent’s priority date. As I explained previously, I have been instructed to assume for purposes of my analysis that March 18, 2016 is the relevant date for determining what is “prior art.” In other words, I should consider as “prior art” anything publicly available prior to March 18, 2016. I further understand that, for purposes of this proceeding

Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

in the United States Patent Trial and Appeal Board, only patents and documents that have the legal status of a “printed publication” may be relied on as prior art.

19. I am informed that prior art qualifies as a “printed publication” if there is sufficient evidence to establish a reasonable likelihood that the reference was publicly accessible before the critical date of the challenged patent. I understand that copyright dates can be prima facie evidence of publication, and that if conventional markers support a finding that the reference is prior art then, absent evidence to the contrary, there should be no reason to suspect otherwise.

B. Claim Construction

20. I understand that under the legal principles, claim terms are generally given their ordinary and customary meaning, which is the meaning that the term would have to a person of ordinary skill in the art in question at the time of the invention, i.e., as of the effective filing date of the patent application. I further understand that the person of ordinary skill in the art is deemed to read the claim term not only in the context of the particular claim in which a claim term appears, but in the context of the entire patent, including the specification.

21. I am informed by counsel that the patent specification, under the legal principles, has been described as the single best guide to the meaning of a claim term, and is thus highly relevant to the interpretation of claim terms. And I

Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

understand for terms that do not have a customary meaning within the art, the specification usually supplies the best context of understanding the meaning of those terms.

22. I am further informed by counsel that other claims of the patent in question, both asserted and unasserted, can be valuable sources of information as to the meaning of a claim term. Because the claim terms are normally used consistently throughout the patent, the usage of a term in one claim can often illuminate the meaning of the same term in other claims. Differences among claims can also be a useful guide in understanding the meaning of particular claim terms.

23. I understand that the prosecution history can further inform the meaning of the claim language by demonstrating how the inventors understood the invention and whether the inventors limited the invention in the course of prosecution, making the claim scope narrower than it otherwise would be. Extrinsic evidence may also be consulted in construing the claim terms, such as my expert testimony.

24. I have been informed by counsel that, in IPR proceedings, a claim of a patent shall be construed using the same claim construction standard that would be used to construe the claim in a civil action filed in a U.S. district court (which I

Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

understand is called the “*Phillips*” claim construction standard), including construing the claim in accordance with the ordinary and customary meaning of such claim as understood by one of ordinary skill in the art and the prosecution history pertaining to the patent.

25. I have been instructed by counsel to apply the “*Phillips*” claim construction standard for purposes of interpreting the claims in this proceeding, to the extent they require an explicit construction. The description of the legal principles set forth above thus provides my understanding of the “*Phillips*” standard as provided to me by counsel.

26. I understand that some claims are independent, and that these claims are complete by themselves. Other claims refer to these independent claims and are “dependent” from those independent claims. The dependent claims include all of the limitations of the claims on which they depend.

C. Anticipation

27. I understand that a patent claim is anticipated if a single prior art document describes every element of the claim such that a person of ordinary skill in the art (“POSITA”) could practice the claimed method without undue experimentation.

Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

28. I understand that anticipation may be by express disclosure in the prior art. I also understand that if the prior art reference does not expressly set forth a particular claim element, the prior art may still anticipate a patent claim if that element is “inherent” in its disclosure—that is, if it is necessarily found in the reference. A property is inherent even if a POSITA would not have appreciated that property as of the date of that prior art.

D. Obviousness

29. I understand that obviousness is a determination of law based on various underlying determinations of fact. In particular, these underlying factual determinations include (1) the scope and content of the prior art; (2) the level of ordinary skill in the art at the time the claimed invention was made; (3) the differences between the claimed invention and the prior art; and (4) the extent of any proffered objective indicia of non-obviousness. I understand that the objective indicia which may be considered in such an analysis include commercial success of the patented invention (including evidence of industry recognition or awards), whether the invention fills a long-felt but unsolved need in the field, the failure of others to arrive at the invention, industry acquiescence and recognition, initial skepticism of others in the field, whether the inventors proceeded in a direction

Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

contrary to the accepted wisdom of those of ordinary skill in the art, and the taking of licenses under the patent by others, among other factors.

30. To ascertain the scope and content of the prior art, it is necessary to first examine the field of the inventor's endeavor and the particular problem for which the invention was made. The relevant prior art includes prior art in the field of the invention, and also prior art from other fields that a person of ordinary skill in the art would look to when attempting to solve the problem.

31. I understand that a determination of obviousness cannot be based on the hindsight combination of components selectively culled from the prior art to fit the parameters of the patented invention. Instead, it is my understanding that in order to render a patent claim invalid as being obvious from a combination of references, there must be some evidence within the prior art as a whole to suggest the desirability, and thus the obviousness, of making the combination in a way that would produce the patented invention.

32. I further understand that in an obviousness analysis, neither the motivation nor the purpose of the patentee dictates. Rather, any problem known in the field can provide a reason for combining the prior art in the manner claimed.

III. LEVEL OF ORDINARY SKILL IN THE ART

33. I understand that an assessment of claims of the '532 patent should be undertaken from the perspective of a person of ordinary skill in the art as of the earliest claimed priority date, which, as I explained above, I assumed to be March 18, 2016. I have also been advised that to determine the appropriate level of a person having ordinary skill in the art, the following factors may be considered: (1) the types of problems encountered by those working in the field and prior art solutions thereto; (2) the sophistication of the technology in question, and the rapidity with which innovations occur in the field; (3) the educational level of active workers in the field; and (4) the educational level of the inventor.

34. In my opinion, a person of ordinary skill in the art at the time of the alleged invention of the '532 patent is a person who would have had at least a bachelor's degree in software engineering, computer science, computer engineering, or related field, with at least two years of experience with developing software with accessibility applications. A person could also qualify as a person of ordinary skill with some combination of (1) more formal education (such as a master's of science degree) and less technical experience or (2) less formal education and more technical or professional experience with accessibility-focused software development.

Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

35. My opinions regarding the level of ordinary skill in the art are based on, among other things, my experience developing, patenting, and testing new software including for accessibility applications, and my experience teaching and mentoring students at graduate and undergraduate levels. Although my qualifications and experience exceed those of the hypothetical person having ordinary skill in the art, my analysis and opinions regarding the '532 patent have been based on the perspective of a person of ordinary skill in the art as of March 18, 2016.

IV. BACKGROUND OF THE TECHNOLOGY

A. Web Pages and Document Object Models (DOMs)

36. Web sites and other documents designed to be displayed in web browsers are typically written using HTML, or HyperText Markup Language. When a web browser is directed to a website's URL (e.g., <https://www.google.com>), the web browser receives the HTML documents, typically from a web server, and renders the documents into web pages by interpreting the HTML code. HTML code may include elements such as images, objects, and forms. They may also include tags which are not displayed by the browser but are used to interpret the content of the webpage. Some tags are used to identify HTML elements (e.g.,), while others may provide other

Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

information such as text. HTML can also be used to embed programs, including those written in JavaScript.

37. When a web browser is directed to a website's URL, in addition to receiving HTML documents from the web server, the web browser may also receive scripts which may include JavaScript. These scripts may perform a number of operations when loaded by a web browser including the modification of elements of an associated website's DOM.

38. When a browser loads a web page, it usually translates the HTML code into a "DOM," or "Document Object Model." This refers to a programming interface (an "object model") for documents that typically represents HTML and XML documents (e.g., website code) in a "tree-like" logical structure that is conducive to being manipulated by scripts and other software. Ex. 1014, at 10-12 (W3C DOM v1.0) (noting that the W3C's DOM allows "JavaScript scripts and Java programs to be portable among Web browsers").

39. When a browser loads a web page, it may also execute script in connection with the web page. The script can manipulate the web page based on either its HTML or a generated DOM. Browsers and other web technologies have long provided support for various extensions and events by which scripts can perform such modifications (e.g., supporting scripts in manipulating HTML

Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

content of an HTTP response before it is used to generate a DOM, providing events that support scripts in manipulating a DOM during or after various stages of its generation).

40. Beginning with the April 2011 version 5 of the HTML specification, the innerHTML attribute of every HTML DOM element has also provided access to and the ability to modify a DOM element's corresponding HTML. *See* Ex. 1023. Accessing the innerHTML attribute of a DOM element “[r]eturns a fragment of HTML or XML that represents the element's contents.” *Id.*, at Section 3.5.5. Scripts can manipulate this HTML as desired, and setting the innerHTML attribute of the DOM element will “replace the contents of the element with nodes parsed from the given string.” *Id.*

41. Available on every HTML DOM element (i.e., up to and including the root Document node generated from an entire HTML document), the innerHTML attribute therefore allows scripts to manipulate a preferred representation of a web page (i.e., to prefer to manipulate HTML, to prefer to manipulate the DOM, or to mix the two representations at different points in a script's manipulation of a web page). The browser's automatic conversion between these two representations according to the preferences of a script highlights the extent to which the HTML and DOM representations are interchangeable and can both be modified.

Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

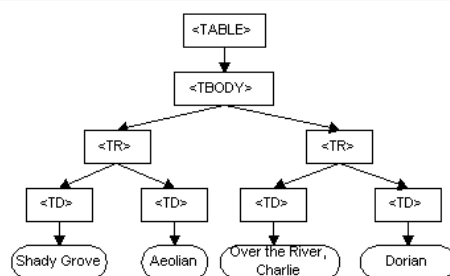
42. An example of a HTML document represented as a DOM tree-like structure is shown below, although both representations of the code are essentially identical other than the visual layout of the hierarchical structure.

What the Document Object Model is

The DOM is a programming API for documents. It closely resembles the structure of the documents it models. For instance, consider this table, taken from an HTML document:

```
<TABLE>
<TBODY>
<TR>
<TD>Shady Grove</TD>
<TD>Aeolian</TD>
</TR>
<TR>
<TD>Over the River, Charlie</TD>
<TD>Dorian</TD>
</TR>
</TBODY>
</TABLE>
```

The DOM represents this table like this:



DOM representation of the example table

Ex. 1014, at 10-11. Since their introduction by W3C in around 1998, DOMs have become the dominant means of programmatically manipulating web pages.

43. HTML is a hierarchical file representation of a webpage that is parsed to create a DOM (i.e., a model) and the DOM is then rendered by one or more views in the browser. A person of ordinary skill would therefore often use the terms HTML or DOM interchangeably, because the HTML used to author a

Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

website is a hierarchical file representation that directly corresponds to the hierarchical DOM that is created when a browser parses and then renders the HTML. Modifications made in one are similarly straightforward to instead make in the other (e.g., adding, removing, or modifying elements in the hierarchy) by either modifying the HTML representation or the in-memory model representation.

44. Interfaces for web code are generally represented as hierarchies, with HTML and the DOM being examples. This is generally due to containment, because an interface element may contain one or more children interface elements. At runtime, each element in a hierarchy indicates some piece of browser-provided code that is responsible for rendering that element.

45. Furthermore, techniques for obtaining a DOM from an XML/HTML file using Java code were well known before the priority date of the '532 patent. Ex. 1036; Ex. 1037; Ex. 1038; Ex. 1039. For example, “[t]he DOM originated as a specification to allow JavaScript scripts and Java programs to be portable among Web browsers” and “[a]nything found in an HTML or XML document can be accessed, changed, deleted, or added using the Document Object Model.” Ex 1036, at ¶¶2, 10. Also, it was known that “[t]he Java™ API for XML processing specification includes by reference both the abstract semantics described for the DOM Level 3 Core Recommendation interfaces” including a “Document Object

Model Level 2 Core API” that “allows programs to dynamically access and update the content and structure of documents. See the specification for more information.” Ex 1037, ¶3; Ex 1038, ¶2. Furthermore, it was well known that Java code existed to obtain a DOM from an XML/HTML file. *See* Ex. 1039, at Section 6.1.1 (a) (describing a parse method to “get DOM representation of the XML file”).

B. Web Accessibility and Web Accessibility Guidelines

46. The need for websites to be accessible to and navigable by individuals with visual, motor, or other impairments has been well recognized since the early days of the internet. In an effort to address this need, web accessibility guidelines were introduced decades ago to provide techniques and metrics that could be implemented by web developers and evaluators to ensure that websites, tools, and other technologies are designed in a way that people with disabilities can use them.¹

47. For example, the W3C (World Wide Web Consortium²) Web Accessibility Initiative (WAI) develops technical specifications, guidelines, techniques, and supporting resources that describe accessibility solutions, which

¹ *See* Introduction to Web Accessibility, available at <https://www.w3.org/WAI/fundamentals/accessibility-intro/>.

² *See* About W3C, available at <https://www.w3.org/Consortium/>

Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

include the Web Content Accessibility Guidelines (WCAG), Authoring Tool
Accessibility Guidelines (ATAG), ARIA for Accessible Rich Internet
Applications, and other similar resources.³

48. Perhaps the most well-known accessibility guideline is WCAG (or the
Web Content Accessibility Guidelines), published by W3C since at least May 1999
to “explain how to make Web content accessible to people with disabilities.” *See*
Ex. 1010 (WCAG v1.0). WCAG 2.0 was published in 2008, building on v1.0 and
applying the same principles to mobile apps and documents. WCAG version 2.1
was published in 2018. Ex. 1020 (history of WCAG); Ex. 1013 (WCAG 2.0). To
this day, WCAG is widely considered to be the gold standard in web accessibility.
WCAG 2.0 has been adopted as an international standards for web accessibility.⁴

49. Web accessibility guidelines such as WCAG are intended to address a
number of disabilities that affect access to the internet, including auditory,
cognitive, neurological, physical, speech, and visual disabilities. *Id.* Most relevant
to the '532 patent are the guidelines aimed at addressing visual disabilities.

³ *See* Introduction to Web Accessibility, available at
<https://www.w3.org/WAI/fundamentals/accessibility-intro/>; W3C Accessibility
Standards Overview, available at <https://www.w3.org/WAI/standards-guidelines/>.

⁴ *See* Introduction to Web Accessibility, available at
<https://www.w3.org/WAI/fundamentals/accessibility-intro/>.

Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

50. For example, one common issue faced by visually impaired individuals navigating the internet is that many websites fail to provide meaningful textual descriptions of web elements, e.g., untagged elements, such as icons, images or other graphics, input fields on forms, or links to other web pages. Meaningful textual descriptions (often referred to as labels for fields or as alt text in the context of images) can be read aloud by text-to-speech technology (e.g., audio players such as screen readers) so visitors of a webpage or website understand the content of a page even though they cannot see it. When websites include elements lacking textual descriptions or untagged elements, visually impaired individuals may be unable to understand or navigate web pages because accurate descriptions are not read aloud. *See* Ex. 1010 (WCAG v1.0) at App'x B (defining screen readers); Ex. 1011 (W3C Note: Techniques for WCAG v1.0); Ex. 1012 (W3C Note: HTML Techniques for WCAG v1.0).

51. From its earliest version, WCAG included techniques to make websites accessible, including modifying existing inaccessible code for improved accessibility. This could be done either by directly modifying the HTML code or by modifying the Document Object Model or "DOM." Ex. 1012 (HTML Techniques v1.0); Ex. 1010 at App'x B (Glossary); Ex. 1014 (W3C DOM).

Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

52. For example, in the context of untagged elements including input fields, procedures for bringing websites or web applications into compliance with accessibility guidelines – include W3C published instructions showing how to insert textual descriptions for input fields directly into HTML code and what such descriptions should look like. Ex. 1010 (WCAG v1.0), at *Abstract*; Ex. 1011 (WCAG Techniques v1.0); Ex. 1012 (WCAG HTML Techniques v1.0). WCAG directs web developers to find input fields in the code that lack textual descriptions, examine those input fields to determine an appropriate and meaningful textual description, and then add such textual descriptions directly into the code (HTML in the examples) as, e.g., “label” text associated with the untagged input field. Ex. 1012 (WCAG HTML Techniques v1.0) § 11.

11.3 Labeling form controls

Checkpoints in this section:

- [12.4](#) Associate labels explicitly with their controls. [Priority 2]
- [10.2](#) Until user agents support explicit associations between labels and form controls, for all form controls with implicitly associated labels, ensure that the label is properly positioned. [Priority 2]

An example of [LABEL](#) used with “[for](#)” in HTML 4.01 is given in the previous section.

A label is implicitly associated with its form control either through markup or positioning on the page. The following example shows how a label and form control may be implicitly associated with markup.

Example.

```
<LABEL for="firstname">First name:
<INPUT type="text" id="firstname" tabindex="1">
</LABEL>
```

Ex. 1012, at § 11.

Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

53. In the context of untagged elements including images, WCAG (v1.0) also addresses this scenario, recognizing that “[s]ome users may not be able to see images, others may use text-based browsers that do not support images, while others may have turned off support for images (e.g., due to a slow Internet connection). The guidelines do not suggest avoiding images as a way to improve accessibility. Instead, they explain that providing a text equivalent of the image will make it accessible.” Ex. 1010 (WCAG v1.0), at § 1 (emphasis in original). As such, WCAG places textual equivalents for images (sometimes referred to as “alternate text” or “alt text”) at the highest priority level among accessibility guidelines. *Id.*, at § 6, Guideline 1. When a web page includes, for example, an image without alt text, WCAG instructs a developer or evaluator to fix that page by modifying existing website code (e.g., HTML) to insert meaningful textual descriptions for images where they previously did not exist. Taking the example of an image without alt text, W3C’s techniques for following WCAG 1.0 included instructions on how to determine meaningful textual descriptions for graphics/images and how to insert those descriptions directly into HTML code. *Id.*, at *Abstract*; Ex. 1011 (WCAG Techniques v1.0); Ex. 1012 (HTML Techniques for Web Content Accessibility Guidelines 1.0). Those instructions included determining a meaningful textual description for each image without such a

Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

description, and adding those descriptions directly into the code (HTML in the examples) as, e.g., “alt” or “longdesc” text.

7.1 Short text equivalents for images ("alt-text")

Checkpoints in this section:

- [1.1](#) Provide a text equivalent for every non-text element (e.g., via "alt", "longdesc", or in element content). *This includes:* images, graphical representations of text (including symbols), image map regions, animations (e.g., animated GIFs), applets and programmatic objects, ASCII art, frames, scripts, images used as list bullets, spacers, graphical buttons, sounds (played with or without user interaction), stand-alone audio files, audio tracks of video, and video. [Priority 1]

When using [IMG](#), specify a short [text equivalent](#) with the "alt" attribute. **Note.** The value of this attribute is referred to as "alt-text".

Example.

```
<IMG src="magnifyingglass.gif" alt="Search">
```

When using [OBJECT](#), specify a [text equivalent](#) in the body of the OBJECT element:

Example.

```
<OBJECT data="magnifyingglass.gif" type="image/gif">  
  Search  
</OBJECT>
```

7.2 Long descriptions of images

Checkpoints in this section:

- [1.1](#) Provide a text equivalent for every non-text element (e.g., via "alt", "longdesc", or in element content). *This includes:* images, graphical representations of text (including symbols), image map regions, animations (e.g., animated GIFs), applets and programmatic objects, ASCII art, frames, scripts, images used as list bullets, spacers, graphical buttons, sounds (played with or without user interaction), stand-alone audio files, audio tracks of video, and video. [Priority 1]

When a short text equivalent does not suffice to adequately convey the function or role of an image, provide additional information in a file designated by the "[longdesc](#)" attribute:

Example.

```
<IMG src="97sales.gif" alt="Sales for 1997"  
  longdesc="sales97.html">
```

In sales97.html:

A chart showing how sales in 1997 progressed. The chart is a bar-chart showing percentage increases in sales by month. Sales in January were up 10% from December 1996, sales in February dropped 3%, ..

Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

Ex. 1012, at § 7.

54. These guidelines and resources are used by web developers and accessibility providers to make web pages accessible, either by creating webpages to be accessible outright or by fixing or remediating previously inaccessible content. At first, this was done by manually reviewing the code of a web page and fixing problems as they were identified. For example, this might include: (1) accessing a website's code; (2) analyzing the code to identify untagged elements such as input fields, images or other elements without meaningful descriptions; (3) looking at the untagged elements to determine an appropriate textual description; and (4) fixing the inaccessible code by adding appropriate textual descriptions for the untagged element. It soon became clear that automated methods for identifying and fixing accessibility problems would be more efficient. *See, e.g.*, Ex. 1007 (Nandakumar), at 1:49-51 (“manually fixing inaccessible elements by setting appropriate accessibility properties is tedious and often time-consuming.”).

C. Automated Accessibility Checking and Remediation Tools

55. In order to more efficiently make websites compliant with accessibility guidelines, a number of tools have been introduced to automatically inspect website code against accessibility guidelines to either identify, or identify

Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

and remediate, accessibility problems in websites. This was done well before the '532 patent's priority date of March 18, 2016.

56. WC3 itself identifies a myriad of web accessibility evaluation tools—“software programs or online services that help you determine if web content meets accessibility guidelines”—that could be used for this purpose. Many of these tools predate the '532 patent, with some going as far back as the early 2000s.⁵

57. Also well before the '532 patent, developers created tools that not only evaluated webpages, but remediated them too by automatically modifying website code to fix the detected accessibility issues. *See, e.g.*, Ex. 1005 (Springer) (filed in 2001); Ex. 1008 (Bigham) (published in 2006); Ex. 1009 (Hendry) (filed in 2011).

58. One such example was U.S. Publication No. 2004/0148568A1 (“Springer,” Ex. 1005), which is relied on and described in greater detail in the invalidity grounds below. Springer was filed in 2001, and is titled “Checker and fixer algorithms for accessibility standards.” As its title suggests, Springer discloses a software system containing a collection of modules (referred to as “checkers”) that search for various accessibility compliance issues in web pages,

⁵ *See* Ex. 1022 (Web Accessibility Evaluation Tools List).

Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

and another collection of modules (referred to as “fixers”) that correct the detected accessibility issues. This included the capability of adding a textual description for untagged elements (e.g., labels for input fields or alt text to images) that were previously lacking it.

59. U.S. Publication No. 2010/020523 (“Lehota,” Ex. 1015) discloses another such tool. More specifically, Lehota, filed in February 2009, discloses systems and methods for increasing and enabling accessibility of web applications and web pages accessible through web browsers. Ex. 1015 at *e.g.*, *Abstract*, ¶¶1, 4, 9. This is illustrated below:

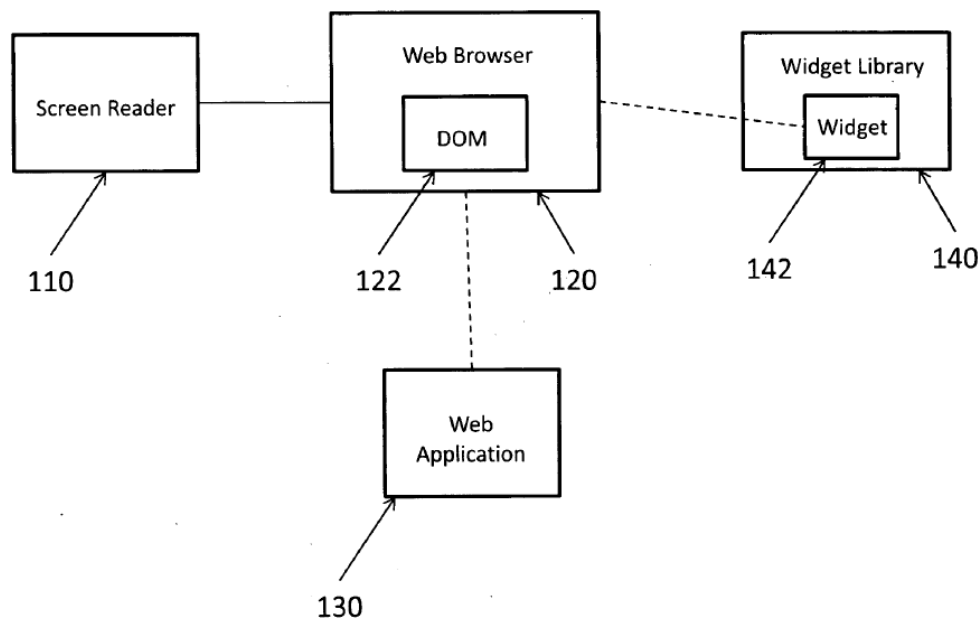


FIG. 1

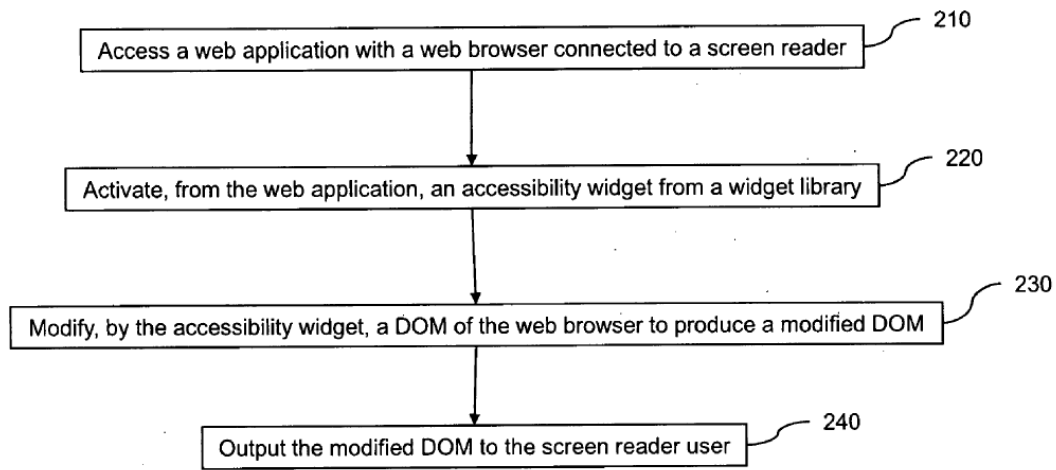


FIG. 2

Ex. 1015, at Figures 1-2.

60. In Lehota's system, when a web application or web page is accessed by the web browser, an "accessibility widget" is activated that dynamically fixes accessibility problems in the web page by modifying the DOM. *Id.*, at *e.g.*, ¶¶8-9, 15, 18. The modified DOM is then output to the screen reader user and/or a rendered web page. *Id.*, at *e.g.*, ¶9. Lehota lists a number of common accessibility issues it fixes where it fixes missing textual description for untagged elements, including "missing 'ALT' tags for images [and] inadequate use of the 'Label' tag for screen elements." *Id.*, at ¶2. To remedy these issues, Lehota discloses "a system for increasing accessibility of a web application to a screen reader" that includes "[a]t least one accessibility widget, from a widget library, . . . to modify the DOM of the web browser." *Id.*, at ¶4. Also, Lehota discloses a system

Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

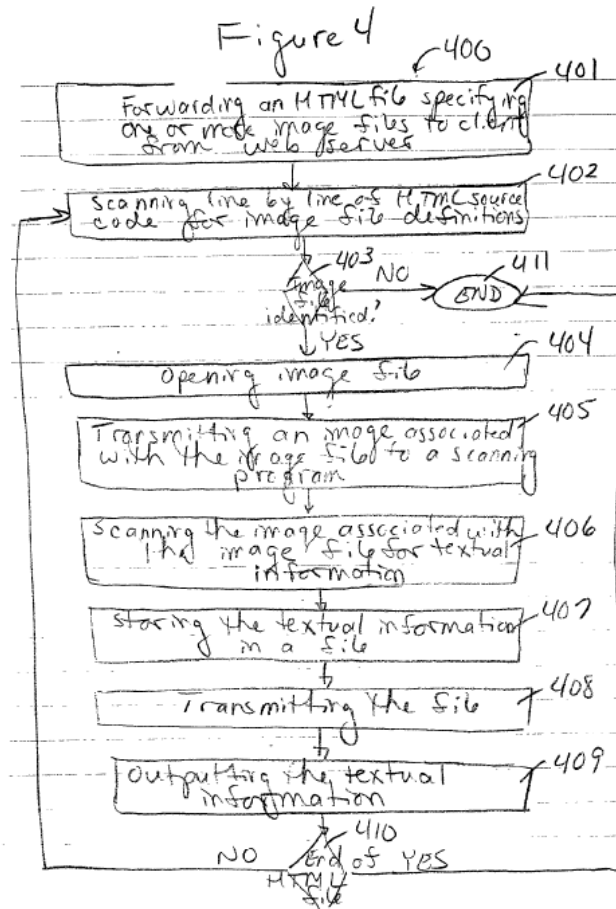
“provides a JavaScript call to a widget library (a collection of files written primarily in JavaScript) that is embedded in a web application. This single call is sufficient to invoke a new web widget (also referred to herein as “accessibility widget”) that dynamically fixes accessibility problems in the web page, executing directly in the client web browser” and, because changes are made directly into the DOM, does not require modification of the underlying vendor website. *Id.*, at ¶15. The disclosed accessibility widget “dynamically fixes accessibility problems with the web page by . . . manipulating the DOM resident in memory by parsing, and applying any additions, deletions, or modifications to the objects directly in the DOM.” *Id.*, at ¶18. In one example, Lehota discloses “the modifying the DOM includes . . . modifying properties of objects in the DOM” and “[t]he objects in the DOM include standard and non-standard objects identified in hypertext markup language (HTML)” such as a “image, . . . input, span, division tag, label” tags among others. *Id.*, at ¶7.

61. Although Lehota disclosed fixing missing textual descriptions for untagged elements such as assigning alt-tags to images without alt text based on pre-configured values or the title attribute of an image (or even assigning a null value), a person of ordinary skill in the art would have understood that the techniques were not so limited, especially in light of later references and the

Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

rapidly improving image recognition technology that was being developed around the same time as Lehota, as well as the known techniques of evaluating context in a website or using optical character recognition (“OCR”) to identify appropriate textual descriptions for images. Indeed, a pre-configured value for alt text of an image could come from a database of manually-input values or a database of values as determined/input by AI image recognition service.

62. For example, another system intended to address similar problems is disclosed even earlier in U.S. Publication No. 2002/0124025 (“Janakiraman I,” Ex. 1016), filed in 2001. Janakiraman I discloses a system and methods for “assisting individuals with disabilities through technology, and more particularly to scanning and outputting textual information in web page images in order to promote accessibility to individuals with disabilities.” Ex. 1016, at ¶5. Janakiraman I’s method included: (1) scanning HTML code for a website line-by-line to identify image files; (2) sending each image file to a “scanning program”; (3) using the scanning program to perform OCR on images to extract textual information; and (4) sending the textual information to the browser so that it could be read by assistive technology like screen readers. *Id.*, at *Abstract*, ¶18. These steps are illustrated in Figure 4.



Ex. 1016, at Figure 4.

63. Other references also recognized alternative methods of labeling alternative text including context-based labeling. A 2007 article by Jeffrey Bigham (herein referred to as “Bigham 2007”), entitled “Increasing Web Accessibility by Automatically Judging Alternative Text Quality,” also recognizes that “the appropriateness of alternative text is highly dependent on the context where the image appears.” Ex. 1018 (Bigham 2007), at 350. Thus, Bigham 2007 “built a classifier that uses features based primarily on the alternative text content

Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

and on the context of where it is located.” *Id.* Bigham 2007 also discloses a method for “filename analysis” that automatically determines appropriate alt text for images based on heuristics that break filenames into words by replacing characters commonly used to separate words (such as underscores and dashes) and by doing a dictionary-based search for words within the filename. *Id.*, at 350-351. It also describes WebInSight’s link-based labeling method that formulates alternative text by following the link and returning candidate alternative text from the title and headings on the linked page. *Id.*, at 350.

64. For issues that could not be automatically remediated, it was well-known that users could manually fix existing issues. For instance, a publication from the 1990’s describes “human generation” of alt text for images (“ALT-server,” Ex. 1031). The following process is described: (1) a browser requests an initial HTML document; (2) the returned HTML contains an image lacking alt-text; (3) the browser queries a server for the textual description; (4) sighted users enter the description in the input field and submit the form to the server; and (5) the server validates the text. *Id.*, at 2-4. ALT-server also recognizes that as “the automatic extraction part is improved, this will also diminish the number of images actually needing some human collaboration.” *Id.*, at 4.

Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

65. A 2006 article entitled “WebInSight: Making Web Images Accessible” by Jeffrey Bigham et al. (“Bigham,” Ex. 1008) also describes a system for automatically determining and inserting alt text into web page code based on “human labeling,” as well as context analysis and OCR.

Images without alternative text are a barrier to equal web access for blind users. To illustrate the problem, we conducted a series of studies that conclusively show that a large fraction of significant images have no alternative text. To ameliorate this problem, we introduce WebInSight, a system that automatically creates and inserts alternative text into web pages on-the-fly. To formulate alternative text for images, we present three labeling modules based on web context analysis, enhanced optical character recognition (OCR) and human labeling. The system caches alternative text in a local database and can add new labels seamlessly after a web page is downloaded, resulting in minimal impact to the browsing experience.

Ex. 1008, at *Abstract*.

66. It is recognized that while automatic labeling methods worked for a large number of images, labeling by humans was necessary to some extent. Thus, the “WebInSight system provides a mechanism for user to request that an image be sent to a labeling service for labeling by humans.” *Id.*, at 186, col. 2, ¶3. In addition to human-labeling, the publication describes context-based labeling, noting that “[t]he context in which an image appears has previously been leveraged to reveal its contents” (*id.*), and OCR labeling, since many images used on

Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

websites contain some form of text (*id.*, at 185-86). The authors of WebInSight even recognized that others in the field were working on “determining an accurate textual label for an arbitrary image using computer vision techniques,” and that such techniques were still “generally unsolved” but could be applied to automatically determine appropriate textual descriptions of images. *Id.*, at 182.

67. The architecture of Bigham’s WebInSight system is illustrated below:

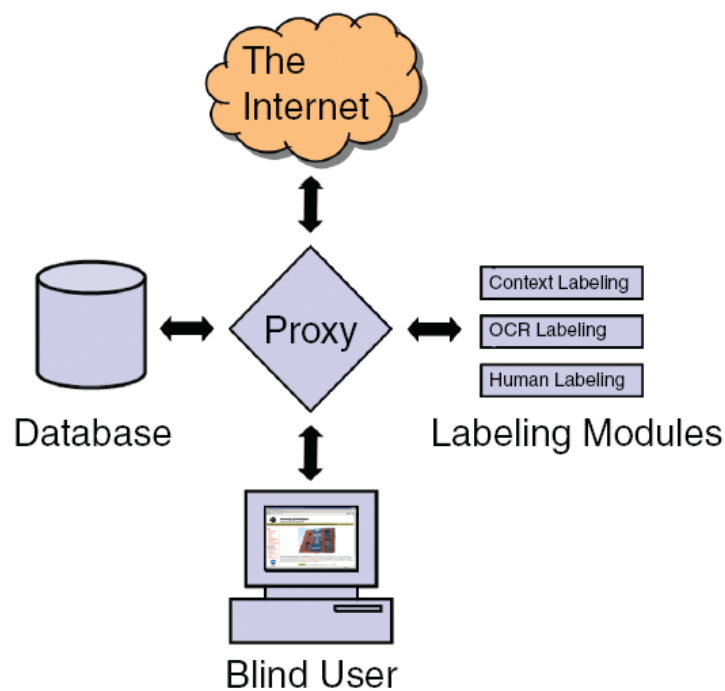


Figure 4: The architecture of WebInSight.

Id., at Figure 4.

Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

68. Bigham also pre-dates the '532 patent. I am informed that Bigham thus qualifies as prior art under at least (post-AIA) 35 U.S.C. § 102(a) because it was published more than one year before the earliest priority date for the '532 patent. I am further informed that Bigham qualifies as a “printed publication” under the standard provided to me because there is sufficient evidence to establish a reasonable likelihood that the reference was publicly accessible well before 2016. Indeed, the declaration of Sylvia Hall-Ellis confirms that Bigham was publicly available in public libraries no later than 2007. *See* Ex. 1021.

69. A 2000 publication by Chieko Asakawa and Hironobu Takagi of IBM, and titled “Annotation-Based Transcoding for Nonvisual Web Access,” (herein referred to as “Asakawa,” Ex. 1032), also recognized the accessibility issues related to visually fragmented groups in Web content and developed an annotation-based transcoding system to convert the already-existing Web pages to be accessible. Asakawa discloses that “[t]he system works between a Web server and a user and consists of two components – one for structural annotations and one for commentary annotations.” Ex. 1032, at 172, Abstract.

70. Specifically, the article discloses that “[t]he structural annotations are used to recognize the visually fragmented groupings and to show the importance and basic role of each group, and the commentary annotations are used to give

Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

users a useful description of each grouping.” *Id.* Commentary annotations not only give users a useful description of each group, but also can be used to describe HTML tags, such as <table> tag, <form> tag, (image) tag, <area> tag and so on. *Id.*, at 177, col. 2, ¶3. Asakawa’s commentary annotations “[d]escribe elements of HTML tags for voice output. *Id.*, at 172, col. 2, ¶2.

71. The following process is described by Asakawa: (1) the transcoding module receives a target HTML document; (2) the transcoding module sends the URL of the target HTML document to the annotation manager; (3) the annotation manager retrieves an is list of proper annotation files and sends it to the annotation database; (4) the transcoding module receives the annotation files that are sent by the annotation database according to the id list and transcodes the target HTML document using the annotation files. *Id.*, at 176, col. 2, ¶1. Also, annotations can be manually input by sighted volunteers. *Id.*, at 177, col. 2, ¶¶3-4. Once an annotation file is registered to the annotation database, it can be used with other pages in the site based on the similarity of the DOM trees. *Id.*, at 176, col. 2, ¶3. The architecture of Asakawa’s annotation-based transcoding system is illustrated in Asakawa’s Figure 4 below (Ex. 1032, at Figure 4).

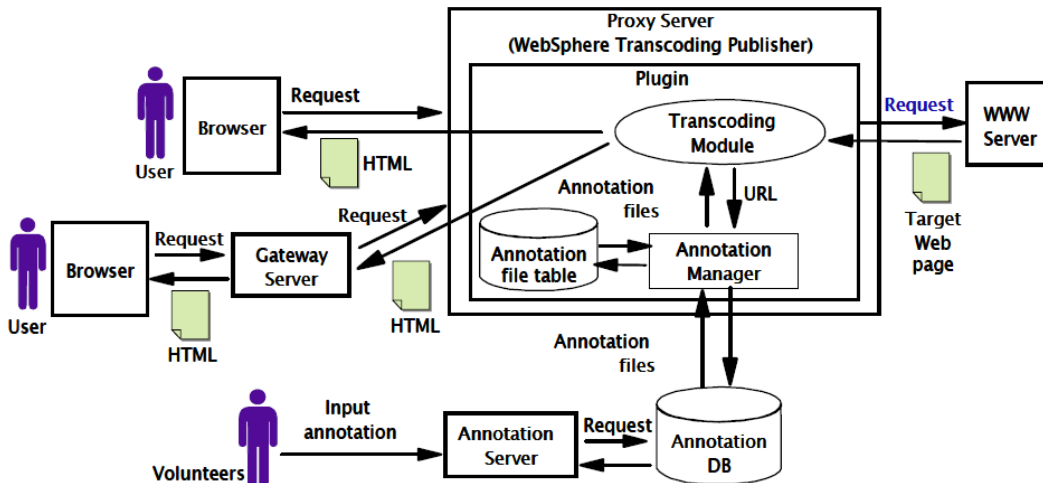


Figure 4: The system architecture

Id., at Figure 4.

72. A 2009 article by Hironobu Takagi et al. (also of IBM) (herein referred to as “Takagi,” Ex. 1007) entitled “Collaborative Web Accessibility Improvement: Challenges and Possibilities,” also recognizes “social accessibility” as a method for improving the accessibility of web sites. Ex. 1007, at 196, col. 2, ¶6. The described method allows end users to report accessibility issues to a supporter community, and in turn, the supporter community creates external metadata, including alternative text labels, to fix the issue. *Id.*

First, an end user reports an accessibility problem on a webpage. This report goes to the supporter community instead of to the site owner (as with a contact form). Then a supporter creates external metadata to fix the reported problem. As a result, the end user can browse the webpage made accessible with the metadata. This reporting-based strategy allows quick solutions to accessibility problems, while reducing the supporters’ work, and focuses on

Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

problems that end-users actually encounter. The public repository allows that metadata to benefit every end user, not just the first user who reported the problem.

Id., at 196, col. 2, ¶6.

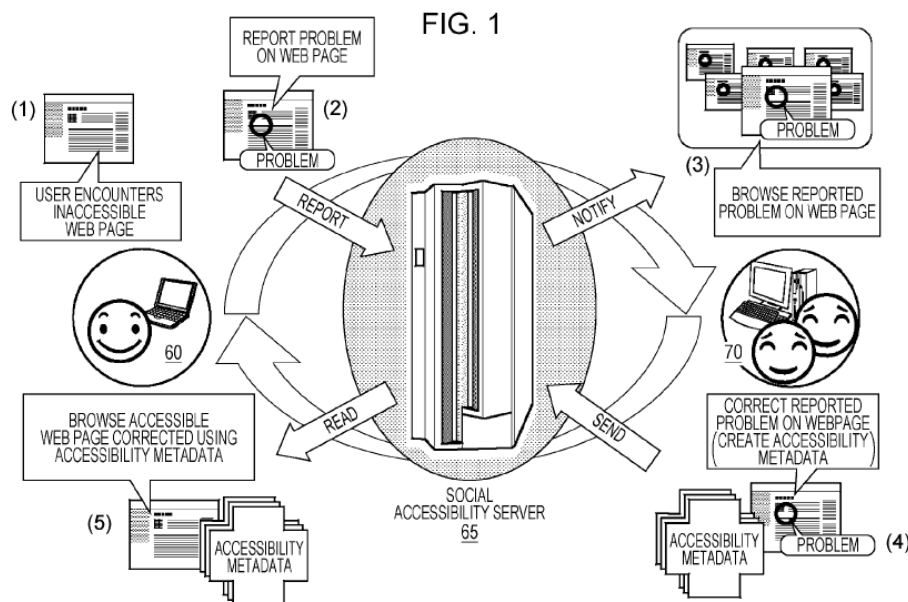
73. Takagi also recognized that social labeling was by no means an innovation and it had already been proposed by many other references, including the 1997 ALT-server, WebInSight, and WebVisum. Ex. 1007 at 196, col. 2, ¶5. ALT-server (Ex. 1031) and Bigam’s WebInSight (Ex. 1008) are discussed above. Webvisum is a free-browser add on, created to “enhance[] web accessibility” was powered by “[c]ommunity driven tagging and page enhancements.” Ex. 1033 (“Webvisum-1”). WebVisum allowed members of the community to “tag graphics, form fields, links, and other page elements,” and to share these changes with the rest of the community. Ex. 1034 (“Webvisum-2”).

74. Takagi pre-dates the ’532 patent. I am informed that Takagi thus qualifies as prior art under at least (post-AIA) 35 U.S.C. § 102(a) because it was published more than one year before the earliest priority date for the ’532 patent. I am further informed that Takagi qualifies as a “printed publication” under the standard provided to me because there is sufficient evidence to establish a reasonable likelihood that the reference was publicly accessible well before 2016.

Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

Indeed, the declaration of Sylvia Hall-Ellis confirms that Takagi was publicly available in public libraries no later than 2009. *See* Ex. 1021.

75. Teraguchi, U.S. Patent No. 8,935,364 (Ex. 1035, “Teraguchi”), filed in Sept. 27, 2012 and published April 4, 2013, discloses a project called Social Accessibility (referred to as “SA” in Teraguchi) which creates external metadata (e.g., remediation data) with the assistance a social network. *Id.*, at 1:18-24. The workflow of SA is depicted in figure 1 of Teraguchi, shown below:



Ex. 1035, at Figure 1.

Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

76. As Figure 1 depicts, (1) a user encounters accessibility issues on a web page; (2) reports the problem to a social accessibility server; (3) the social accessibility server notifies its volunteers of the issue; (4) a volunteer corrects the issue and send the metadata to the accessibility server; (5) the user can now browse the accessible web page. Ex. 1035, at Figure 1, col. 1:25-36.

77. Thus, well-before the priority date of the '532 patent, it was well known in the existing literature that accessibility issues could be manually remediated.

V. THE '532 PATENT

A. Overview of the '532 Patent

78. The '532 patent is titled “Modular Systems and Methods for Selectively Enabling Cloud-Based Assistive Technologies.” Ex. 1001 (the '532 patent), Title. It issued on July 13, 2021 from U.S. Application No. 17/169,229, which was filed on February 5, 2021. Ex. 1001, face page.

79. Like the references discussed above, the '532 patent also relates to automating the process of remediating websites to make them compliant with web accessibility guidelines. *Id.*, at *Abstract*, 2:10-29. In general, the steps disclosed by the '532 patent are similar to those previously used to automatically or manually fix accessibility issues in websites. Those steps include: (1) analyzing

Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

the HTML or DOM associated with a webpage; (2) detecting one or more compliance issues, such as untagged elements lacking adequate textual descriptions; and (3) selecting remediation code to the one or more compliance issues that: (i) determines an appropriate textual description for the untagged element; and (ii) associates the textual description with the untagged element by inserting that description into the code so that it can be read by assistive technology (e.g., a screen-reader tool⁶).

80. This high level process is illustrated, for example, in Figure 5:

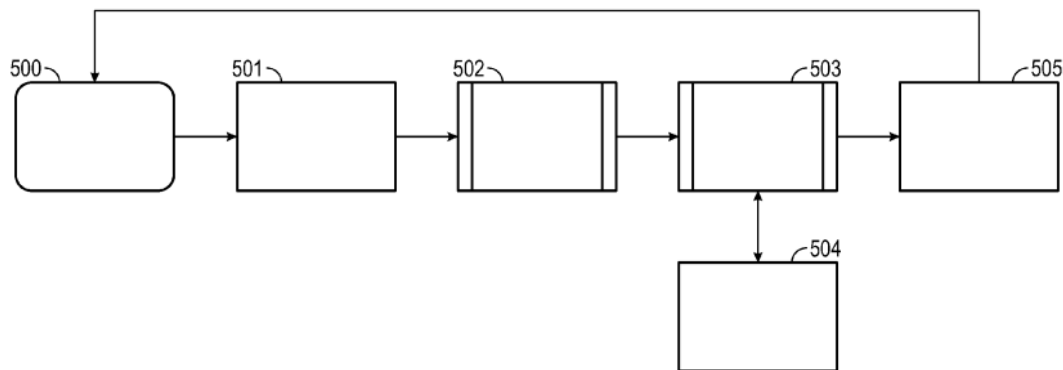


FIG. 5

Id., at Figure 5. The patent does not provide many details on how the various steps are performed. With respect to Figure 5, for example, it states that “once the AE [AudioEye] JavaScript loads **500**, manual remediations present in the configuration

⁶ The '532 patent acknowledges that screen readers were pervasive in the prior art. Ex. 1001, at 1:19-67.

Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

are applied and executed **501**. A process **502** then runs to scan the page to detect compliance issues. Using a heuristics engine **503**, detected compliance issues are mapped to pre-existing remediation scripts **504**. A process **505** then programmatically executes/applies each matched remediation.” *Id.*, at 8:56-63, Figure 5.

81. The ’532 patent claims focus heavily on remediating untagged elements such as unlabeled form fields, such as input fields, or images lacking appropriate alt text. To remediate input fields, the patent recites that the “AE JavaScript may be configured to programmatically inspect nearby text elements and infer and inject an appropriate label based on contextual cues such as the words ‘name,’ ‘first,’ and ‘last’ as well as relative positions and types of attributes, i.e., a SPAN tag in front of an INPUT field.” *Id.*, at 10:18-23. No further details for remediating input fields are disclosed, despite the fact that the claims are all directed to remediating inaccessible input fields. Figure 6 illustrates another feature involving what the patent refers to as “Smart Text Replacement.”

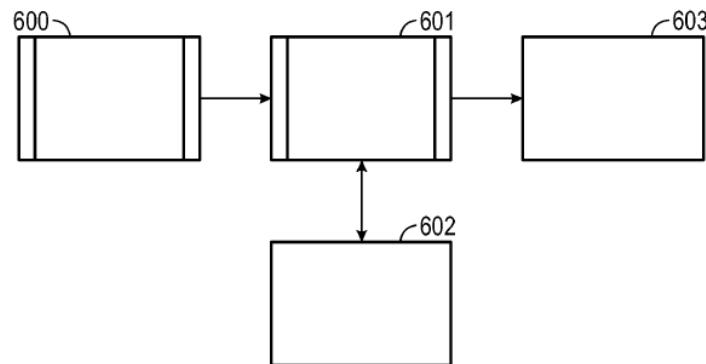


FIG. 6

Id., at Figure 6. According to the '532 patent, sites often share “Like us on Facebook” and “Follow us on Twitter” elements or tracking pixels to count clicks. These too need to be labeled and tagged properly to be handled by screen readers. “The AE JavaScript may be configured to identify common file paths with hyperlinks (e.g., a hyperlink to a Facebook ‘like us’ landing page), and programmatically assign best practice descriptive alt text to the element based on the file path/hyperlink. In this way, when a screen reader or the AE player encounters the element, it is properly and consistently vocalized as ‘like us on Facebook’ even if the element is not coded that way.” Ex. 1001, at 9:31-38.

82. In relation to Figure 6 (excerpted above), the “[smart text lookup] process 600 may be configured to search the DOM for commonly occurring elements (e.g., Facebook Like Us, advertisements, App Store links) that do not have text descriptions or have ambiguous text descriptions. An element

classification process 601 suitably employs a heuristic engine 602 to identify relevant attributes (e.g., H REF) to empower process 603 to apply text description tags to the un-tagged or improperly tagged element.” *Id.*, at 9:39-48.

83. Figure 7 shows an exemplary “smart image description process.”

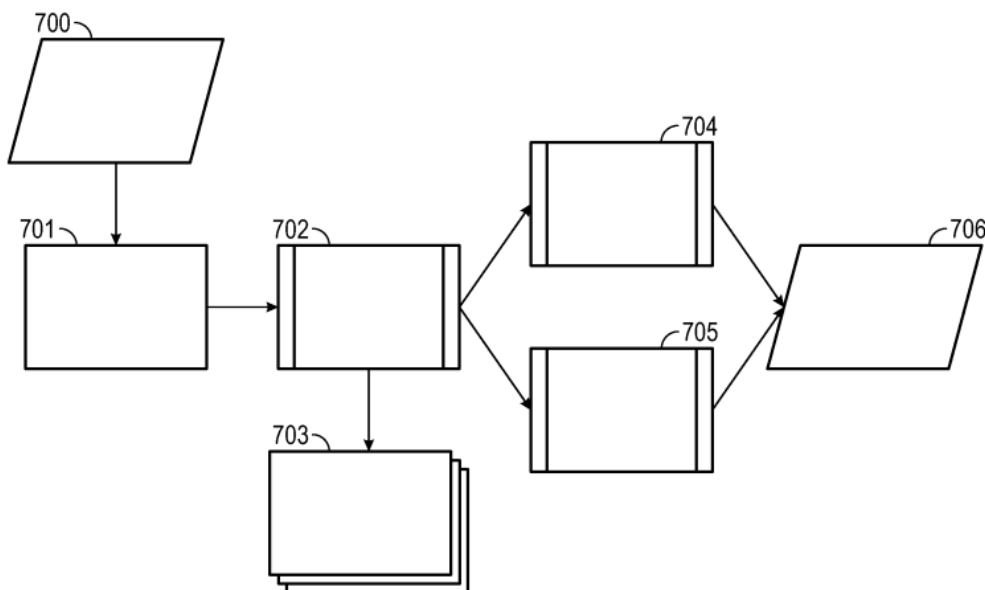


FIG. 7

Id., at Figure 7. Here, “for images lacking an adequate alt text[,] the system may be configured to programmatically send the image to an image recognition server to determine the nature of the image and insert a best fit alt text. . . . The dynamically inserted text can then be spoken to the user, either using a native screen reader or the AE Player.” *Id.*, at 9:1-11. “In particular, an image 700 is passed to a smart image description system 701. A process 702 then sends the image one or more internal and/or external image recognition systems 703, to

Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

obtain one or more interpretations of the image. Image descriptions may then be returned either through a heuristics engine 704 or a natural language engine 705 to generate a human readable, preferably grammatically correct version of the image description, which is then returned 706 to the calling process.” *Id.*, at 9:12-22.

84. Once initial fixes are implemented, “site developers may access a development portal and attend to manual fixers.” *Id.*, at 6:5-10. The ’532 patent states that “a further feature of the present invention involves auto-remediation of manually fillable form fields.” *Id.*, at 10:10-11. The patent then describes this manual process:

In an alternative embodiment, the developer portal allows the developer to write the markup while simultaneously viewing the rendered remediated page in real time thru an iFrame from the same developer portal, without having to separately access the underlying website outside of the developer portal.

Id., at 10:49-54.

85. No further details for manually remediating compliance issues are disclosed, despite the fact the independent claims 1 and 11 are both directed to “manually remediat[ing] a non-programmatically-fixable compliance issue associated with the web page.”

B. Prosecution History

Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

86. The '532 patent issued from U.S. Application No. 17/169,229, which was filed on February 5, 2021. Ex. 1001, face page.

87. The Applicant submitted a Preliminary Amendment prior to Examination which cancelled pending claims 1-41 and added new claims 42-61. Ex. 1004, at 169.

88. The Examiner allowed the amended claims as filed and there were no office actions. In the Notice of Allowance, the Examiner found that “the combination of cited prior art fails to teach the all of features claimed[,]” without noting which features were not disclosed. Ex. 1004, at 50.

89. The examiner listed Hendry as a relevant reference, but did not perform any significant analysis of the patent. Instead, the examiner found Hendry disclosed “automated addition of accessibility features to documents, wherein a first document is analyzed to identify a first set of one or more tags and responsive to identifying the first set of one or more tags, automatically producing a second document based in part on first set of one or more tags, where the second document includes one or more accessibility features that were not in the first document,” a word-for-word copy of Hendry’s abstract. *Id.*, at 154.

Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

90. None of the other prior art relied on in this Petition was cited during prosecution of the '532 patent, and the Patent Office never considered the prior art combinations or arguments in Grounds 1-6, below.

C. The Challenged Claims

91. This Declaration addresses Claims 1-4, 6-14, and 16-20 of the '532 patent. Claims 1 and 11 are the independent claims, and are reproduced below.

92. Independent Claim 1 recites:

1[pre]	1. A computer-implemented method of programmatically assigning a descriptive attribute to an untagged element on a web page to enable an audible description of the untagged element, the web page having an associated document object model (DOM), the method comprising:
1[a]	dynamically analyzing, by a computer system, code associated with the web page, the code comprising at least the DOM or HTML code;
1[b]	detecting, by the computer system, one or more compliance issues relating to web accessibility standards in the code, wherein at least one of the one or more compliance issues comprises the untagged element lacking an adequate descriptive attribute;
1[c][1]	applying, by the computer system, one or more pre-existing remediations to the one or more compliance issues, wherein the one

Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

	or more pre-existing remediations is generated by a remote server system performing at least:
1[c][2]	accessing the code associated with the web page;
1[c][3]	receiving, by the remote server system, via a remediation interface input from a user, the input comprising a remediation action to manually remediate a non-programmatically-fixable compliance issue associated with the web page;
1[c][4]	generating, by the remote server system, the one or more pre-existing remediations based on the input comprising the remediation action to manually remediate a non-programmatically-fixable compliance issue;
1[c][5]	storing, by the remote server system, the one or more pre-existing remediations in an electronic data storage medium; and
1[d]	assigning, by the computer system, a descriptive attribute to the untagged element in the code of the web page based on the one or more pre-existing remediations, the descriptive attribute assigned to the untagged element adapted to enable an assistive technology to speak the descriptive attribute to a user.

93. Independent Claim 11 recites:

11[pre]	11. A non-transitory electronic storage medium with computer code stored thereon, the computer code configured to programmatically assign a descriptive attribute to an element on a web page to enable an
----------------	--

Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

	audible description of the element, the web page having an associated document object model (DOM), and the computer code configured to perform, when executed by a computer processor, the steps of:
11[a]	dynamically analyzing, by a computer system, code associated with the web page, the code comprising at least the DOM or HTML code;
11[b]	detecting, by the computer system, one or more compliance issues relating to web accessibility standards in the code, wherein at least one of the one or more compliance issues comprises the element lacking an adequate descriptive attribute;
11[c][1]	applying, by the computer system, one or more pre-existing remediations to the one or more compliance issues, wherein the one or more pre-existing remediations is generated by a remote server system performing at least:
11[c][2]	accessing the code associated with the web page;
11[c][3]	receiving, by the remote server system, via a remediation interface input from a user, the input comprising a remediation action to manually remediate a non-programmatically-fixable compliance issue associated with the web page;
11[c][4]	generating, by the remote server system, the one or more pre-existing remediations based on the input comprising the remediation action to manually remediate a non-programmatically-fixable compliance issue;

Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

11[c][5]	storing, by the remote server system, the one or more pre-existing remediations in an electronic data storage medium; and
11[d]	assigning, by the computer system, a descriptive attribute to the element in the code of the web page based on the one or more pre-existing remediations, the descriptive attribute assigned to the element adapted to enable an assistive technology to speak the descriptive attribute to a user.

94. I have added indices of the form 1[a], 1[b], 1[c], *etc.* to each of the claim limitations for ease of reference, and to match the indices used in the Petition.

VI. CLAIM CONSTRUCTION

95. On November 23, 2021, the Court provided a claim construction order in a district court litigation styled *AudioEye, Inc. v. accessiBe Ltd.*, Case No. 6:20-cv-00997-ADA (W.D. Tex.) (*accessiBe I*). Ex. 1024 (Court’s Amended Claim Construction Order). The Court construed certain claim terms in connection with the following patents which are all related to the ’532 patent at issue in this IPR proceeding: 10,867,120 (“’120 patent”), 10,444,934 (“’934 patent”), 10,866,691 (“’691 patent”), 10,845,946 (“’946 patent”), 10,845,947 (“’947 patent”), 10,809,877 (“’877 patent”), and 10,860,173 (“’173 patent”).

Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

96. I understand that the '532 patent at issue in this IPR was asserted by AudioEye against accessiBe in a litigation styled, *AudioEye, Inc. v. accessiBe Ltd.*, Case No. 6:21-cv-726-ADA (W.D. Tex.) ("*accessiBe I*"). I understand that the parties agreed that the district court's claim construction in *accessiBe I* would be applicable to the '532 patent and agreed "alt text label" would be construed as "alternative text label" and that "the [untagged] element lacking an adequate descriptive HTML attribute" in claims 1 and 11 of the '532 Patent is indefinite under the Court's claim construction order in *accessiBe I* rendering all claims besides claims 4 and 14 indefinite. Ex. 1024.

97. I understand that the question of indefiniteness is not before the Board in this IPR proceeding and that despite the district court's indefinites ruling, the Board may also assess the validity of the claims under §§ 102 and 103. Accordingly, I performed my analysis of the claims at issue in view of the Court's claim constructions in *accessiBe I* and the parties' subsequent agreement regarding their applicability to the claim terms at issue in the '532 patent in forming my opinions below.

98. For the terms not construed by the Court, I have afforded claim terms their plain and ordinary meaning as viewed by a POSITA reading the '532 patent.

VII. APPLICATION OF THE PRIOR ART TO ASSERTED CLAIMS

99. I have reviewed and analyzed the prior art references and materials listed in Part I.B. above. In my opinion, the claims of the '532 patent are anticipated or rendered unpatentable as obvious based on the following prior art:

Ground	Claims	References and Type of Challenge
1	1-4, 6-14, 16-20	Springer [Ex. 1005] in view of Takagi [Ex. 1007] (35 U.S.C. § 103(a))
2	1-4, 6-14, 16-20	Springer [Ex. 1005] and Takagi [Ex. 1007] in view of Hendry [Ex. 1009] (35 U.S.C. § 103(a))
3	1-4, 6-14, 16-20	Springer [Ex. 1005] and Bigham [Ex. 1008] in view of Hendry [Ex. 1009] (35 U.S.C. § 103(a))
4	1-4, 6-14, 16-20	Markel [Ex. 1006] in view of Takagi [Ex. 1007] (35 U.S.C. § 103(a))
5	1-4, 6-14, 16-20	Markel [Ex. 1006] and Takagi [Ex. 1007] in view of Hendry [Ex. 1009] (35 U.S.C. § 103(a))
6	1-4, 6-14, 16-20	Markel [Ex. 1006] and Bigham [Ex. 1008] in view of Hendry [Ex. 1009] (35 U.S.C. § 103(a))

100. The six grounds presented in this Petition rely primarily on Springer (Ex. 1005), Markel (Ex. 1006), Takagi (Ex. 1007), Bigham (Ex. 1008) and Hendry (Ex. 1009).

101. Springer (Ex. 1005) describes a series of tools that check for, and then fix, accessibility compliance issues for web pages. These are referred to by Springer as “checkers” and “fixers.” Each checker tool in Springer looks for a

Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

specific accessibility compliance issue in the website code and then sends identified compliance issues to a corresponding fixer tool.

102. Markel (Ex. 1006) discloses a system and method for evaluating and correcting websites by “identif[ying] design errors and standard violations including... missing HTML tags” (*id.*, at ¶72), and generating a rule “which specifies that an image tag having an alternative form is required” (*id.*, at ¶73). Corrections can be made automatically, semi-automatically, or manually. *Id.*, at Abstract.

103. Takagi (Ex. 1007) discloses an open community for improving accessibility to a webpage called “social accessibility pilot system.” Takagi’s social accessibility pilot system is characterized by its use of external metadata and the participation of an open community, where it allows for improving the accessibility of any webpage without asking the site owners to modify the website themselves. Ex. 1007, at Abstract.

104. Bigham (Ex. 1008) a system that automatically creates and inserts alternative text into web pages. To formulate alternative text for images, Bigham presents three labeling modules based on web con-text analysis, enhanced optical character recognition (OCR) and human labeling. Bigham’s system, referred to as “WebInSight” caches alternative text in a local database and can add new labels

Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

seamlessly after a web page is downloaded, resulting in minimal impact to the browsing experience. Ex. 1008, at Abstract.

105. Hendry (Ex. 1009) discloses a remediation system and method using a distributed computing system where certain steps (e.g., determining the appropriate descriptive attribute) occurs on one computer system while other steps (assigning the attribute to the input field) occurs on another local (e.g., user) computer system. Ex. 1009, at ¶¶52-58.

106. I am informed by counsel that each reference listed above qualifies as prior art to the challenged claims because each reference was filed and/or published before the earliest claimed priority date of the '532 patent. I reserve the right to respond in the future to any arguments or positions that the Patent Owner may raise, taking account of new information as it becomes available to me.

A. Brief Summary of Prior Art

1. References Cited in Grounds 1-6

a. Springer [Ex. 1005]

107. U.S. Publ. No. 2004/0148568A1 (“Springer,” Ex. 1005) is titled “Checker and fixer algorithms for accessibility standards” and was published on July 29, 2004. I am informed that Springer qualifies as prior art under at least (post-AIA) 35 U.S.C. § 102(a) because it was published more than one year before the earliest priority date for the '532 patent.

Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

108. Springer teaches using a set of tools that scan for, and then fix, accessibility issues in web pages. Ex. 1005, at Abstract. The Springer system accesses the code (e.g., HTML) or data model (e.g., DOM) of a website, runs a series of checker tool to detect accessibility compliance issues; and applies fixer tools (each corresponding to a specific checker tool) to remediate the code. This is described, for example, at a high level as follows:

Aspects of the invention can include one or more of the following advantages. A compliance retrofitter in accordance with the invention can be used to scan the HyperText Mark-up Language (HTML) code that powers the web, identify violations with Section 508 Standards (checkers), and correct those violations by inserting the necessary corrections into the code in real time (fixers). Incorporation into the system of “tolerances” allows the checkers to be customized to individual clients and greatly enhances the value of such tools when distributed on an organization-wide level.

Id., at ¶10; *see also id.*, at ¶7 (“The method may include, and the computer program product may implement, the steps of running a checker against an HTML document; flagging a violation of a requirement of Section 508; and fixing a section of the HTML document containing the flagged violation by modifying HTML code.”).⁷

⁷ Springer describes Section 508 Standards as “standards for web accessibility, including web accessibility for assistive technologies” that are “are based in part

Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

109. Each of Springer’s “checkers” checks the data model of a web page (e.g., the DOM) (*see id.*, at Abstract) and/or scans the HTML code (*see* Ex. 1005, at ¶10) and identifies a type of accessibility compliance issue in the website code. The type of accessibility compliance issue can include untagged elements lacking meaning textual descriptions. In one example, a type of untagged element that Springer focuses on remediating is untagged form fields. Springer discloses that “[f]orm fields are a way for users to enter input into a web page, usually by filling in text boxes or selecting from drop-down menus. Form fields are HTML elements that allow data entry into a form. A common form field is the INPUT element.” Ex. 1005, ¶37. In order to detect and remediate unlabeled form fields, Springer discloses “[a] form label checker may be used to identify a form field in a document that does not have a label explicitly associated with it. A form label fixer may then be used to find a piece of text in the HTML page that is most likely the intended label for the form element.” *Id.*, ¶38. Furthermore, Springer’s FormChecker “[c]hecks that no form controls are missing labels” and “[c]hecks that labels are positioned properly.” *Id.*, at Tables I(a)-(b).

on the World Wide Web Consortiums' (W3C) Web Accessibility Initiative's (WAI) Web Content Accessibility Guidelines 1.0 (WCAG 1.0).” Ex. 1005 at ¶5. Springer also discloses that the same methods can be applied to any accessibility standards and are not limited to then-existing Section 508. *Id.*, at ¶58.

Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

TABLE I(a)-continued

<u>36 CFR § 1194.22 Checkers</u>			
36 CFR § 1194.22 Paragraph	Description	Checker	Manual Review
L	into compliance with the other requirements in § 1194.22. Paragraph (l) requires that when web pages rely on special programming instructions called “scripts” to affect information displayed or to process user input, functional text shall be provided. Should have textual alternative.	NoScriptChecker AnchorChecker	If site uses heavy javascript, ensure it is readable with assistive technology.
M	This provision requires that web pages which provide content such as Real Audio or PDF files, also provide a link to a plug-in that will meet the software provisions.	PlugInChecker	No Manual Review Required
N	Paragraph (n) requires that people with disabilities have access to interactive electronic forms.	FormChecker	No Manual Review Required

Ex. 1005, at Table I(a) (excerpt, annotated).

Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

TABLE I(b)-continued

		<u>Checker Functionality</u>
36 CFR § 1194.22 Paragraph	Checker	Functionality
		user to check that a colorless version of the page would be accessible.
D	StyleChecker	Diagnoses style sheet dependent elements as candidate violations. Enabling "Manual Review" prompts user to verify that the supplied previews of elements without style sheets are accessible.
E	SSIMAPChecker	Diagnoses server-side image maps as candidate violations. Enabling "Manual Review" prompts user to check that textual alternatives are in place.
F	ImageMapChecker	Checks that all AREAs have valid alt attributes.
G & H	TableChecker	Checks that THEAD and TBODY are used properly. Checks that cells are properly associated with a table header. Checks that a table has either a caption element or title attribute. Checks that a table has a summary. Checks that a table header has an "abbr" attribute if the enclosed text is longer than one word. Checks that a table defines the dir attribute.
I	FrameChecker	Checks that valid NOFRAMES elements are used.
	FrameTextChecker	Checks that a FRAME has a valid title attribute.
J	BlinkChecker	Checks to see if BLINK/MARQUEE elements are used.
	CSSChecker	Checks to see if BLINK/MARQUEE css attributes are used.
L	NOSCRIPTChecker	Checks that valid NOSCRIPT elements are used.
	AnchorChecker	Checks that links do not directly target javascript.
M	PlugInChecker	Checks that all embedded files have links to download plug-in.
N	FormChecker	Checks that OPTIONS are grouped by OPTGROUP tags. Checks that no form controls are missing labels. Checks that labels are positioned properly. Checks that labels are explicitly associated with form controls. Checks that FORMs are associated with FIELDSET groups. Checks that a FIELDSET group contains a LEGEND. Checks that one OPTION form control is selected by default.

Ex. 1005, at Table I(b) (excerpt, annotated).

Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

110. In another example, Springer’s “ImageChecker” tool can be used to check “that images have valid alt attributes” and “that long descriptions are valid.” *See* Ex. 1005, at ¶10, Tables I(a)-(b). Springer also includes an ImageMapChecker that checks that AREAs have valid alt attributes. *Id.*; *see also id.*, at ¶15 (“As noted in the sixth row of Table I(b), ImageMapChecker, in part, checked the HTML code of www.yahoo.com to determine if all of the AREAS had valid “alt attributes,” i.e., to determine if each AREA had associated with it an alternative textual description. (An AREA element defines a part of an image that functions as a link. Since some users do not use browsers that display images, having valid alt attributes for AREAs allows these users to navigate through the page.)”).

TABLE I(a)			
36 CFR § 1194.22 Checkers			
36 CFR § 1194.22 Paragraph	Description	Checker	Manual Review
A	Paragraph (a) requires that a text equivalent for every non-text element shall be provided. The provision is necessary because assistive technology cannot describe pictures, but can convey the text information to the user.	ImageChecker ObjectTextChecker ImageMapChecker AsciiArtChecker AppletChecker	No Manual Review Required

Ex. 1005, at Table I(a) (excerpt, annotated).

TABLE I(b)		
<u>Checker Functionality</u>		
36 CFR § 1194.22 Paragraph	Checker	Functionality
A	ImageChecker	Checks that images have valid alt attributes. Checks that image links have valid alt attributes. Checks that image buttons have valid alt attributes.
	ObjectTextChecker	Checks that long descriptions are valid. Checks that OBJECT elements have enclosed textual descriptions.
	ImageMapChecker	Checks that AREAs have valid alt attributes.
	AppletChecker	Checks for valid textual alternatives in APPLET elements.

Ex. 1005, at Table I(b) (excerpt, annotated).

111. Springer discloses “fixers” corresponding to each “checker.” When a specific checker finds a compliance issue, it selects that issue to the corresponding fixer, and that specific fixer is used for remediation. As described above, the Springer discloses a FormFixer tool which “[a]dds label elements to forms” and “[e]xplicitly associates a LABEL element with the form control.” *See* Ex. 1005, at ¶10, Tables II(a)-(b). Referring to the same “ImageChecker” and “ImageMapChecker” examples discussed above, Springer discloses a corresponding “ImageFixer” and “ImageMapFixer.” *See id.*, at Tables II(a)-(b). ImageFixer and ImageMapFixer remediate compliance issues stemming from websites lacking a text equivalent for every non-text element such as an image. *Id.*

Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

ImageFixer remediates this issue, for example, by adding alt attributes to images, image buttons, and image links, as well as replacing alt attributes with more appropriate (e.g., in length) or meaningful descriptions. *Id.* ImageMapFixer does the same things for AREAS (described above). *Id.* Springer does not limit how alt attributes for images are determined, thus leaving the system open to numerous means for determining an appropriate alt text for untagged images. Furthermore, Springer’s “ImageFixer” tool can be used to “[add] alt attributes to images” that were previously determined to be non-compliant. *Id.*, at Table II(b). Springer also teaches that its ImageChecker and ImageFixer tools require *no* manual input. *Id.*, at Tables I(a)-II(b).

TABLE II(a)			
<u>36 CFR § 1194.22 Fixers</u>			
36 CFR § 1194.22 Paragraph	Description	Fixers	Manual Fixes
A	Paragraph (a) requires that a text equivalent for every non-text element shall be provided. The provision is necessary because assistive technology cannot describe pictures, but can convey the text	ImageFixer ObjectTextFixer ImageMapFixer AsciiArtFixer AppletFixer	None

Ex. 1005, at Table II(a) (excerpt, annotated).

Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

TABLE II(a)-continued

36 CFR § 1194.22 Fixers			
36 CFR § 1194.22 Paragraph	Description	Fixers	Manual Fixes
	information to the user. This provision requires that whenever an image is used an appropriate alt attribute must be defined.		
B	Paragraph (b) provides that equivalent alternatives for any multimedia presentation shall be synchronized with the presentation.	n/a	Add equivalent alternatives for any multimedia presentation.
C	Paragraph (c) prohibits the use of color as the single method for indicating important information on a web page.	ColorContrastFixer	Add alternatives for any information that requires color.
D	Paragraph (d) provides that documents must be organized so they are readable without requiring browser support for style sheets.	n/a	Ensure that style sheets are not required for functionality.
E	Paragraph (e) requires web page designers to include redundant text links for each active region of a server-side image map on their web pages.	ImageMapFixer	If a server-side image map is used, insert links with textual alternatives into page.
F	Paragraph (f) provides that client-side image maps shall be provided instead of server-side image maps except where the regions cannot be defined with an available geometric shape.	ImageMapFixer	None
G & H	Paragraphs (g) and (h) permit the use of tables, but require that the tables be coded according to the rules for developing tables of the markup language used.	TableFixer	None
I	Paragraph (i) addresses the use of frames and requires that they be titled with text to identify the frame and assist in navigating the frames.	FrameFixer FrameTextFixer	None
J	Paragraph (j) sets limits on the blink or flicker rate of screen elements.	BlinkFixer CSSFixer	None
K	Paragraph (k) requires that a text-only web page shall only be provided as a last resort method for bringing a web site into compliance with the other requirements in §1194.22.	n/a	None
L	Paragraph (l) requires that when web pages rely on special programming instructions called "scripts" to affect information displayed or to process user input, functional text shall be provided. Should have textual alternative.	NoScriptFixer AnchorFixer ManualScriptFixer	If page cannot be used without scripts, create alternate page.
M	This provision requires that web pages which provide content such as Real Audio or PDF files, also provide a link to a plug-in that will meet the software provisions.	PluginFixer	None
N	Paragraph (n) requires that people with disabilities have access to interactive electronic forms.	FormFixer	None

Ex. 1005, at Table II(a) (excerpt, annotated).

Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

TABLE II(b)

		<u>Fixer Functionality</u>
36 CFR § 1194.22		
Paragraph	Fixer	Functionality
A	ImageFixer	Adds alt attributes to images
		Adds alt attributes to image buttons
		Adds alt attributes to image links
		Replaces alt attributes that are too long with shorter ones
		Replaces vague alt attributes on images with more meaningful ones
		Replaces empty or invalid longdesc descriptions with valid ones
		Replaces vague alt attributes on image buttons with more meaningful ones
		ObjectTextFixer Adds a valid textual description to OBJECT elements
		AppletFixer Adds textual alternatives to APPLETT elements
C	ColorContrastFixer	Allows developer to select a new color which has sufficient color contrast
F	ImageMapFixer	Adds alt attributes to AREAs
		Replaces vague alt attributes in AREAs with more meaningful ones
G & H	TableFixer	Inserts a caption element or title attribute to a table
		Enters a single word abbreviation for a table header
		Enters a dir attribute for a table
		Associates cells with a table header
		Enters a summary attribute for the table
I	FrameFixer	Adds a NOFRAMES tag to the page
		FrameTextFixer Adds a valid title attribute to a FRAME
J	BlinkFixer	Replaces BLINK elements with other forms of emphasis
		CSSFixer Replaces BLINK/MARQUEE CSS elements with other forms of emphasis
L	NOSCRIPTFixer	Adds a NOSCRIPT element
		AnchorFixer Modifies anchors so they do not directly target javascript
M	PlugInFixer	Adds a link to download identified media types
N	FormFixer	Adds OPTGROUP tags to element groups
		Adds LABEL elements to forms
		Explicitly associates a LABEL element with the form control
		Associates a FIELDSET grouping with a form
		Adds a LEGEND to a FIELDSET
		Adds a "selected" attribute to one OPTION element in each SELECT form control

Ex. 1005, at Table II(b) (excerpt, annotated).

Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

112. Springer's fixers, such as the FormFixer, ImageFixer and ImageMapFixer, can be performed without manual input from an end user (see Table II(a), above) to automatically insert the remediation into the code (e.g., HTML or document model). *Id.*, at ¶¶38, 56.

113. A person of ordinary skill in the art would have recognized that Springer was capable of checking and applying fixes both to HTML and DOMs associated with websites. DOMs were both well known in the art and described in the '532 patent itself as tree-like representations of a webpage. For example, the '532 patent states:

More particularly, web pages and other electronic documents accessible online (or otherwise accessible through a graphical user interface (GUI)) typically have an established reading order embodied in the DOM, including visual elements that are tagged with alternative text descriptions. The nodes of the document are organized in a tree structure, called the DOM tree. When an HTML page is rendered, the browser downloads the HTML into local memory and automatically parses it before displaying the page.

The W3C Document Object Model (DOM), a W3C (World Wide Web Consortium) standard for accessing documents, is a platform and language-neutral interface that allows programs and scripts (such as JavaScript) to dynamically access and update the content, structure, and style of a document. The HTML DOM is a standard object model and programming interface for HTML, and defines the properties of all HTML elements and the methods to access the HTML elements. In other words, the HTML

Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

DOM is a standard for how to get, change, add, delete, or otherwise interact with HTML elements.

Ex. 1001, at 4:5-24.

114. A DOM is a standard way, and indeed the most common way, to represent a logical tree structure for HTML pages. It is also the most common way to visualize HTML code, and the most common format to use when modifying website HTML code with other software. Ex. 1014, at 2 (W3C DOM v1.0); *see also* Ex. 1014, at 11 (W3C DOM v1.0) (“In the DOM, documents have a logical structure which is very much like a tree.”); *id.* (describing the DOM as a “tree-like representation of a document [e.g., website].”); *id.* (“The name ‘Document Object Model’ was chosen because it is an ‘object model’ in the traditional object oriented design sense”); *id.*, at 12 (“The DOM originated as a specification to allow JavaScript scripts and Java programs to be portable among Web browsers.”).

115. Springer states that “[t]he checkers implemented by the compliance retrofitter are sets of tests that are run against documents, for example files or web pages, to analyze compliance with a standard, e.g., Section 508.” Ex. 1005, at ¶14. Springer also discloses that it “check[s] a data model of a web page” (e.g., DOM) (*id.*, at Abstract), and that it interacts with webpages through a “document tree” or “tree format.” *Id.*, at ¶¶25, 38, 49. Springer also teaches that its software can be written using Java or other platform independent language, and is designed to

Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

interface with various web development tools, such as FrontPage, DreamWeaver, and GoLive web page development programs. *Id.*, at ¶57. Java is understood to be an object-oriented programming language. Because a Java program would manipulate a hierarchical document through a Java object model, Springer’s “document tree” is a type of document object model. From at least the above disclosures, a person of ordinary skill in the art would have understood that Springer can be used to check and fix Document Object Models (DOM) associated with a webpage as well as the HTML code itself. *See also id.*, at ¶7.

116. Springer also discloses manual remediations in instances where the compliance retrofitter cannot automatically fix a compliance issue. The user within the application will be prompted to enter a “manual re-coding.” *Id.*, at ¶56. The manual changes made by the user can be saved to an “Autofix library” which allows a user to “build up a store of fixes to be applied to other documents” and which can be referenced by the compliance retrofitter. *Id.*, at ¶18. The “Autofix Library” includes a “button” which, when selected, “will apply the corrections contained in the Autofix library to the current violation.” A user can also select the button to fix multiple pages.

Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

b. Markel [Ex. 1006]

117. U.S. Publ, No. 2002/0156799A1 (“Markel,” Ex. 1006) entitled “System and Method for Verifying and Correcting Websites,” (hereinafter “Markel”) was filed in April 24, 2002 and published on October 24, 2002.

118. I am informed that Markel qualifies as prior art under at least (post-AIA) 35 U.S.C. § 102(a) because it was published more than one year before the earliest priority date for the ’532 patent.

119. Markel had disclosed a system and method for evaluating and correcting websites automatically, semi-automatically, or manually. Ex. 1006, at Abstract. Markel’s tool allows for “diagnos[ing], evaluat[ing], report[ing], and retrofit[ting] code violations existing in websites” (*id.*, at ¶52), by “identif[ying] design errors and standard violations including... missing HTML tags” (*id.*, at ¶72), and generating a rule “which specifies that an image tag having an alternative form is required” (*id.*, at ¶73).

120. A “web crawler or spider” requests “the information located at the URL entered by the user... the information sent represents a blueprint of the website requested and includes code such as HTML, Java Script, or any other suitable programming language.” *Id.*, at ¶68. An assessment module “identifies design errors” including image tags which require an “alternative form.” *Id.*, at

Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

¶¶72-73. A reporting module then categorizes the type of correction item required including “automatic correction items, semi-automatic assisted correction items, [and] manual correction instruction item.” *Id.*, at ¶77.

121. For items that must be semi-automatically corrected, the user is prompted to enter an appropriate label. For instance, where the website is lacking an HTML tag, such as an ALT tag, “the user is prompted to enter the pertinent information to allow the system to replace the ALT tag.” *Id.*, at ¶83. The user then selects the “Fix button” and the code is corrected. *Id.*, at ¶114. The system also directs the user to areas of the code that may require manual correction. *Id.*, at ¶115.

122. The correction module’s primary function is “automated execution or semi-automated wizard-assisted implementation and support for manual correction of reported ruleset violations.” *Id.*, at ¶78.

c. Takagi [Ex. 1007]

123. “Collaborative Web Accessibility Improvement: Challenges and Possibilities” by Hironobu Takagi et al. (“Takagi,” Ex. 1007) was published in October 2009. I am informed that Takagi qualifies as prior art under at least (post-AIA) 35 U.S.C. § 102(a) because it was published more than one year before the earliest priority date for the ’532 patent. *See* Ex. 1021 (Hall-Ellis Declaration).

Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

124. Takagi discloses “social accessibility” as a method for improving the accessibility of web sites. Ex.1007, 196. This allows end users to report accessibility issues to a support community, and in turn, the support community creates external metadata, including alternative text labels, to be provided to the end user to fix the issue. *Id.* “ As a result, the end user can browse the webpage made accessible with the metadata. . . . The public repository allows that metadata to benefit every end user, not just the first user who reported the problem. *Id.* Takagi describes the social accessibility method including a social accessibility pilot system characterized by its use of external metadata and the participation of an open community, where it allows for improving the accessibility of any webpage without asking the site owners to modify the website themselves. Ex. 1007, at 196, col. 1 ¶6. Takagi describes that the social accessibility pilot system consists of the three components (1) end-user tools for problem reporting and client-side webpage transcoding based on external metadata; (2) volunteer tools for metadata authoring; and (3) a public server as the metadata repository. *Id.* Figure 1 of Takagi shows a social accessibility server as part of the social accessibility service (Ex. 1007, Figure 1):

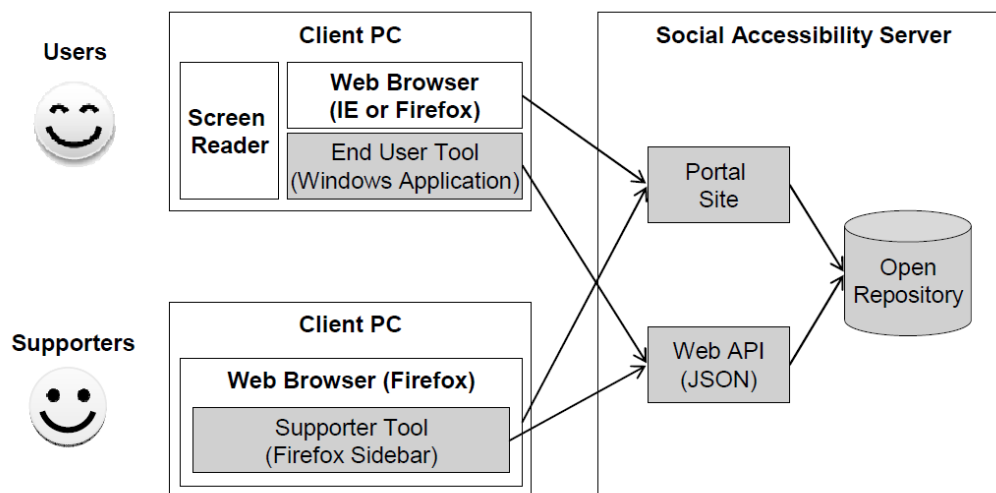


Figure 1. Basic Architecture of Social Accessibility Service

Ex. 1007, at Figure 1.

125. Furthermore, Takagi discloses that “[b]oth end users and volunteers (also called supporters) access the repository using client-side tools, which interact with the server through Web APIs.” Ex. 1007, at 196, col. 1 ¶6. Takagi also describes that the end users and supporters “can also access the repository by visiting the portal site with their Web browsers.” *Id.* Takagi further discloses that both end users and supports “can also access the repository by visiting the portal site with their Web browsers” as shown in Figure 2 of Takagi. Ex. 1007, at 196, col. 1 ¶6, Figure 2. Figure 2 of Takagi also shows the social accessibility server including the metadata repository (Ex. 1007, at Figure 2), shown below:

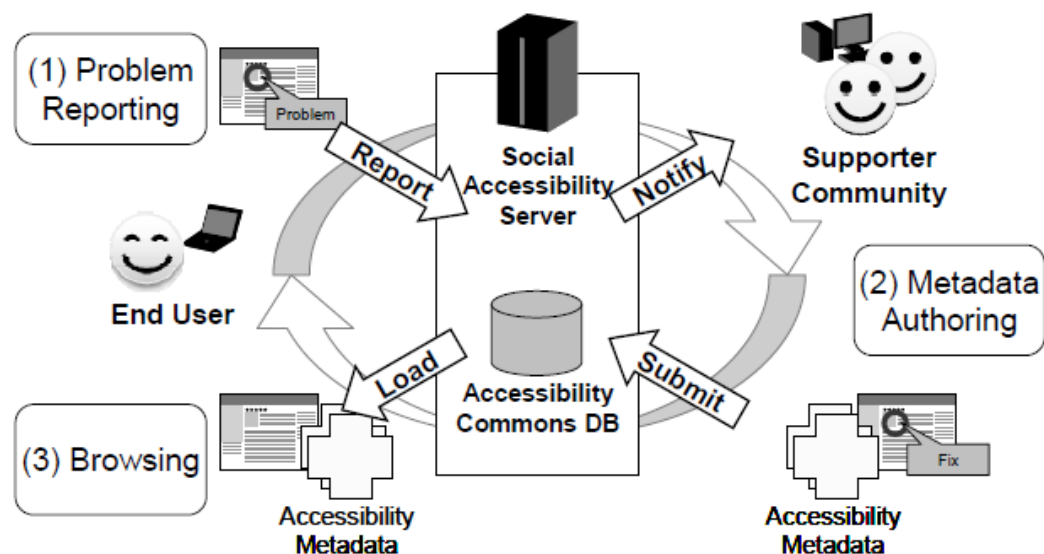


Figure 2. Current Process Model

Ex. 1007, at Figure 2.

126. Additionally Takagi describes the process for social accessibility service based accessibility improvement:

First, an end user reports an accessibility problem on a webpage. This report goes to the supporter community instead of to the site owner (as with a contact form). Then a supporter creates external metadata to fix the reported problem. As a result, the end user can browse the webpage made accessible with the metadata. This reporting-based strategy allows quick solutions to accessibility problems, while reducing the supporters' work, and focuses on problems that end-users actually encounter.

Ex. 1007, at 196, col. 1 ¶6.

127. Takagi further discloses that “the public repository allows that metadata to benefit every end user, not just the first user who reported the

Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

problem.” *Id.* Takagi also recognized that social labeling was by no means an innovation and it had already been proposed by many other references, including the 1997 ALT-server, WebInSight (Bigham), and WebVisum. Ex. 1007 at 196, col. 2, ¶5.

d. Bigham [Ex. 1008]

128. “WebInSight: Making Web Images Accessible” by Jeffrey P. Bigham (“Bigham,” Ex. 1008) was published as early as October 2006, and in any case no later than January 2007, well before the ’532 patent’s earliest priority date. I am informed that Bigham qualifies as prior art under at least (post-AIA) 35 U.S.C. § 102(a) because it was published more than one year before the earliest priority date for the ’532 patent. *See* Ex. 1021 (Hall-Ellis Declaration).

129. Bigham discloses software for automatically generating and adding alternative text for images in websites and for measuring the quality of alternative text of web elements by considering context surrounding those elements. Ex. 1008 at 181, Abstract, 182 col. 2, ¶¶1, 5, 184 col. 2, ¶2. Like the references discussed above, Bigham also recognized that “[m]uch previous work exists on the difficult task of deriving textual labels for arbitrary images.” *Id.*, at 182, col. 2, ¶5.

130. The WebInSight system disclosed by Bigham “retrieves alternative text from its database for each image when it is present and requests that

Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

alternative text be calculated for any image not already in its database.” *Id.*, at 184, col. 2, ¶2. Bigham also describes automatically generating potential alternative text and as well as using human labeling services to generate descriptive text. *Id.*

131. Bigham also describes a combination of “image labeling modules that provide a mechanism for labeling arbitrary web images,” including web context labeling where the system “retrieves the contents of pages linked to by images within anchor tags and uses this text to formulate alternative text,” OCR image labeling “to accurately capture text from many web images,” and human labeling where “users request that an image be sent to a labeling service for labeling by humans.” *Id.*, at 185 col. 2, ¶¶2, 186 col. 1, ¶2, col. 2, ¶3.

132. Further, Bigham recognized that others in the field were working on using “determining an accurate textual label for an arbitrary image using computer vision techniques,” and that such techniques were still “generally unsolved” and that such techniques could be applied to automatically determine appropriate textual label of images. *Id.*, at 182, col. 2, ¶2.

133. Bigham also recognized the utility of human labeling, and noted how it had already been harnessed by computer games and photo sharing sites to generate keywords for images and/or photos, allowing these captions to be used as alternative text. *Id.*, at 183, col. 1, ¶1.

Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

134. Bigham also acknowledged that WebInSight “currently implements a simple web-based application for humans to label images.” *Id.*, at 186, col. 2, ¶4. When a web user prefers additional text, the user clicks a link following the unlabeled image, and the image is sent to a labeling service for manual labeling. *Id.*, at 186, col. 2, ¶¶4, 5.

e. Hendry [Ex. 1009]

135. U.S. Publ. No. 2013/0104029 (“Hendry,” Ex. 1009), titled “Automated Addition of Accessibility Features to Documents” was published on April 25, 2013. Ex. 1009. I am informed that Hendry qualifies as prior art under at least (post-AIA) 35 U.S.C. § 102(a) because it was filed more than one year before the earliest priority date for the ’532 patent.

136. Hendry, like Springer, discloses a system for automatically remediating inaccessible web pages. Specifically, Hendry discloses methods for “automatically modifying documents to improve document accessibility.” *Id.*, at ¶12.

137. Hendry defines many of its terms. For example, as used in Hendry, “[d]ocuments (114) include, but are not limited to, pdfs, web pages, word processing documents, spread sheet documents, slides, images, video files, etc.” *Id.*, at ¶42. Hendry explains that “[m]odifying a document to improve document

Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

accessibility refers to modifications which result in proper or better interpretation by an assistive technology, or which result in improving a user's ability to understand the content of a document." *Id.*, at ¶12.

138. Hendry defines "accessibility features" as "particular tags which, when added to a document, allow for proper interpretation of the document by assistive technologies. For example, the accessibility features (118) correspond to formatting, styles, or other markup that can be applied to one or more elements within a document." *Id.*, at ¶21. One example of accessibility features disclosed by Hendry include "text alternatives for non-text content," such as descriptive text for unlabeled input fields. *Id.*, at ¶24. Hendry teaches that the accessibility features and/or completely remediated documents (with accessibility features added to specific elements) are then stored separately in a data repository for future use. *Id.*, at ¶¶21, 41.

139. Hendry is relied on for its disclosure of distributed computing for accessibility remediation systems and methods. For example, Hendry teaches that its methods can be performed on: (1) a local client computer, (2) a remote server, or (3) a combination of the two. Hendry also teaches that its "[e]xamples [that] refer to a specific machine performing the steps [are merely] for the purposes of clarity and understanding" and that any step can be performed on any machine.

Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

Id., at ¶59. It even teaches that the various steps recited can be repeated, omitted, or performed in different order. *Id.*

140. In one specific embodiment, Hendry discloses a method where certain steps are performed on a local computer while other are performed by a remote server, where the two are in communication. *Id.*, at ¶58. This key embodiment involves the steps of: (1) a client machine sending a remediation request for a document to a remote server; (2) the remote server accessing the relevant document and adding JavaScript remediation code for remediating the document (i.e., the code will remediate the document when it is executed); (3) the server sending the document and remediation code back to the client machine; and (4) the client machine executing the remediation code that was generated by the remote server, to modify the document by adding accessibility features. *Id.*

141. Figure 1 of Hendry, shown below, is a block diagram which illustrates an exemplary system for automatically producing documents to improve document accessibility using components across one or more clients and one or more servers. As noted above, the various steps and elements may be performed on either the local or remote computer/server, or in any combination of the two.

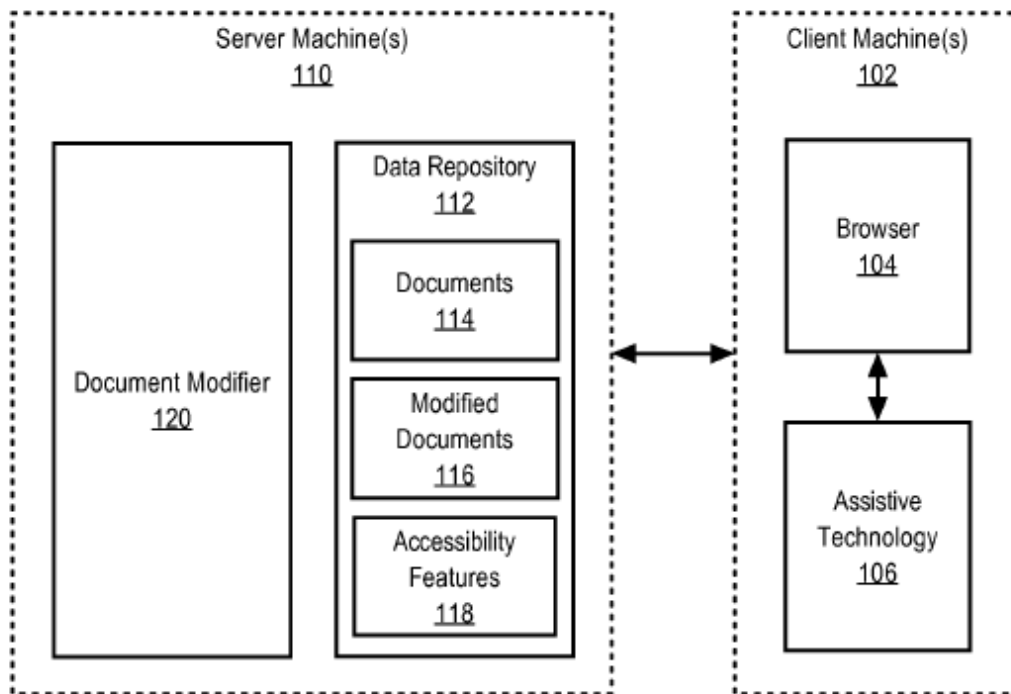


FIG. 1

142. As shown above in Figure 1, excerpted above, an embodiment of the Hendry system (100) includes a client machine (102) (i.e., local computer) and a server machine (110) (i.e., a remote computer). *Id.*, at ¶34. Other embodiments might have different configurations, as Hendry makes clear that “[d]ata or components described with reference to a particular machine may be stored on or implemented by another machine. Additional devices and/or components not described herein may be used to perform any of the functionality described herein.” *Id.* Hendry also states that that “[i]n one or more embodiments, the example system (100) is accessible over a network connection (not shown), such as

the internet by one or more users. Information and/or services provided by the example system (100) may also be stored and accessed over the network connection.” *Id.*, at ¶48.

143. Hendry also discloses providing the automatically modified document “to a user for manual changes, verifications, and corrections.” *Id.*, at ¶85.

B. GROUND 1: Claims 1-4, 6-14, 16-20 are Unpatentable as Obvious Over Springer and Takagi.

1. Motivation to Combine

144. A POSITA would have been motivated to combine Springer and Takagi, which both disclose checking for and remediating accessibility issues in web pages. A POSITA would have been motivated to combine Springer and Takagi in order to improve on Springer’s system for checking and remediating accessibility issues by adding the metadata authoring and sharing taught by Takagi. A POSITA would also have been motivated to combine Springer and Takagi in order to improve upon Springer’s compliance retrofitter functions for determining and assigning descriptive attributes for untagged elements, e.g., such as labels for untagged input fields and images lacking alt-text. Springer provides the various tools for checking and remediating web pages, but does not take advantage of collaborative metadata authoring and the powerful functions of a remote server that were commonplace by the time of the ’532 patent.

Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

145. As I described above, Springer describes tools for remediating untagged elements such as unlabeled form fields and elements lacking a text equivalent for a non-text element (e.g., an image or an AREA). Specifically Springer discloses a form label checker, form label fixer, image checker, and image fixer. Ex. 1005, at Tables I(a)-(b), II(a)-(b). The form label checker finds inaccessible elements, identifying form fields (e.g., untagged input fields) in a website code that do not have a label (e.g., descriptive attribute) explicitly associated with it. The form label fixer remediates these elements, by finding a piece of text in the HTML code that is most likely the intended label for the form element, generating a label using that text, and adding a label tag with descriptive text in the code as the previously-untagged input field's label. *Id.*, at ¶38. Springer also describes an ImageFixer and ImageMapFixer used to remediate compliance issues stemming from websites lacking a text equivalent for every non-text element such as an image. *Id.*, at Tables II(a)-(b). Springer's ImageFixer adds alt attributes to images, image buttons, and image links, as well as replacing alt attributes with more appropriate (e.g., in length) or meaningful descriptions. *Id.* Springer's ImageMapFixer does the same things for AREAs (described above). *Id.*

146. Specifically, Springer describes its form label fixer as using a heuristic method (e.g., a series of rules) to analyze nearby elements and attributes

Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

in the website code (e.g., context cues) to identify candidate text for the form control label (e.g., input field). *Id.*, at ¶38. Springer allows for manual remediations, where a human can select which candidate text should be inserted as the label, and also allows for automatic remediations, where the system uses an “Autofix” library to automatically apply corrections based on previously-determined corrections, such as inserting text for labels (for untagged input fields) as previously-identified by the form label fixer, image fixer or by a human being. *Id.*, at ¶¶18, 21. Springer’s compliance retrofitter described above is used to build and maintain compliant HTML. It can be used to retrofit web sites by enabling automated and user-driven accessibility enhancements. It updates HTML code to make it compliant with 36 CFR § 1194.22. *Id.*, at ¶26.

147. Springer also discloses a set of imager fixer tools, e.g., ImageFixer and ImageMapFixer, which remediate compliance issues stemming from websites lacking a text equivalent for every non-text element such as an image. *Id.*, at Tables II(a)-(b). ImageFixer remediates by adding alt attributes to images, image buttons, and image links, as well as replacing alt attributes with more appropriate or meaningful descriptions. *Id.* ImageMapFixer does the same things for AREA elements. *Id.* Springer does not limit how alt attributes for images are determined,

Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

thus leaving the system open to numerous means for determining appropriate alt text for untagged images. *Id.*

148. Springer also discloses how to verify compliance accessibility standards and automatically retrofit the HTML of web pages to ensure compliance, as well as “[t]he fixers, or algorithms that make changes to HTML documents to bring the documents into compliance with the standard include manual fixes. Ex. 1005, at ¶16. For example, the ImageMapFixer adds alt attributes to AREA elements that currently do not have valid alt attributes.” *Id.*, at ¶15. Springer already discloses that descriptions can be manually input and stored in the autofix library. Further, Springer describes systems, including a compliance label fixer and image fixer, where “various modifications may be made.” *Id.*, at ¶56. To this end, a POSITA would have been motivated to improve Springer’s fixer tools with a community driven remediation system, which can be implemented via a remote server. Such a combination would have combined Springer’s checker and fixer tools with a database that stores manually entered descriptions that are accessible via a remote server. To the extent Springer does not explicitly disclose retrieving manual remediations from a server, it would have been obvious. Takagi, Bigham, and Hendry all disclose remediations originating from a remote server. Ex. 1006, at 196, col. 1, ¶¶3, 6; Ex. 1008, at 184, col. 2, ¶2; Ex. 1009, at ¶52.

Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

149. Furthermore, well before the '532 patent and similar to Springer's disclosure of an "Autofix" library by which a user can "build up a store of fixes to be applied to other documents," it was well known that it would be desirable to further store and share manual fixes across a community, e.g., referred to as a social accessibility service, as taught by Takagi. Ex. 1005, at ¶¶18, 21; Ex. 1006, at 195, col. 2, ¶2, 196, col. 1., ¶6. Thus, a POSITA would have been motivated to apply the teachings of Takagi for sharing manual remediation from a community, with Springer's "Autofix" library which allows for manual remediations, where individuals from Takagi's social accessibility service can provide any input needed for remediations.

150. A POSITA would also have been motivated to do so because it would save the number of manual remediations that would have to be done by any one individual in a community, and cut down on time it took to wait for a new manual remediation (if one already existed), which would then not delay other remediation requests for images that had no description. A POSITA would have expected the combination to work without undue experimentation because its already operating over the web and methods for pulling data from a wide range of sources over a network, such as databases, was well known. Ex. 1007, at 196, col. 1 ¶¶3, 6; Ex.

Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

1008, at 184, col. 2, ¶2; Ex. 1009, ¶¶14, 15, 52. *See also* Ex. 1005, at Tables I(a)-(b), II(a).

151. Furthermore, a POSITA would have understood that collaborative web accessibility improvements leveraged by the participation of an open community were well known by the time of the '532 patent. Ex. 1007, at 196, col. 1, ¶3. For example, various types of social approaches in addition to Takagi, such as the 1997 ALT-server, Asakawa, Teraguchi, WebInSight (Bigham), and WebVisum, could provide an improvement over the existing automated accessibility enhancements described by Springer. *Id.*; Ex. 1031, at 2-4; Ex. 1032, at 177, col. 2, ¶¶3-4, Figure 4; Ex. 1033; Ex. 1034; Ex. 1035, at 1:18-36, Figure 1.

152. A POSITA would have understood that Takagi teaches how to implement a social labeling system, including at the end user level, the supporting community tasks and using a remote public server as a metadata repository. Ex. 1007, at 196, col. 1, ¶6, col. 2 ¶4. Indeed, Takagi discloses a “system [that] consists of the three components[,] . . . end-user tools for problem reporting and client-side webpage transcoding based on external metadata; volunteer tools for metadata authoring: and a public server as the metadata repository.” Ex. 1007, at 196, col. 1, ¶6. Takagi’s approach also results in improved and effective generation of alternative text labels: “[t]he supporters resolved 85.1% of the 323

Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

submitted requests and only 5.9% remained unresolved as of May 1, 2009. The requests are first labeled as “unanswered” and supporters can change the status with the authoring tool for supporters.” *Id.*, at 198, col. 1, ¶2. Retrieving descriptions that had already been manually input by others would reduce the number of manual remediations that would have to be done by every other user, as well as the time it would take to receive a manual remediation, thereby increasing the speed, performance, and accuracy of the system. A POSITA would also have been motivated to combine Takagi’s social labeling approach with Springer’s fixers in order to improve the accuracy of creating appropriate label text for untagged input fields, or alt attributes to images, by leveraging Takagi’s social labeling via a remote public server compared to the only heuristic methods disclosed by Springer.

153. A POSITA would also have expected this combination to succeed with minimal modification. Indeed, Springer already includes a form fixer tool that analyzes context of a web page to find and extract an appropriate label for a web element, an image fixer for adding alt attributes to images and an image map fixer for “adding alt attributes to AREAs.” Ex. 1005, at Tables I(a)-(b), II(a)-(b), ¶¶16, 37-38. Takagi teaches that label extraction can be improved using an input label from a public repository authored by a volunteer. Ex. 1007, at 196, col. 1, ¶6.

Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

A POSITA could simply modify Springer’s “fixer” code to also query a database for matching descriptive attributes that had been authored by others. A POSITA could also add an interface (e.g., a quick fix wizard in Takagi) that allows supporters to respond to users’ requests and quickly make alternative text metadata via a remote server with minimal work and the rest of Springer system, such as its FormChecker tool, would work as intended (with improved results from using the Takagi social labeling methods). Ex. 1007, at 197, ¶¶3, 5; Ex. 1005, at Tables I(a)-(b), III.

154. Takagi also discloses WYSISYG-style editor, referred to as a “page map,” for metadata authoring but does not elaborate on the details of how untagged elements, e.g., inaccessible images include those without alternative text or with useless alternative text (such as “banner” or “spacer”), are detected. Ex. 1007, at 197, col. 1, ¶4. Thus a POSITA would have understood the benefit of combining the Takagi with the Springer system which discloses how to verify compliance accessibility standards and automatically or manually retrofit the HTML of web pages to ensure compliance. Ex. 1005, at ¶¶8, 15, 16, 38. Similar remediation techniques were well known. *See also* Ex. 1015, at ¶¶2, 4, 15, 18; Ex. 1006, at ¶72.

Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

155. Additionally, as I explained in Section IV, it was a commonly known approach to store alternative text in a remote server, where the alternative text was generated with the help manual remediation. Ex. 1008, at 186, col. 2 ¶¶3-5; Figure 4, Ex. 1007, at 196, col. 1, ¶6; *see also* Ex. 1031, at 2-4; Ex. 1032, at 177, col. 2, ¶¶3-4, Figure 4; Ex. 1033; Ex. 1034; Ex. 1035, 1:18-36, Figure 1.

156. A POSITA could have also reached a similar conclusion or combination by utilizing Springer’s “checkers” along with Takagi’s “screen-reader-independent end user tool,” as both features were designed for, and capable of, checking for accessibility compliance issues. Takagi, at 196, col. 2, ¶4 (“This native application also added an advanced feature to handle elements ignored and inaccessible with earlier approaches. These elements include images with no alternative text . . .”).

157. Therefore, it would have been obvious to a POSITA to combine Springer, which discloses a heuristic method for checking an HTML document of a web page for compliance by running a checker and fixer tool to remediate a section of the HTML document containing the flagged violation by modifying HTML code, with Takagi, which discloses inputting alternative text from a remote server machine.

2. Independent Claim 1

- a. **“A computer-implemented method of programmatically assigning a descriptive attribute to an untagged element on a web page to enable an audible description of the untagged element, the web page having an associated document object model (DOM), the method comprising:” (Claim 1[pre])**

158. To the extent the preamble is limiting, Springer and Takagi disclose this element because Springer teaches using fixers including an “Autofix” library to assign (to programmatically assign) a descriptive attribute (label text for input fields, or an alt attribute) to an element (untagged element such as an input field, an image, an AREA element) on a web page having an associated document object model.

159. Springer discloses a fixer that may use an “Autofix” library to automatically apply corrections, e.g., based on previously-determined corrections, such as inserting text for labels (for untagged input fields) as previously-identified by a form label fixer or by a human being, alt attributes for an image, or an AREA element missing such an attribute Ex. 1005, at ¶¶15, 16, 18, 21, 52, Tables I(b), II(b). For example, Springer discloses a FormFixer tool for “Add[ing] LABEL elements to forms” and “Add[ing] a “selected” attribute to one OPTION element in each of SELECT form control.” *Id.*, at Tables I(b), II(b). Also, Springer discloses an ImageFixer for “Add[ing] alt attributes to images” and a ImageMapFixer for “Add[ing] alt attributes to AREAs” which includes “ add[ing] alt attributes to

Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

AREA elements that currently do not have valid alt attributes.” *Id.*, at Tables I(b), II(b), ¶16. Springer discloses that Springer’s system is intended to be used by “a user of an assistive technology that permits verbal communication of web content.” *Id.*, at ¶29. A POSITA would therefore have understood that Springer’s system for providing added descriptive attributes enables commonly used screen readers to provide an audible description of the input field, alt attribute for an image, or an alt attribute for an AREA element.

160. Springer discloses its checkers “check a data model of a web page” (e.g., DOM) (*id.*, at Abstract), and that it interacts with webpages through a “document tree” or “tree format” (*id.*, at ¶¶15, 25, 38, 49). A POSITA would have understood Springer’s “document tree” to be a document object model, as discussed below. From at least the above disclosures, a person of ordinary skill in the art would have understood that Springer can be used to check and fix Document Object Models (DOM) associated with a webpage as well as the HTML code itself. *See also id.*, at ¶7. To the extent Springer does not explicitly disclose Document Object Models (DOM) associated with a webpage, a person of ordinary skill in the art would have understood that Springer can be used to check and fix Document Object Models (DOM) associated with a webpage as well as the HTML code itself. *See also id.*

Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

161. Furthermore, Springer's system can check for compliance and correct an HTML document and has an "associated document object model (DOM)," as claimed. In particular, Springer discloses "checking an HTML document of a web page for compliance with Section 508 of the Rehabilitation Act of 1973" and "a method for fixing and/or checking an HTML document for accessibility." *Id.*, at ¶7. Springer also states that its "method, apparatus and computer program may use checkers to check a data model of a web page for accessibility." *Id.*, at Abstract.

162. A POSITA would have understood the data model of a webpage provided as an HTML document to be a DOM. *See* Ex. 1014 (W3C DOM v1.0), at 2 ("The Document Object Model provides a standard set of objects for representing HTML and XML documents, a standard model of how these objects can be combined, and a standard interface for accessing and manipulating them."). Thus, Springer discloses "[a] computer-implemented method of programmatically assigning a descriptive attribute to an untagged element on a web page to enable an audible description of the untagged element, the web page having an associated document object model (DOM)" as claimed.

163. Furthermore, Takagi also discloses. Takagi discloses a "script to apply [] metadata [that] directly modifies the DOM as appropriate for each target HTML element," where the "most frequent metadata type was the "alttext. This metadata

covers alternative texts for non-textual objects, labels.” Ex. 1007, at 198, col. 1,

¶1. “As a result, the end user can browse the webpage made accessible with the metadata.” Ex.1007, 196, col. 1, ¶6. Figure 1 of Takagi also recognizes that the end user utilizes a screen reader to speak the new metadata or descriptive attributes. Takagi thus discloses the preamble.

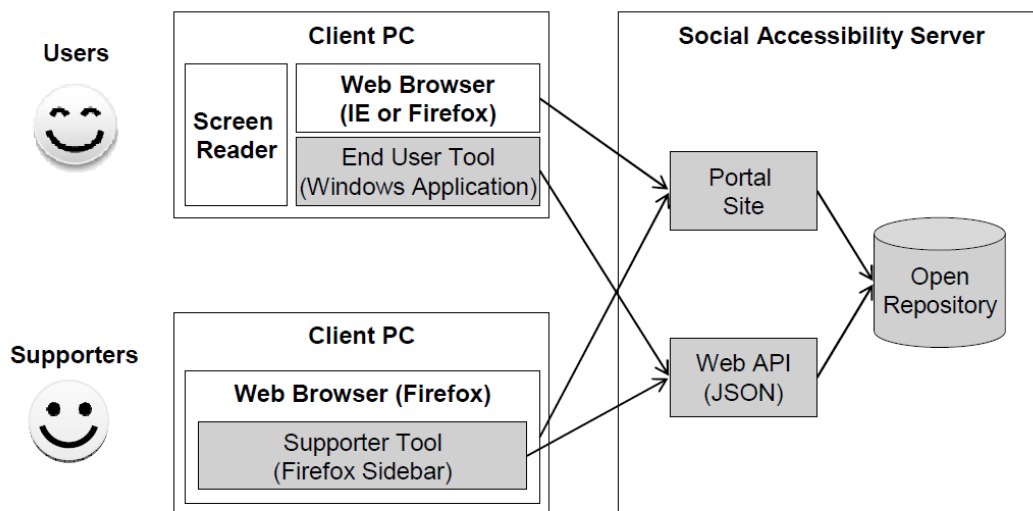


Figure 1. Basic Architecture of Social Accessibility Service

Ex. 1007, Figure 1.

164. Accordingly, Springer and Takagi disclose and render obvious the preamble of claim 1.

- b. “dynamically analyzing, by a computer system, code associated with the web page, the code comprising at least the DOM or HTML code;” (Claim 1[a])

Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

165. Springer and Takagi disclose this element, because Springer discloses using a computer system to dynamically analyze the data model associated with a web page (the DOM) to check for accessibility compliance issues. For example, Springer discloses a “compliance retrofitter,” a software, that can be executed on a computer system for checking for compliance “an HTML document of a web page” where the checking includes using “checkers to check a data model of a web page for accessibility.” Ex. 1005, at ¶7, Abstract. Springer also states that “[t]he checkers implemented by the compliance retrofitter are sets of tests that are run against documents, for example files or web pages, to analyze compliance with a standard, e.g., Section 508.” *Id.*, at ¶14. Springer discloses that it interacts with webpages through a data model, “document tree,” and “tree format.” *Id.*, at ¶¶15, 25, 38, 49. As explained in the above element, the data model and “document tree” is a software-accessible format known as a document object model or DOM.

166. Furthermore, Springer discloses a “compliance retrofitter” that “can be used to scan the HyperText Mark-up Language (HTML) code that powers the web, identify violations with Section 508 Standards (checkers), and correct those violations by inserting the necessary corrections into the code in real time (fixers).” Ex. 1005, at ¶10. Springer also states that the “compliance retrofitter may be platform independent by using a platform independent language such as Java (Sun

Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

Micro Systems, Inc. of Palo Alto, Calif.), and can be run on any platform for which a Java Virtual Machine exists, including Windows 95/98, Windows 2000, Windows NT and Linux.” *Id.*, at ¶57. Therefore, the compliance retrofitter is executed on a computer system. Thus, Springer discloses “dynamically analyzing, by a computer system, a code associated with the web page, the code comprising at least HTML or the DOM” as claimed.

167. Similarly, Takagi discloses this element because Takagi discloses using a social accessibility pilot service which makes use of external metadata and the participation of an open community to improve the accessibility of webpages. Ex. 1007, at 196, col. 1, ¶6. For example, Takagi discloses a social accessibility pilot system that consists of the three components: 1) end-user tools for problem reporting and client-side webpage transcoding based on external metadata; 2) volunteer tools for metadata authoring; and 3) a public server as the metadata repository. *Id.* Both end users and volunteers access the repository using client-side tools, which interact with the server through Web APIs. *Id.*

168. Takagi includes an end user tool such as a “native application for Windows®.” Ex. 1007, at 196, col. 2, ¶4. A POSITA would have understood that a “native application for windows” to mean that the application was developed to run on a computer system. Takagi’s native application enabled advanced features

Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

such as “handl[ing] elements . . . includ[ing] images with no alternative text.” *Id.*

Thus, Takagi discloses “dynamically analyzing, by a computer system, a code associated with the web page, the code comprising at least HTML or the DOM” as claimed.

169. Furthermore, Takagi's native application “works by injecting special JavaScript code into the webpages for the selected functions,” including Takagi's “client-side webpage transcoding based on external metadata.” Ex. 1007, at 196, col. 2, ¶3, col. 1, ¶6. A POSITA would have understood this to describe Takagi analyzing the DOM or HTML code of a webpage as part of the transcoding process. Furthermore, a POSITA would have understood from this disclosure that Takagi's end user tool analyzed the HTML or DOM of a web page to identify compliance issues such as images without alt text.

170. Therefore, Springer and Takagi thus disclose and render obvious this element.

- c. **“detecting, by the computer system, one or more compliance issues relating to web accessibility standards in the code, wherein at least one of the one or more compliance issues comprises the untagged element lacking an adequate descriptive attribute;” (Claim 1[b])**

171. Springer and Takagi disclose this element because Springer discloses using a computer system to detect one or more untagged elements such as an input

Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

field lacking an adequate descriptive attribute, and images and AREA elements that lack valid alt text (more compliance issues) relating to web accessibility standards in the code. For example, Springer discloses a “FormChecker” tool that detects form fields that lack a descriptive label (e.g., an untagged element such as an INPUT field lacking an adequate descriptive attribute). Springer also discloses “form fields are HTML elements that allow data entry into a form” and that “[a] common form field is the INPUT element.” Ex. 1005, at ¶37. Springer further discloses that its checkers, such as FormChecker, “[c]hecks that labels are explicitly associated with form controls.” (*id.*, at Tables II(a)-II(b)). Each of these compliance issues are related to web accessibility compliance standards. *Id.*, at ¶¶37-38.

172. In addition, Springer discloses its checkers, “ImageChecker” and “ImageMapChecker,” can detect images that have or lack valid alt text descriptions/attributes (i.e., an image lacking an adequate description). For example, the ImageChecker tool can be used to check “that images have valid alt attributes” and “that long descriptions are valid” and an ImageMapChecker that checks that portions of images that functions as a link (called AREA elements) to have valid alt attributes. *Id.*, at Tables I(a)-(b), ¶15. Springer uses those checkers to scan the web page “to determine if all of the AREAS had valid ‘alt attributes,’

Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

i.e., to determine if each AREA had associated with it an alternative textual description. (An AREA element defines a part of an image that functions as a link. Since some users do not use browsers that display images, having valid alt attributes for AREAs allows these users to navigate through the page.)” *Id.*, at ¶15. This is also shown, for example, in Tables I(a) and I(b), excerpted below.

Table I(a). 36 CFR § 1194.22 Checkers

36 CFR § 1194.22 Paragraph	Description	Checker	Manual Review
A	Paragraph (a) requires that a text equivalent for every non-text element shall be provided. The provision is necessary because assistive technology cannot describe pictures, but can convey the text information to the user.	ImageChecker ObjectTextChecker ImageMapChecker AsciiArtChecker AppletChecker	No Manual Review Required

Table I(b). Checker Functionality

36 CFR § 1194.22 Paragraph	Checker	Functionality
A	ImageChecker ObjectTextChecker ImageMapChecker AppletChecker	Checks that images have valid alt attributes. Checks that image links have valid alt attributes. Checks that image buttons have valid alt attributes. Checks that long descriptions are valid. Checks that OBJECT elements have enclosed textual descriptions. Checks that AREAs have valid alt attributes. Checks for valid textual alternatives in APPLET elements.

Id., at Tables I(a)-(b) (excerpted).

Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

173. Thus, Springer discloses “detecting, by the computer system, one or more compliance issues relating to web accessibility standards in the code, wherein at least one of the one or more compliance issues comprises an untagged element, the untagged element is an element lacking an adequate descriptive attribute” as claimed.

174. Takagi also discloses this element. As discussed above, Takagi includes an end user tool such as a “native application for Windows®.” Ex. 1007, at 196, col. 2, ¶4. A POSITA would have understood that a “native application for windows” to mean that the application was developed to run on a computer system. Takagi’s native application enabled advanced features such as “handl[ing] elements . . . includ[ing] images with no alternative text.” *Id.* A POSITA would have understood from this disclosure that Takagi’s end user tool analyzed the HTML or DOM of a web page to identify compliance issues such as images without alt-text.

175. Moreover, Takagi also discloses a quick fix wizard which can be used to detect accessibility problems related to untagged elements such as images. *Id.* at 197, col. 1, ¶5. For example, Takagi discloses a “quick fix wizard” which includes “a simpl[e] interface for making alternative text metadata. [The quick fix wizard] sequentially displays each [one] of [any detected] inaccessible images. Each

Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

image is followed by the description and the same form for new metadata. This wizard allows a supporter to quickly fix all of the image-related accessibility problems in a webpage by simply filling in the fields and clicking on the “next” button, as long as the heuristics successfully detect all of the inaccessible images.”

Id. This disclosure further demonstrates to a POSITA that the Takagi authors knew how to detect compliance issues as recited by this claim element.

176. Springer and Takagi thus disclose “detecting, by the computer system, one or more compliance issues relating to web accessibility standards in the code, wherein at least one of the one or more compliance issues comprises an untagged element, the untagged element is an element lacking an adequate descriptive attribute” as claimed.

- d. **“applying, by the computer system, one or more pre-existing remediation code to the one or more compliance issues, wherein the one or more pre-existing remediations is generated by a remote server system performing at least:” (Claim 1[c][1])**

177. Springer and Takagi disclose this element. Takagi discloses using a social accessibility service to remediate accessibility issues on a webpage. The Takagi system “uses external metadata and the participation of an open community, and it allows [for] improving the accessibility of any webpage without asking the site owners to modify the website. Both end users and volunteers (also

Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

called supporters) access the repository using client-side tools, which interact with the server through Web APIs. They can also access the repository by visiting the portal site with their Web browsers.” Ex.1007, at 196, col. 1, ¶6. For example, Takagi discloses using a social accessibility pilot service to apply external metadata (remediation code) to an accessibility issue on a webpage, where the external metadata is generated from a public server as the metadata repository. For example, Takagi discloses:

[S]ocial Accessibility [pilot system that] uses external metadata and the participation of an open community, and it allows [for] improving the accessibility of any webpage without asking the site owners to modify the website. The [social Accessibility] pilot system consists of the three components shown in Figure 1: end-user tools for problem reporting and client-side webpage transcoding based on external metadata; volunteer tools for metadata authoring; and a public server as the metadata repository. Both end users and volunteers (also called supporters) access the repository using client-side tools, which interact with the server through Web APIs. They can also access the repository by visiting the portal site with their Web browsers.”

Ex. 1007, at 196, col. 1, ¶6.

178. Takagi’s Figure 1 shows a basic architecture for the Social Accessibility Pilot Service. As shown in Figure 1 below, and described above, Takagi’s social accessibility pilot system consists of end-user tools, located in a client PC, supporter tools, located in another client PC, and an open repository,

Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

part of a social accessibility server. Ex. 1007, at Figure 1, 196, col. 1, ¶6. A POSITA would have understood that the Takagi's social accessibility service provides remediations to a user through a remote server. *Id.* Takagi discloses that "[t]he public repository allows that metadata to benefit every end user, not just the first user who reported the problem." Ex.1007, at 196, col. 1, ¶6. The accessibility metadata is provided by a remote supporter and stored in the accessibility server/database to be loaded, and applied to the webpage of an end user. Ex.1007, 196, Fig. 2. Provides remediations to a user through a remote server is shown in Figures 1-2 of Takagi, below:

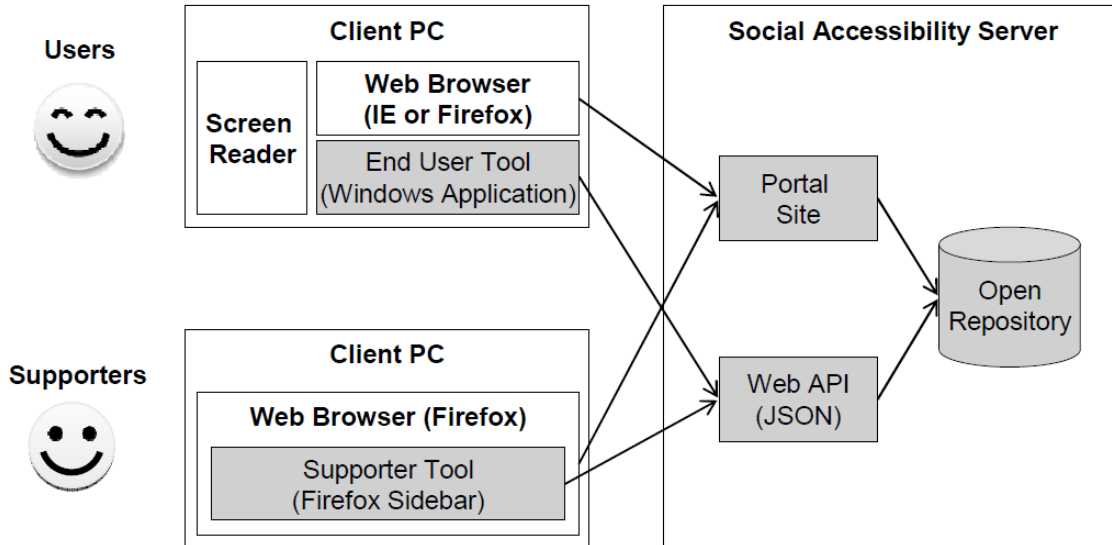
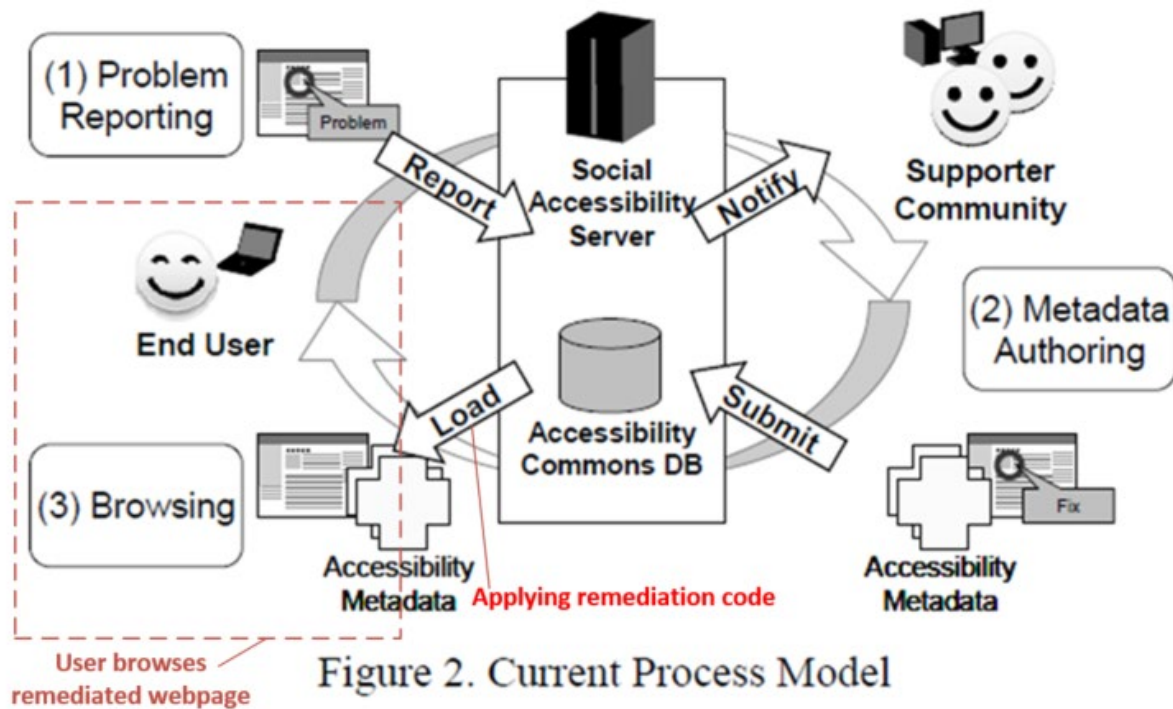


Figure 1. Basic Architecture of Social Accessibility Service

Id., at Figure 1.

Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

179. Furthermore, Takagi's social accessibility pilot system provides "accessibility improvement" by "[1][f]irst, an end user report[ing] an accessibility problem on a webpage. This report goes to the supporter community instead of to the site owner (as with a contact form). [2] Then a supporter creates external metadata to fix the reported problem. As a result, the end user can [3] browse[s] the webpage made accessible with the metadata. This reporting-based strategy allows quick solutions to accessibility problems." Ex. 1007, at 196, col. 1, ¶6. Further, Takagi describes that "[t]he public repository allows that metadata to benefit every end user, not just the first user who reported the problem." *Id.* Takagi's Figure 2 shows the process model used by the social accessibility pilot system including a "Social Accessibility Server":



Id., at Figure 2 (annotated).

180. As shown in Figure 2 of Takagi, and described above, the end user [1] reports a problem via the social accessibility server to notify the remote supporter community, [2] the remote supporter community can then submit accessibility metadata to remediate the reported problem which is submitted to the metadata repository or accessibility database (DB), and [3] the accessibility metadata is loaded, e.g., applied, to the webpage via the social accessibility server for the user to browse. Ex. 1007, at 196, col. 1, ¶6, Figure 2.

181. To the extent “applying . . . one or more pre-existing remediation code to the one or more compliance issues” is explicitly performed by a remote server, it

Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

would have been obvious. Indeed, Asakawa depicts an earlier version of the same system of Takagi that utilizes the remote server to access the code, demonstrating that this was known by Takagi and Asakawa, e.g., Takagi is a co-author to the publication of Asakawa. Ex. 1032, at Figure 4. Thus, a POSITA at the time of the '532 patent, would have considered the arrangement to be one of many known options. The architecture of Asakawa's annotation-based transcoding system is illustrated in Asakawa's Figure 4 below (Ex. 1032, Figure 4).

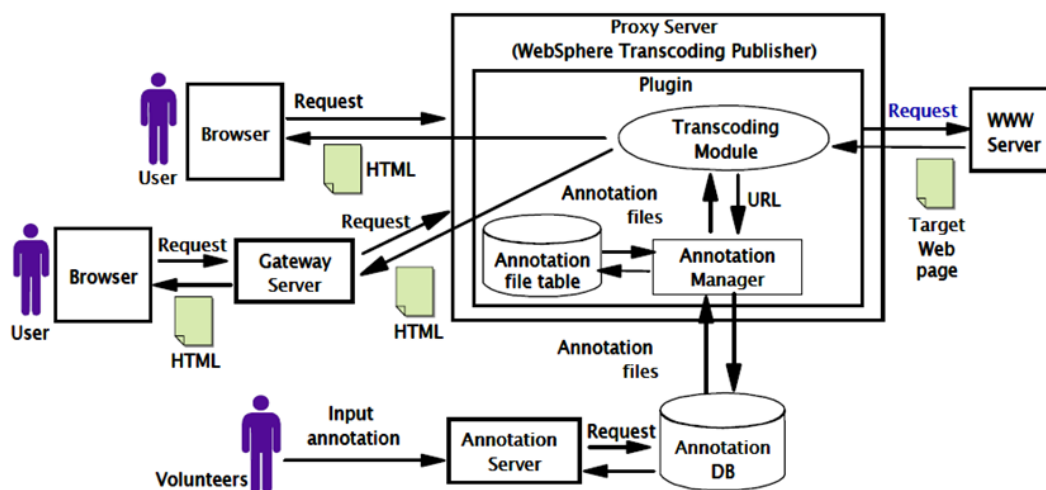


Figure 4: The system architecture

Ex. 1032, Figure 4

182. As discussed above, a POSITA would have been motivated to combine Takagi with the Springer system in order to improve upon Springer's compliance retrofitter functions for determining and assigning appropriate

descriptive attributes for untagged elements, e.g., such as labels for untagged input fields and images lacking appropriate alt-text.

183. Springer discloses using a computer system to select detected compliance issues to pre-existing remediation scripts (e.g., remediation code) for the type of specifically detected issue. Springer also discloses that its Checker modules have corresponding Fixer tools (“pre-existing remediation scripts”), such that: (1) a specific Checker tool is run, looking for a specific type of compliance issue; (2) each violation/issue detected by that Checker is mapped to a corresponding Fixer tool, and (3) the Fixer tool, in the “autofix” embodiment, automatically remediates that type of violation/issue. Ex. 1005, at ¶10; *see also id.*, at ¶21 (discussing autofix mode), Tables I(b) (listing Checker tools), II(b) (listing corresponding Fixer tools), Abstract. Each fixer tool is pre-existing software that, when run, remediates the web code – and is thus a pre-existing remediation code.

184. After a checker tool is run to detect specific types of compliance issues, each violation/issue detected by that checker which is selected for a corresponding fixer tool to fix and/or remediate the issue.

185. In one example, Springer discloses that a form label checker (e.g., “FormChecker”) may be used to identify a form field in a document that does not

Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

have a label explicitly associated with it, and that a form label fixer (e.g., “FormFixer”) may then be used to find a piece of text in the HTML page that is most likely the intended label for the form element and input that text as the label. Ex. 1005, at Tables I(a)-(b), II(a)-(b); *see also id.*, at ¶¶37-38. Springer further discloses a fixer that may use an “Autofix” library to automatically apply corrections, e.g., based on previously-determined corrections, such as inserting text for labels (for untagged input fields) as previously-identified by a form label fixer or by a human being, alt attributes for an image, or an AREA element missing such an attribute *Id.*, at ¶¶15-16, 18, 21, 52, Tables I(b), II(b).

186. The one or more compliance issues detected by Springer include elements lacking an adequate descriptive attribute. For example, Springer discloses its “ImageChecker” and “ImageMapChecker” can detect images that have or lack valid alt text descriptions/attributes (i.e., a graphic lacking an adequate description). *Id.*, at ¶¶52, 15.

187. Springer uses those checkers to scan the web page to check the code “to determine if all of the AREAS had valid ‘alt attributes,’ i.e., to determine if each AREA had associated with it an alternative textual description. (An AREA element defines a part of an image that functions as a link. Since some users do not use browsers that display images, having valid alt attributes for AREAs allows

Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

these users to navigate through the page.)” *Id.*, at ¶15. This is also shown, for example, in Tables I(a) and I(b), excerpted below.

Table I(a). 36 CFR § 1194.22 Checkers

36 CFR § 1194.22 Paragraph	Description	Checker	Manual Review
A	Paragraph (a) requires that a text equivalent for every non-text element shall be provided. The provision is necessary because assistive technology cannot describe pictures, but can convey the text information to the user.	ImageChecker ObjectTextChecker ImageMapChecker AsciiArtChecker AppletChecker	No Manual Review Required

Table I(b). Checker Functionality

36 CFR § 1194.22 Paragraph	Checker	Functionality
A	ImageChecker ObjectTextChecker ImageMapChecker AppletChecker	Checks that images have valid alt attributes. Checks that image links have valid alt attributes. Checks that image buttons have valid alt attributes. Checks that long descriptions are valid. Checks that OBJECT elements have enclosed textual descriptions. Checks that AREAs have valid alt attributes. Checks for valid textual alternatives in APPLET elements.

Id., at Tables I(a)-(b) (excerpted).

188. Springer also discloses fixers such as ImageFixer and ImageMapFixer that correspond to its checkers, i.e., ImageChecker and ImageMapChecker:

Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

Table II(b). Fixer Functionality

36 CFR § 1194.22 Paragraph	Fixer	Functionality
A	ImageFixer	Adds alt attributes to images Adds alt attributes to image buttons Adds alt attributes to image links Replaces alt attributes that are too long with shorter ones Replaces vague alt attributes on images with more meaningful ones Replaces empty or invalid longdesc descriptions with valid ones Replaces vague alt attributes on image buttons with more meaningful ones
	ObjectTextFixer	Adds a valid textual description to OBJECT elements
	AppletFixer	Adds textual alternatives to APPLET elements
C	ColorContrastFixer	Allows developer to select a new color which has sufficient color contrast
F	ImageMapFixer	Adds alt attributes to AREAs Replaces vague alt attributes in AREAs with more meaningful ones

Ex. 1005, at Table II(b) (excerpted).

189. For example, the ImageFixer can add an alt attribute to an image and the ImageMapFixer will add alt attributes to AREAs. *Id.* Springer’s compliance retrofitter can check and fix various elements of a web page each time the compliance retrofitter is run. Springer also discloses the ability to check, and if necessary, fix elements that already have assigned alt attributes and insert alt attributes for an AREA element missing such an attribute. For example, the ImageFixer of Springer will “[r]eplace[] alt attributes that are too long with shorter ones,” replace “vague alt attributes on images with more meaningful ones” and

Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

“add[ing] alt attributes to AREA elements that currently do not have valid alt attributes.” *Id.*, at ¶16, Tables I(b), II(a)-(b).

190. Furthermore, as discussed in claim element 1[a] above Springer discloses a “compliance retrofitter,” a software, that can be executed on a computer system and also discloses the “compliance retrofitter may be platform independent by using a platform independent language such as Java (Sun Micro Systems, Inc. of Palo Alto, Calif.), and can be run on any platform for which a Java Virtual Machine exists, including Windows 95/98, Windows 2000, Windows NT and Linux.” *Id.*, at ¶¶7, 57. Therefore, Springer in view of Takagi discloses “applying, by the computer system, one or more pre-existing remediation code to the one or more compliance issues, wherein the one or more pre-existing remediations is generated by a remote server system performing at least . . .” as claimed.

**e. “accessing the code associated with the web page”
(Claim 1[c][2])**

191. Springer and Takagi disclose this element. Takagi discloses a visual metadata authoring interface that allows a remote computer to access the code of the webpage. Ex. 1007, at 197, col. 1, ¶4, Figure 3.

192. For example, Takagi discloses:

[A] WYSISYG-style editor for metadata authoring. It creates an image of the webpage and allows supporters to quickly choose a webpage element for external metadata

Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

by just clicking on that element. The created image adds exclamation marks for such metadata targets as inaccessible images to help the supporters add useful metadata (Figure 3 (a)). Inaccessible images include those without alternative text or with useless alternative text (such as “banner” or “spacer”). Also marked are image links pointing to the same destinations as nearby text links. Clicking on an exclamation mark invokes a dialog box with a description of the accessibility problem and how to fix it, with an easy-to-use form for new metadata. This visual editor also uses our brightness-gradation technique [22] to visualize the estimated times required for a screen reader to reach each element to help supporters with headings metadata. Darker elements take longer to access, indicating that additional headings may help. When a supporter clicks on such an element, a dialog box appears to specify a heading level (h1-h6) with an optional text description. The same dialog box also has options for other types of metadata such as alternative text for an image that the system failed to recognize as inaccessible. Once metadata is created, an icon indicating the metadata type is shown at the position where the metadata was added (Figure 3 (b)). A supporter can edit and improve metadata by clicking on a metadata icon.”

Ex. 1007, at 197, col. 1, ¶4.

193. Takagi also discloses:

The quick fix wizard is a simpler interface for making alternative text metadata. It sequentially displays each of the inaccessible images. Each image is followed by the description and the same form for new metadata. This wizard allows a supporter to quickly fix all of the image-related accessibility problems in a webpage by simply filling in the fields and clicking on the “next” button, as long as the heuristics successfully detect all of the inaccessible images.

Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

Id., at 197, col. 1, ¶5.

194. Figure 3 of Takagi (below) shows a page map “WYSISYG-style editor” for accessing the code associated with a webpage:



Figure 3. Page Map

Id., at Figure 3. A POSITA would have understood, for example, that Takagi’s WYSISYG-style editor for metadata authoring accesses code associated with a webpage in order to create an image of the webpage that adds exclamation marks for metadata targets such as inaccessible images to help the supporters add useful metadata.

195. In addition, a POSITA would have been motivated to extend Springer based using the social accessibility pilot service of Takagi. A POSITA would have

Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

understood, that Springer's checkers could have been executed on different computers, including a user's local computer or on a remote machine. A POSITA also would have understood that Springer's methods could have easily been implemented using, or for, Takagi's metadata authoring tool, e.g., the WYSISYG-style editor, to take advantage of the benefits provided by social accessibility pilot service as disclosed by Takagi. Ex. 1007, at 197, ¶¶4, 5, Figure 2, 3.

196. Springer also discloses accessing the HTML and DOM associated with the web page in order to detect compliance issues. Springer states: "[t]he compliance retrofitter will operate on the HTML of the document and will check compliance." Thus, Springer discloses accessing the code associated with the web page in order to run a set of accessibility tests. Ex. 1005, at ¶14.

197. The compliance retrofitter after accessing the code on the web page will analyze and describe violations. Enhanced reporting may be included and may include summary reporting by total violations and violation type, page-by-page HTML reports, page-by-page spidering (building multiple pages by crawling through a publicly available website) and overall spidering speed increase." Ex. 1005, at ¶27.

198. Springer further discloses ImageMapChecker, one of the checkers ran in the creation of screen shot 100 to check compliance of the web page.

Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

Specifically in one embodiment, Springer discloses: “ImageMapChecker, in part, checked the HTML code of www.yahoo.com to determine if all of the AREAS had valid "alt attributes," i.e., to access the HTML code and determine if each AREA had associated with it an alternative textual description.” Note that an AREA element defines a part of an image that functions as a link, and having valid alt attributes for AREAs allows users who do not use browsers that display images to navigate through the webpage. The results created by running the checkers are presented in tree format by each analyzed page and the corresponding violations, e.g., AREA missing alt attribute, in the history window. In the embodiment of Springer, an HTML window 40 provides the HTML code for the currently selected document (www.yahoo.com 24) with highlights of the currently selected instance. Below HTML window 40 is fixer window 50 that displays information related to fixers with various functionalities, or algorithms that make changes to HTML documents or other documents to bring the documents into compliance with the standard. As can be seen, the ImageMapFixer adds alt attributes to AREA elements that currently do not have valid alt attributes. Ex. 1005, at ¶¶15-16.

199. Therefore, Springer and Takagi disclose and render obvious this element.

- f. “receiving, by the remote server system, via a remediation interface input from a user, the input

**comprising a remediation action to manually
remediate a non-programmatically-fixable
compliance issue associated with the web page;
(Claim 1[c][3])**

200. Springer and Takagi disclose this element.

201. Takagi describes multiple ways in which a remote server system receives, via a remediation interface input from a user, input comprising a remediation action to manually remediate a non-programmatically-fixable compliance issue associated with the web page. Takagi's Figure 1 shows a basic architecture for the Social Accessibility Pilot Service including showing a "Social Accessibility Server." As shown in Figure 1 below, and described above, Takagi's social accessibility pilot system consists of end-user tools, e.g., a remediation interface, located in a client PC, supporter tools, located in another client PC, and an open repository, part of a social accessibility server. Ex. 1007, at Figure 1, 196, col. 1, ¶6. Takagi teaches that users use an end user tool to report problems and request a remediation action, and directly interact with the social accessibility server through Web APIs as shown in Figure 1 of Takagi. *Id.*

202. As shown in Figure 1 (below), users use an end user tool to report problems to, and request remediation action from, a remote server. Ex.1007, 196, Figure 1. A POSITA would have understood that the Takagi's social accessibility service to disclose a remote server. *Id.* Annotated Figure 1 of Takagi shows a

basic architecture for the Social Accessibility Pilot Service including a “Social Accessibility Server”:

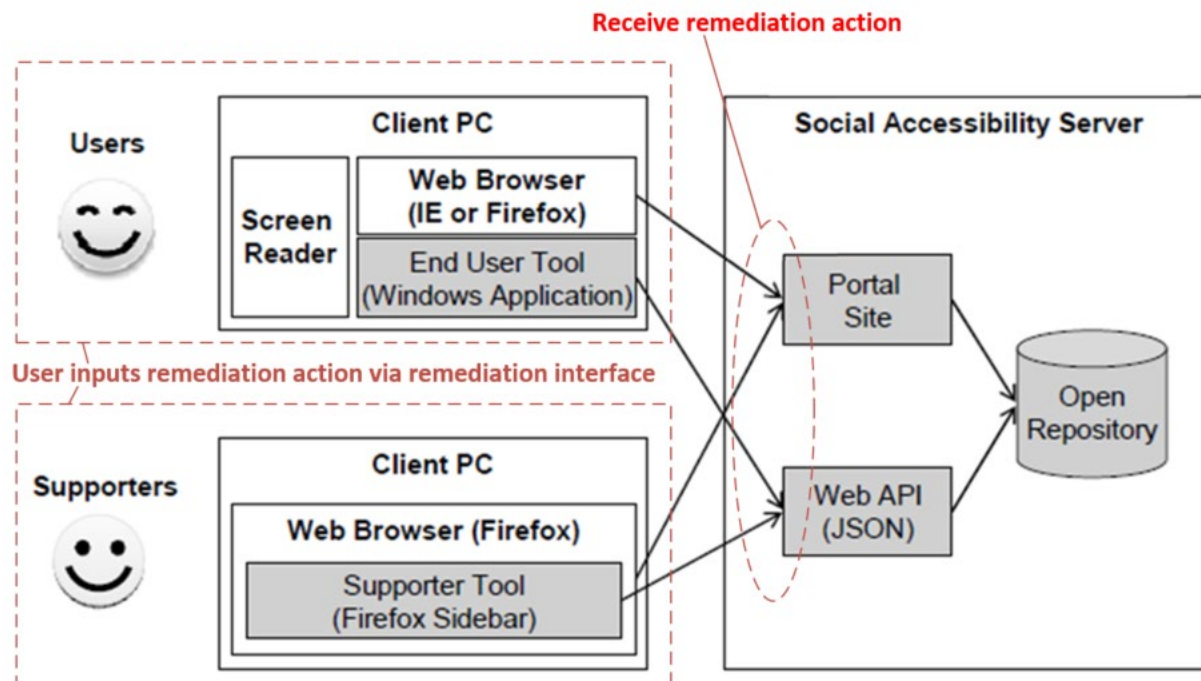


Figure 1. Basic Architecture of Social Accessibility Service

Id., Figure 1 (annotated). In addition, the end user tools also “allow expert users to create alternative text labels for the previously inaccessible elements based on the context.” Ex.1007, 196, col. 2, ¶4. Each of these is an input comprising a remediation action to manually remediate a non-programmatically-fixable compliance issue associated with the web page.

203. Additionally, Takagi social accessibility pilot system provides accessibility improvement by “[1][f]irst, an end user report[ing] an accessibility problem on a webpage. This report goes to the supporter community instead of to

Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

the site owner (as with a contact form). [2] Then a supporter creates external metadata to fix the reported problem. As a result, the end user can [3] browse the webpage made accessible with the metadata. This reporting-based strategy allows quick solutions to accessibility problems.” Ex. 1007, at 196, col. 1, ¶6. Further, Takagi describes that “[t]he public repository allows that metadata to benefit every end user, not just the first user who reported the problem.” *Id.* Annotated Figure 2 of Takagi shows the process model used by the social accessibility pilot system including a “Social Accessibility Server”:

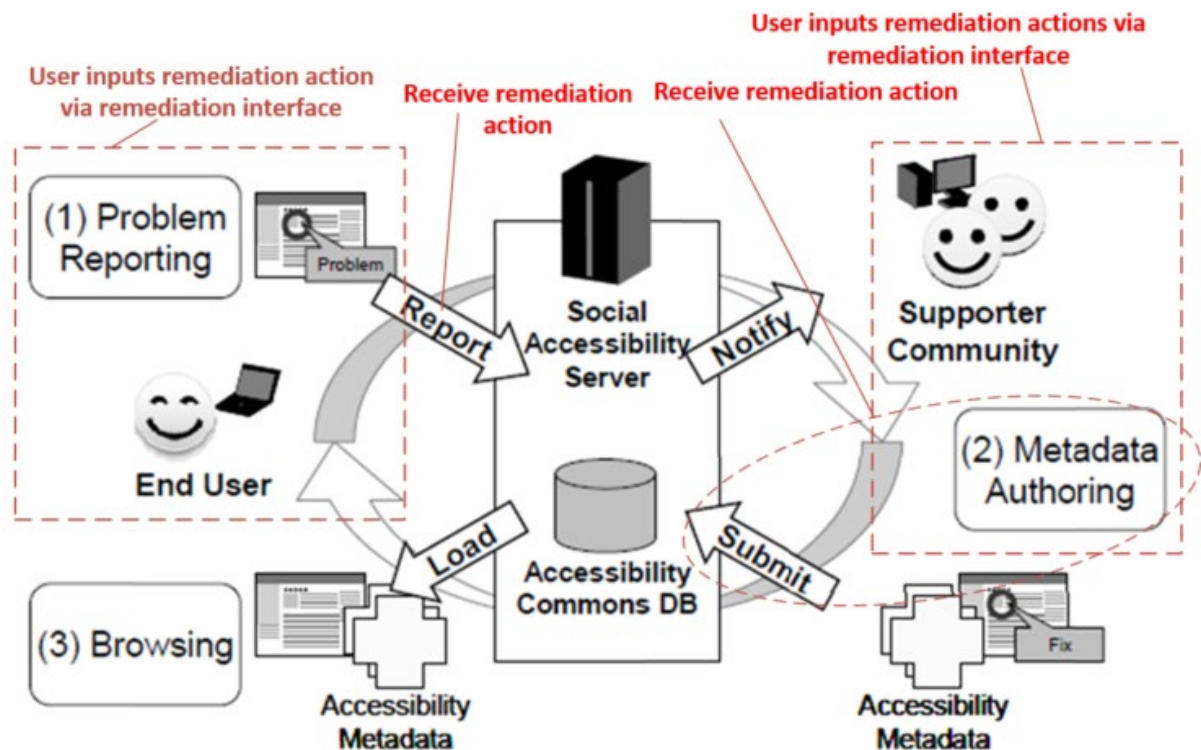


Figure 2. Current Process Model

Id., at Figure 2 (annotated).

Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

204. As shown in Figure 2 of Takagi, and described above, the end user, using an end-user tool shown in Figure 1, [1] reports a problem via the social accessibility server to notify the supporter community, [2] the supporter community can then submit accessibility metadata to remediate the reported problem which is submitted to the metadata repository or accessibility database (DB), and [3] the accessibility metadata is loaded to the webpage via the social accessibility server for the user to browse. Ex. 1007, at 196, col. 1, ¶6, Figure 1, Figure 2.

205. Furthermore, Takagi's Figure 3 shows an exemplary supporter tool, e.g., a remediation interface, referred to in Takagi as a "page map." *Id.*, at 197, col. 1, ¶¶3, 4, Figure 3:

Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

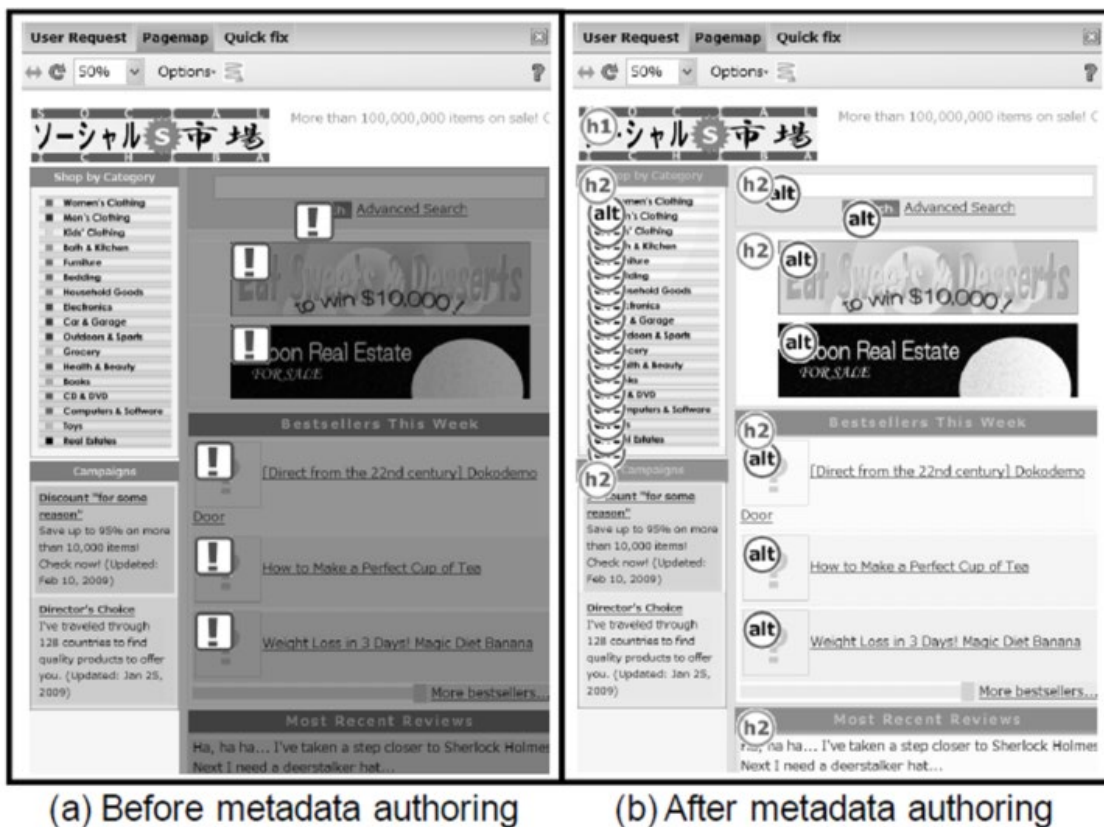


Figure 3. Page Map

206. As shown in Figure 3 of Takagi, and described above, Takagi discloses a “page map” supporter tool that “allows supporters to quickly choose a webpage element for external metadata by just clicking on that element.” Takagi also discloses that the supporter tool “help[s] the supporters add useful metadata,” and provides for “an easy-to-use form for new metadata.” Ex. 1007, at 197, col. 1, ¶4.

207. For example, Takagi discloses:

Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

When a supporter clicks on such an element, a dialog box appears to specify a heading level (h1-h6) with an optional text description. The same dialog box also has options for other types of metadata such as alternative text for an image that the system failed to recognize as inaccessible. Once metadata is created, an icon indicating the metadata type is shown at the position where the metadata was added (Figure 3 (b)). A supporter can edit and improve metadata by clicking on a metadata icon.

Id.

208. Additionally, Takagi also discloses a “quick fix wizard” which “allows a supporter to quickly fix all of the image-related accessibility problems in a webpage by simply filling in the fields and clicking on the “next” button.” *Id.*, at 197, col. 1, ¶5.

209. Takagi also describes that using “volunteer tools for metadata authoring: and a public server as the metadata repository. Both end users and volunteers (also called supporters) access the repository using client-side tools, which interact with the server through Web APIs. They can also access the repository by visiting the portal site with their Web browsers.” Ex. 1007, at 196, col. 1, ¶6. Takagi further discloses that “an end user reports an accessibility problem on a webpage. This report goes to the supporter community instead of to the site owner (as with a contact form). Then a supporter creates external metadata

Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

to fix the reported problem. As a result, the end user can browse the webpage made accessible with the metadata.” *Id.*

210. Takagi also discloses that the end user tools “allow expert users to create alternative text labels for the previously inaccessible elements based on the context.” Ex. 1007, at 196, col. 2, ¶4. A POSITA would have understood that previously inaccessible elements to disclose non-programmatically-fixable compliance issue associated with the web page that an “expert user” can manually remediate. *Id.*

211. Therefore, Springer and Takagi discloses “receiving, by the remote server system, via a remediation interface input from a user, the input comprising a remediation action to manually remediate a non-programmatically-fixable compliance issue associated with the web page” as claimed.

212. Therefore, Springer and Takagi thus disclose and render obvious this element.

- g. “generating, by the remote server system, the one or more pre-existing remediations based on the input comprising the remediation action to manually remediate a non-programmatically-fixable compliance issue; ” (Claim 1[c][4])**

213. Springer and Takagi disclose this element because Takagi discloses using a social accessibility pilot service to create external metadata (remediation

Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

code or action) that was manually remediated via social accessibility pilot service to remediate a non-programmatically-fixable compliance issue. Takagi also discloses that after a problem is reported by an end user, “a supporter creates external metadata to fix the reported problem. As a result, the end user can browse the webpage made accessible with the metadata.” Ex.1007, 196, col. 1, ¶6.

214. As shown in Annotated Figure 1 of Takagi below, and described in element 1[c][3] above, Takagi’s social accessibility pilot system consists of end-user tools, located in a client PC, supporter tools, located in another client PC, and an open repository, part of a social accessibility server. Ex. 1007, at Figure 1, 196, col. 1, ¶6. A POSITA would have understood that the Takagi’s social accessibility service to disclose a remote server. *Id.*

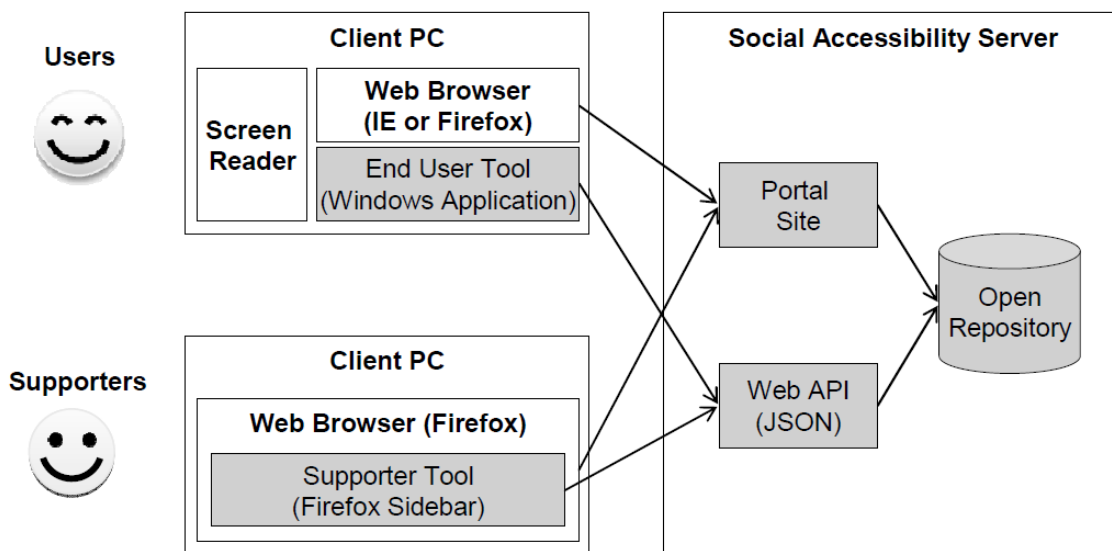


Figure 1. Basic Architecture of Social Accessibility Service

Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

Id., at Figure 1.

215. Takagi's social accessibility pilot system provides accessibility improvement by "[1][f]irst, an end user report[ing] an accessibility problem on a webpage. This report goes to the supporter community instead of to the site owner (as with a contact form). [2] Then a supporter creates external metadata to fix the reported problem. As a result, the end user can [3] browse the webpage made accessible with the metadata. This reporting-based strategy allows quick solutions to accessibility problems." Ex. 1007, at 196, col. 1, ¶6. Further, Takagi describes that "[t]he public repository allows that metadata to benefit every end user, not just the first user who reported the problem." *Id.* Takagi also discloses "create[ing] external metadata to fix the reported problem" (*id.*), and discloses "supporters created 18,296 metadata records in response to requests from the users." *Id.* Figure 2 of Takagi shows the process model used by the social accessibility pilot system including a "Social Accessibility Server":

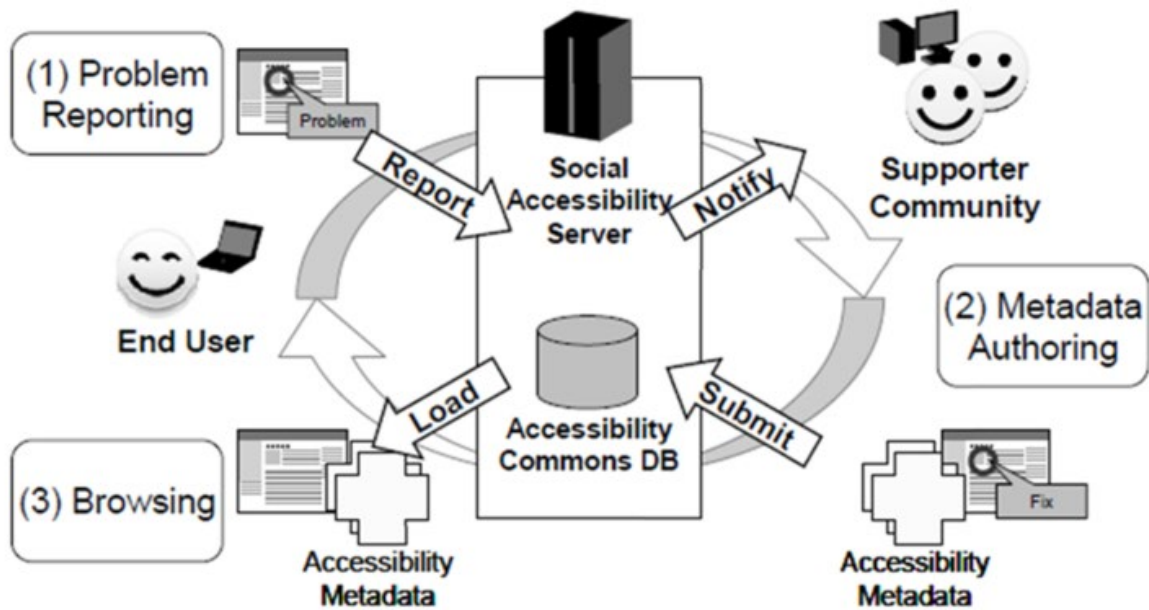


Figure 2. Current Process Model

Ex. 1007, at Figure 2.

216. As shown in Figure 2 of Takagi, and as described above, Takagi discloses a remote server, for creating accessibility metadata to fix a reported problem via a supporter, e.g., generating pre-existing remediations code based on the input comprising the remediation action to manually remediate a non-programmatically-fixable compliance issue. Ex. 1007, at 196, col. 1, ¶6, Figure 1, Figure 2.

217. Therefore, Springer and Takagi disclose and render obvious this element.

h. “storing, by the remote server system, the one or more pre-existing remediations in an electronic data storage medium; and” (Claim 1[c][5])

218. Springer and Takagi disclose this element. Takagi discloses a repository or database for storing supporter-authored metadata such as alt-text. Ex.1007, 196, 198. For example, Takagi discloses using a public server as the metadata (the one or more pre-existing remediations) repository. A POSITA would have understood that a public server used as a repository for metadata to disclose storing the one or more pre-existing remediations since the metadata stored by the accessibility server comprises alt-text. Ex. 1007, at 196, col. 1, ¶6, 198, col. 1, ¶1.

219. Further, Takagi discloses that “end-user tools [are used] for problem reporting and client-side webpage transcoding based on external metadata; volunteer tools [are used] for metadata authoring; and a public server [is used as] as the metadata repository. Both end users and volunteers (also called supporters) access the repository using client-side tools, which interact with the server through Web APIs.” Ex. 1007, at 196, col. 1, ¶6. Takagi also discloses that “most frequent metadata type was the “alttext.” [And] This metadata covers alternative texts for non-textual objects, labels.” Ex. 1007, at 198, col. 1, ¶1. Specifically, Figure 1 of Takagi clearly shows the social accessibility server including an open repository

which is a “public server,” e.g., an electronic data storage medium, that is used as the metadata repository. Ex. 1007, at 196, col. 1, ¶6, Figure 1. Annotated Figure 1 of Takagi shows the remote accessibility server including the electronic storage medium via the open repository, below:

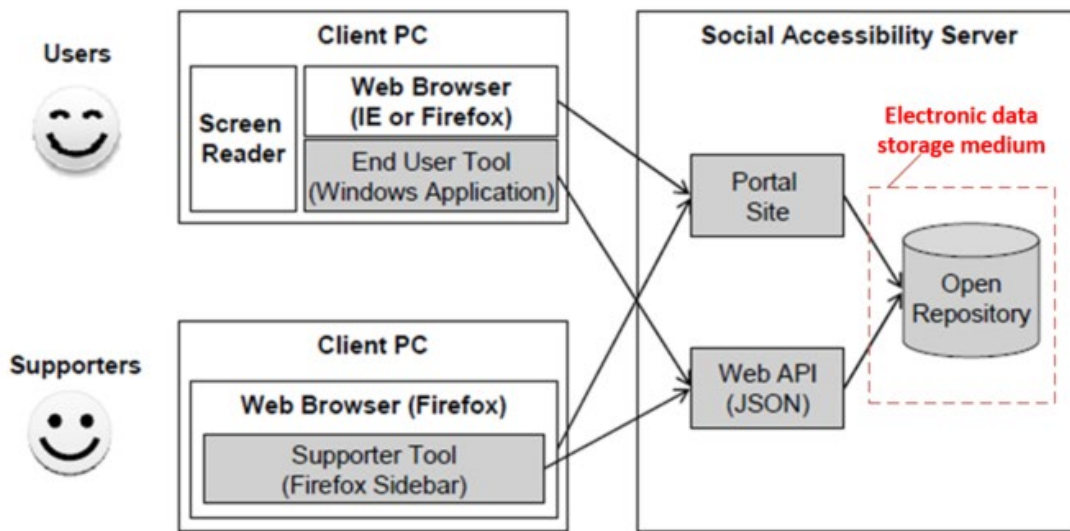


Figure 1. Basic Architecture of Social Accessibility Service

Ex. 1007, at Figure 1 (annotated).

220. Figure 2 of Takagi also clearly shows the social accessibility server including an “accessibility commons DB,” e.g., DB being a database, where a database is an electronic storage medium. Figure 2 of Takagi also shows that as part of the accessibility metadata authoring process, the accessibility metadata is submitted by the supporter community, e.g., stored, into the accessibility commons DB. Ex. 1007, at 196, col. 1, ¶6, Figure 2. Annotated Figure 2 below shows the

Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

social accessibility server, including the accessibility commons DB, e.g., an
electronic data storage medium:

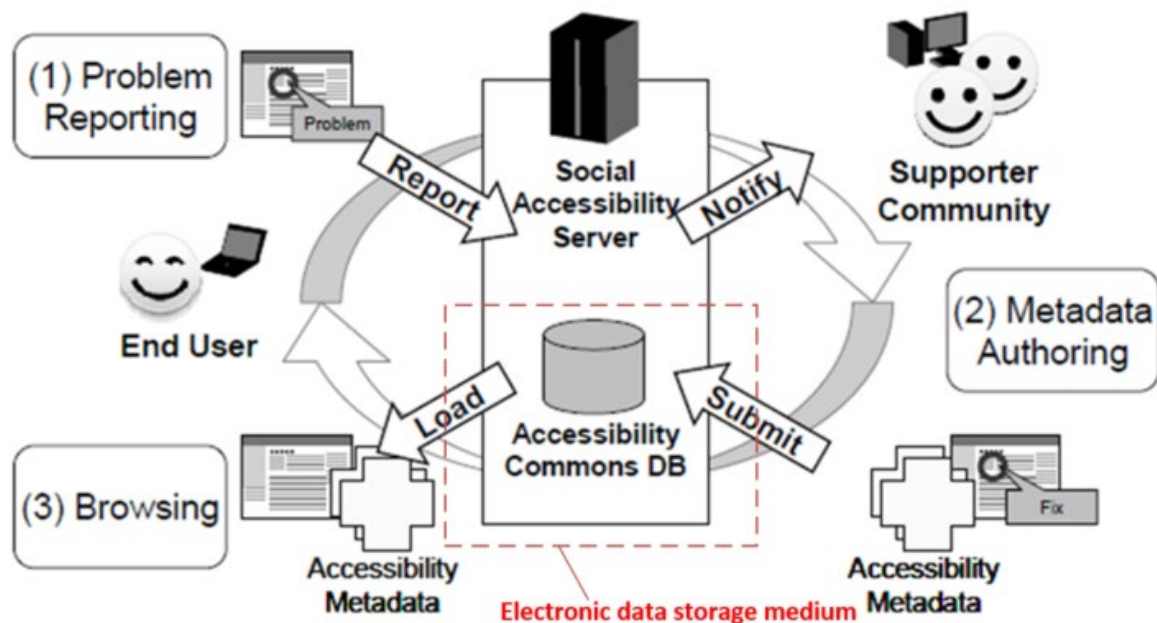


Figure 2. Current Process Model

Ex. 1007, at Figure 2 (annotated).

221. A POSITA would have understood that the metadata repository and/or database of Takagi, e.g., an electronic storage medium, stores the authored metadata (e.g., remediated code) submitted by the supporter community, therefore Takagi discloses this element. Ex. 1007, at 196, col. 1, ¶6, 198, col. 1, ¶1, Figure 1, Figure 2.

222. Therefore, Springer and Takagi thus disclose and renders obvious this element.

- i. **“assigning, by the computer system, a descriptive attribute to the untagged element in the code of the web page based on the one or more pre-existing remediations, the descriptive attribute assigned to the untagged element adapted to enable an assistive technology to speak the descriptive attribute to a user.” (Claim 1[d])**

223. Springer and Takagi disclose this element because Springer discloses using a computer system to assign a label to an input field or alt-text for an image lacking an appropriate label (a descriptive attribute to the untagged element) to enable an assistive technology to speak the descriptive attribute to a user. As mentioned above, Springer discloses that the system is intended to be used by “a user of an assistive technology that permits verbal communication of web content,” Ex. 1005, at ¶29. Furthermore, as described above, Springer discloses that a form label fixer can insert the piece of text it identified as the proper label (e.g., an untagged element lacking a descriptive attribute, such as an input field lacking an appropriate label) directly into the DOM (i.e., document tree) by assigning it to the appropriate input field. *Id.*, at ¶38 (“If such a textual element is found, the label will enclose the textual element in the appropriate markup and make the proper insertion into the document tree.”).

224. As described in claim element 1[c] above, Springer discloses its checkers, “ImageChecker” and “ImageMapChecker,” can detect images that have

Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

or lack valid alt text descriptions/attributes (i.e., an image lacking an adequate description). *Id.*, at ¶¶52, 15. Furthermore, Springer discloses fixers such as “ImageFixer” and “ImageMapFixer” that correspond to its checkers, where Springer uses those fixers to add an alt attribute to an image (e.g., using the ImageFixer) and to add alt attributes to AREAs which may be missing such attributes (e.g., via the ImageMapFixer). Ex. 1005, at ¶16, Table II(b).

225. A POSITA would have understood that Springer’s form label checker and form label fixer can be used to assign descriptive attributes to an appropriate form field (e.g., untagged element) to enable commonly used screen readers to provide an audible description of the said form field. Springer also discloses “assistive technology” that can provide an audible description based on the added alt attributes. In particular, Springer states that “assistive technology generally refers to software that enables an individual with a certain disability to interact with a computer.” *Id.*, at ¶5. As an example, Springer discloses “[a] screen **reader** is assistive technology that **reads the contents** of a computer screen to the user” (*id.* (emphasis added)), and thus provides an audible description.

226. As discussed in claim element 1[a] above Springer discloses a “compliance retrofitter,” a software that can be executed on a computer system, and also discloses the “compliance retrofitter may be platform independent by

Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

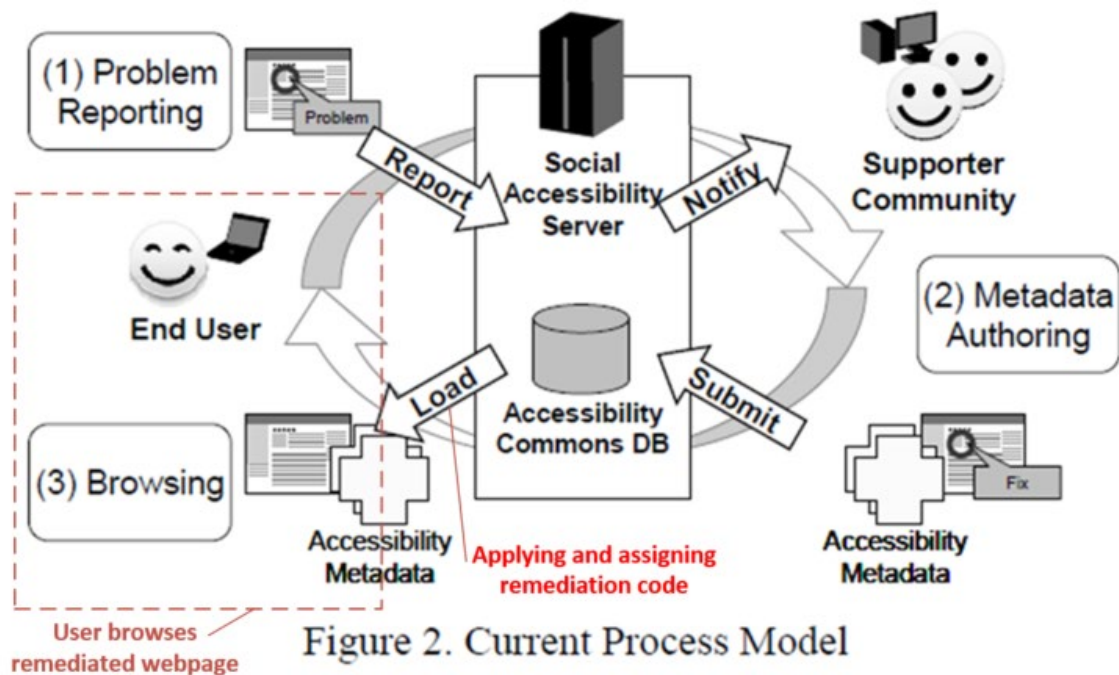
using a platform independent language such as Java (Sun Micro Systems, Inc. of Palo Alto, Calif.), and can be run on any platform for which a Java Virtual Machine exists, including Windows 95/98, Windows 2000, Windows NT and Linux.” *Id.*, at ¶¶7, 57.

227. Takagi also discloses this element because Takagi discloses using a social accessibility pilot system to allow a supporter to create external metadata (a descriptive attribute to an untagged element) corresponding to (assigned to) a reported accessibility problem on a webpage for use with a screen reader independent end user tool (adapted to enable an assistive technology to speak the descriptive attribute to a user). Takagi’s social accessibility pilot system provides accessibility improvement” by “[1][f]irst, an end user report[ing] an accessibility problem on a webpage. This report goes to the supporter community instead of to the site owner (as with a contact form). [2] Then a supporter creates external metadata to fix the reported problem. As a result, the end user can [3] browse the webpage **made accessible with the metadata**. This reporting-based strategy allows quick solutions to accessibility problems.” Ex. 1007, at 196, col. 1, ¶6 (emphasis added). Takagi describes that “[t]he public repository allows that metadata to benefit every end user, not just the first user who reported the problem.” *Id.* Takagi also discloses that its end user tool, which will be used by

Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

end users to display the webpage including the remediation is “independent of the screen reader.” *Id.*, at 196, col. 2, ¶3. A POSITA would have thus understood from this (and from Figure 1’s depiction of a “User” having a “Screen Reader” on his “Client PC”) that the sight-impaired user would utilize a screen reader, and that the descriptive attribute is adapted to enable any assistive technology to speak the attribute to the user.

228. Furthermore, Figure 2 of Takagi clearly shows the “load[ing]” of the accessibility metadata to a webpage, e.g., applying remediation code and assigning a descriptive attribute, to be browsed by the end user. Annotated Figure 2 of Takagi is shown below:



Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

Ex. 1007, at Figure 2 (annotated).

229. Therefore Springer and Takagi disclose “assigning, by the computer system, a descriptive attribute to the untagged element in the code of the web page based on the one or more pre-existing remediations, the descriptive attribute assigned to the untagged element adapted to enable an assistive technology to speak the descriptive attribute to a user” as claimed.

230. Therefore, Springer and Takagi disclose and renders obvious this element.

3. Claim 2

231. Claim 2 depends on claim 1 and recites “**wherein the one or more pre-existing remediations are javascript.**” Springer and Takagi disclose this claim. Springer discloses its code may be platform independent by using a platform independent language such as Java. By the time of the ’532 patent, Java had largely been replaced by JavaScript as the preferred platform-independent code for interacting with websites and code. It thus would have been obvious to a POSITA to use JavaScript code as Springer does not limit itself to any specific platform independent languages.

232. Takagi also explicitly discloses that in some embodiments, end-user tool that injects special JavaScript code into the webpages for the selected functions:

For that reason, we also developed a native application for Windows® as an alternative to the Jaws-based end-user tool. This resident utility shows a menu dialog when the user presses a short-cut key. The menu includes problem reporting and metadata loading. The resident utility works by injecting special JavaScript code into the webpages for the selected functions. This approach is independent of the screen reader and supports two major Web browsers, Microsoft Internet Explorer® and Mozilla Firefox®.

Ex. 1007, at 196, col. 2, ¶3.

233. Furthermore, Takagi discloses enabling site owners to integrate metadata into their web pages by providing “mechanism for applying metadata inside of webpages without any external tools” by “using a JavaScript library,” noting the approach “is already used by Google with AxsJAX technology” for improving accessibility. Ex. 1007, at 201, col. 2, ¶2.

234. Therefore, Springer and Takagi thus disclose and render this claim obvious.

4. Claim 3

235. Claim 3 depends on claim 1 and recites the additional requirement of: **“wherein the untagged element is an input field.”** Springer and Takagi disclose

Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

this claim because Springer teaches programmatically assigning a descriptive attribute to an input field on a web page, disclosing a “FormFixer” tool which “[a]dds label elements to forms” and “[e]xplicitly associates a LABEL element with the form control.” Ex. 1005, at Tables II(a)-(b). Also, Springer discloses that “[f]orm fields are a way for users to enter input into a web page, usually by filling in text boxes or selecting from drop-down menus. Form fields are HTML elements that allow data entry into a form. A common form field is the INPUT element.” Ex. 1005, at ¶37.

236. Furthermore, Springer discloses using the form label fixer with the <INPUT> element. Specifically, Springer discloses using the “form label fixer” to insert “a label element into the HTML” and “explicitly associate it with the form field” and provides an example of this: “<INPUT type=“text” name=“emailinput” size=“8” class=“NavBoldWhite” id=“Email”>.” *Id.* Therefore, a POSITA would have understood that Springer discloses “determining, by the computer system, one or more contextual cues associated with an untagged element, the untagged element being an input field” as claimed. *Id.*

237. Takagi also discloses this claim. In Takagi’s social labeling approach, the metadata covers alternative texts for non-textual objects, labels for input forms, and textual descriptions added to content. Ex. 1007, at 198, col. 1, ¶1. Specifically

Takagi discloses “metadata [that] covers alternative texts for non-textual objects, labels for input forms, and textual descriptions added to content. The script to apply the metadata directly modifies the DOM as appropriate for each target HTML element.” *Id.*

238. Therefore, Springer and Takagi thus disclose and render this claim obvious.

5. Claim 4

239. Claim 4 depends on claim 1 and recites the additional requirement of: **“wherein the untagged element is an image.”** Springer and Takagi disclose this claim. Springer teaches using a series of tools that check for, and then fix, accessibility compliance issues for web pages such as fixing missing textual description for untagged elements (e.g., images and AREA elements lacking appropriate alt text). Springer describes checker tools which include an ImageChecker that can be used to check “that images have valid alt attributes” and “that long descriptions are valid.” Ex. 1005, at Table I(b). The checker tools of Springer also include an ImageMapChecker that checks that AREAs (e.g., elements which define a part of an image that functions as a link) have valid alt attributes.

Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

240. For every checker tool in Springer, there is a corresponding fixer tool, which fixes the compliance issues that the checker tool identified. The fixer tools include an ImageFixer and ImageMapFixer. ImageFixer and ImageMapFixer remediate compliance issues stemming from websites lacking a text equivalent for every non-text element such as an image (e.g., images as untagged elements). *Id.*, at Tables II(a)-(b). ImageFixer remediates this issue, for example, by adding alt attributes to images, image buttons, and image links, as well as replacing alt attributes with more appropriate (e.g., in length) or meaningful descriptions. *Id.* ImageMapFixer does the same thing for AREAS (described above). *Id.*

241. Similarly, Takagi discloses this claim. For example, Takagi discloses a “native application also added an advanced feature to handle elements ignored and inaccessible with earlier approaches, which include images with no alternative text.” Ex. 1007, at 196, col. 2, ¶4. Takagi also discloses its WYSISYG-style editor for metadata authoring that “creates an image of the webpage” and that “[t]he created image adds exclamation marks for such metadata targets as inaccessible images to help the supporters add useful metadata (Figure 3 (a)). Inaccessible images include those without alternative text or with useless alternative text (such as “banner” or “spacer”). Also marked are image links pointing to the same destinations as nearby text links.” Ex. 1007, at 197, col. 1, ¶4.

Takagi also discloses a quick fix wizard editor tool “sequentially displays each of the inaccessible images. Each image is followed by the description and the same form for new metadata. This wizard allows a supporter to quickly fix all of the image-related accessibility problems in a webpage by simply filling in the fields and clicking on the “next” button.” Ex. 1007, at 197, col. 1, ¶5.

242. Springer and Takagi thus disclose and render this claim obvious.

6. Claim 6

243. Claim 6 depends on claim 1 and recites the additional requirement of:
“wherein the untagged element lacks the adequate descriptive attribute when one or more descriptive attributes of the untagged element are erroneous.”

Springer and Takagi disclose this claim because Springer teaches using a computer system to detect one or more untagged elements such as an input field lacking an adequate descriptive attribute (e.g., more compliance issues) relating to web accessibility standards in the code. Specifically, Springer discloses a “FormChecker” tool that detects form fields that lack a descriptive label (e.g., an untagged element such as an INPUT field lacking an adequate descriptive attribute). Springer also discloses “form fields are HTML elements that allow data entry into a form” and that “[a] common form field is the INPUT element.” Ex. 1005, at ¶37. Springer further explicitly discloses that its checkers, such as

Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

FormChecker, “[c]hecks that labels are explicitly associated with form controls.”

(*id.*, at Tables II(a)-II(b)). Each of these compliance issues are related to web accessibility compliance standards. *Id.*, at ¶¶37-38. Thus, Springer discloses “detecting, by the computer system, one or more compliance issues relating to web accessibility standards in the code, wherein at least one of the one or more compliance issues comprises an untagged element, the untagged element is an element lacking an adequate descriptive attribute” as claimed.

244. Springer further teaches that “[t]he compliance retrofitter may use a system of customizable parameters or “tolerances” to increase the accuracy of automated accessibility checking.” Specifically, tolerances are used to increase the accuracy of tests by decreasing the number of misdiagnoses and false positives, where misdiagnoses occur whenever a testing tool fails to mark a violating HTML element as a violation and false positives refer to an HTML element that is inaccurately marked as a violation. Tolerances can be edited to customize the tests for an individual website's look and feel, and are stored in an external XML file that is machine readable and can be edited by the user using a GUI. This feature allows the user to alter the configuration of the tools' tests and customize the tests for a particular client or website.” Ex. 1005, at ¶50.

Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

245. Springer also discloses, for example, that its ImageFixer replaces alt attributes that are too long, vague, invalid, or not meaningful. Ex. 1005, Table II(b).

246. Takagi also discloses this claim. Specifically, Takagi discloses a “page map” which is a WYSISYG-style editor for metadata authoring. Ex. 1007, at 196, col. 2, ¶4. Takagi’s page map editor “creates an image of the webpage and allows supporters to quickly choose a webpage element for external metadata by just clicking on that element. The created image adds exclamation marks for such metadata targets as inaccessible images to help the supporters add useful metadata (Figure 3 (a)). Inaccessible images include those without alternative text or **with useless alternative text (such as “banner” or “spacer”).**” Ex. 1007, at 196, col. 2, ¶4 (emphasis added). A POSITA would have understood that a “webpage element for external metadata” that includes “inaccessible images [which] include those without alternative text or with useless alternative text” to disclose an untagged element, e.g., an image, that lacks the adequate descriptive attribute, e.g., lacking alternative text, when one or more descriptive attributes of the untagged element are erroneous, e.g., an image with useless alternative text.

247. Therefore, Springer and Takagi thus disclose and render this claim obvious.

7. Claim 7

248. Claim 7 depends on claim 1 and recites the additional requirement of:

“wherein the assigning comprises changing the DOM or HTML code associated with the web page.” Springer and Takagi disclose this claim because Springer discloses using a computer system to access the data model associated with a web page (e.g., the DOM) to check for accessibility compliance issues. Specifically, Springer discloses a “compliance retrofitter,” a software, that can be executed on a computer system for checking for compliance “an HTML document of a web page” where the checking includes using “checkers to check a data model of a web page for accessibility.” Ex. 1005, at ¶7, Abstract. Springer also states that “[t]he checkers implemented by the compliance retrofitter are sets of tests that are run against documents, for example, files or web pages, to analyze compliance with a standard, e.g., Section 508.” Ex. 1005, at ¶14. Springer discloses that it interacts with webpages through a data model, “document tree,” and “tree format.” *Id.*, at ¶¶15, 25, 38, 49. As explained in the above element, the data model and “document tree” is a software-accessible format known as a document object model or DOM.

249. Takagi also discloses this claim. In Takagi’s social labeling approach, the metadata covers alternative texts for non-textual objects, labels for input forms,

and textual descriptions added to content. Ex. 1007, at 198, col. 1, ¶1. Specifically Takagi discloses “metadata [that] covers alternative texts for non-textual objects, labels for input forms, and textual descriptions added to content. The script to apply the metadata directly modifies the DOM as appropriate for each target HTML element.” *Id.*

250. Therefore, Springer and Takagi thus disclose and render this claim obvious.

8. Claim 8

251. Claim 8 depends on claim 1 and recites the additional requirement of: “**wherein the descriptive attribute is an HTML label.**” Springer and Takagi disclose this claim because Springer teaches “a method that may include a computer program product that implements the steps of identifying for a form field not associated with a label, a piece of text that is a candidate for the label; and associating the piece of text with the label or prompting a user to select the candidate label or insert a label. In one aspect, identifying a piece of text that is a candidate for the label may include identifying a piece of text that is not a child of any of “a,” “applet,” “script,” “noscript,” “select,” “object,” “head,” or “label;” and has a predetermined text length.” Ex. 1005, at ¶9.

Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

252. Springer further teaches that form fields are a way for users to enter input into a web page, which are HTML elements that allow data entry into a form. A common form field is the INPUT element, which can be rendered as a checkbox, text entry, radio button, password field, or button. Form fields in violation of paragraph (n) are those form fields not explicitly associated with labels. Springer states that “in order to be accessible to people with disabilities, form fields must have either “label” attributes or LABEL elements that are explicitly associated with them. These labels enable users of assistive technologies to understand the type of input that must be entered into the form field.” *Id.*, at ¶37.

253. Springer also discloses “fixers” corresponding to each “checker.” When a specific checker finds a compliance issue, it selects that issue to the corresponding fixer, and that specific fixer is used for remediation. As described above, the Springer discloses a FormFixer tool which “[a]dds label elements to forms” and “[e]xplicitly associates a LABEL element with the form control.” *See* Ex. 1005, at ¶10, Tables II(a)-(b) (emphasis added).

254. In addition Takagi discloses this claim because Takagi’s metadata includes “labels for input forms,” a POSITA would have understood a label for an input form to include an HTML label. Ex. 1007, at 198, col. 1, ¶1. Takagi discloses “metadata [that] covers alternative texts for non-textual objects, labels for

input forms, and textual descriptions added to content. The script to apply the metadata directly modifies the DOM as appropriate for each target HTML element.” *Id.*

255. Therefore, Springer and Takagi thus disclose and render this claim obvious.

9. Claim 9

256. Claim 9 depends on claim 1 and recites the additional requirement of: **“wherein the descriptive attribute is an alt-text label.”** Springer and Takagi disclose this claim because Springer disclose that “[t]he compliance checker algorithm finds the HTML elements that are missing alt attribute in a selected instance. Below HTML window 40 is fixer window 50 that displays information related to fixers, or algorithms that make changes to HTML documents or other documents to bring the documents into compliance with the standard, e.g., § 1194.22.” A list of § 1194.22 fixers are listed in Table 11(a) along with the relevant standard and manual fixes (if any). The functionality of each fixer is explained in Table 11(b). As can be seen, the ImageMapFixer adds alt attributes to AREA elements that currently do not have valid alt attributes. Ex. 1005, at ¶16.

Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

36 CFR § 1194.22 Paragraph	Fixer	Functionality
A	ImageFixer ObjectTextFixer AppletFixer	Adds alt attributes to images Adds alt attributes to image buttons Adds alt attributes to image links Replaces alt attributes that are too long with shorter ones Replaces vague alt attributes on images with more meaningful ones Replaces empty or invalid longdesc descriptions with valid ones Replaces vague alt attributes on image buttons with more meaningful ones Adds a valid textual description to OBJECT elements Adds textual alternatives to APPLET elements
C	ColorContrastFixer	Allows developer to select a new color which has sufficient color contrast
F	ImageMapFixer	Adds alt attributes to AREAs Replaces vague alt attributes in AREAs with more meaningful ones

Id., at Table 11(b).

257. Takagi discloses this claim. Takagi discloses using its WYSISYG-style editor, referred to as a “page map,” for metadata authoring that allows supports to add “add useful metadata” to inaccessible images. Ex. 1007, at 197, col. 1, ¶4. Takagi discloses that the “[i]naccessible images include those without alternative text or with useless alternative text (such as “banner” or “spacer”).” *Id.* Takagi also discloses that its WYSISYG-style editor “has options for other types of metadata such as **alternative text for an image** that the system failed to recognize as inaccessible. Once metadata is created, an icon indicating the metadata type is shown at the position where the metadata was added (Figure 3 (b)). A supporter can edit and improve metadata by clicking on a metadata icon.” *Id.* (emphasis added). Figure 3 of Takagi shows where metadata, such as

Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

alternative text for an image, was added for labeling. Annotated Figure 3 of
Takagi is shown below:

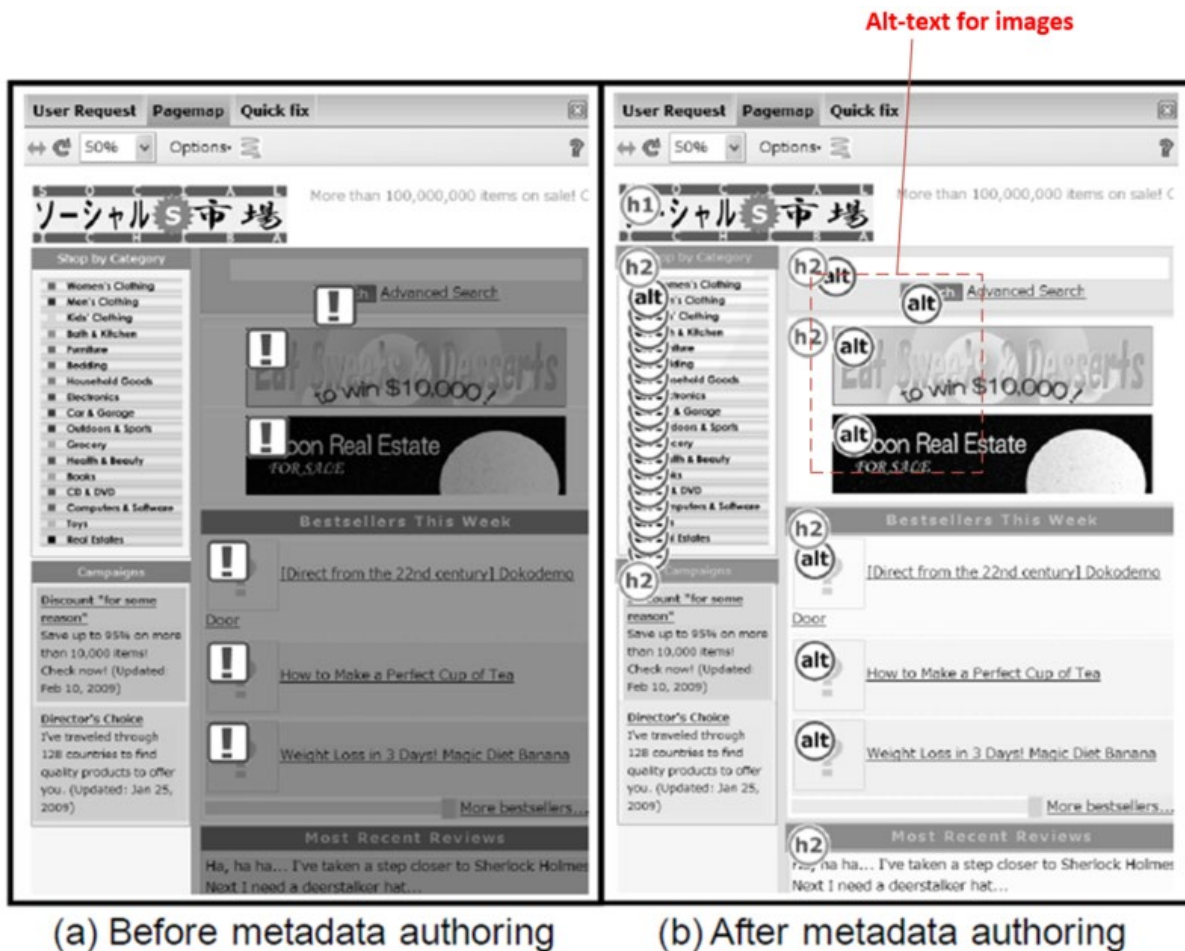


Figure 3. Page Map

Ex. 1007, at Figure 3 (annotated).

258. Takagi also discloses “wherein the descriptive attribute is an alt-text label.” Specifically, Takagi teaches a native application that has an advanced feature to handle elements ignored and inaccessible with earlier approaches. Ex. 1007, at 196, col. 2, ¶4. These elements include images with no alt text. *Id.* The

built-in menus and submenus of this native application enables expert users to create alt-text labels for the previously inaccessible elements based on the context.

Id.

259. Furthermore Takagi discloses “metadata [that] covers alternative texts for non-textual objects, labels for input forms, and textual descriptions added to content. The script to apply the metadata directly modifies the DOM as appropriate for each target HTML element.” Ex. 1007, at 198, col. 1, ¶1. Thus Takagi also discloses this claim.

260. Therefore, Springer and Takagi thus disclose and render this claim obvious.

10. Claim 10

261. Claim 10 depends on claim 1 and recites the additional requirement of: “**wherein the computer system comprises one or more computing systems.**” Springer and Takagi disclose this claim because Springer disclose “compliance retrofitter may be platform independent by using a platform independent language such as Java (Sun Micro Systems, Inc. of Palo Alto, Calif.), and can be run on any platform for which a Java Virtual Machine exists, including Windows 95/98, Windows 2000, Windows NT and Linux.” Ex. 1005, at ¶57. A POSITA would have understood that one or more computer systems to include “any platform for

which a Java Virtual Machine exists, including Windows 95/98, Windows 2000, Windows NT and Linux” as Springer discloses. *Id.*

262. Takagi also discloses this claim because Takagi’s Social Accessibility Pilot Service includes at least one computer system including a client PC and a social accessibility server, as clearly shown in Takagi Figure 1 (Ex. 1007, at Figure 1). Figure 1 of Takagi is shown below:

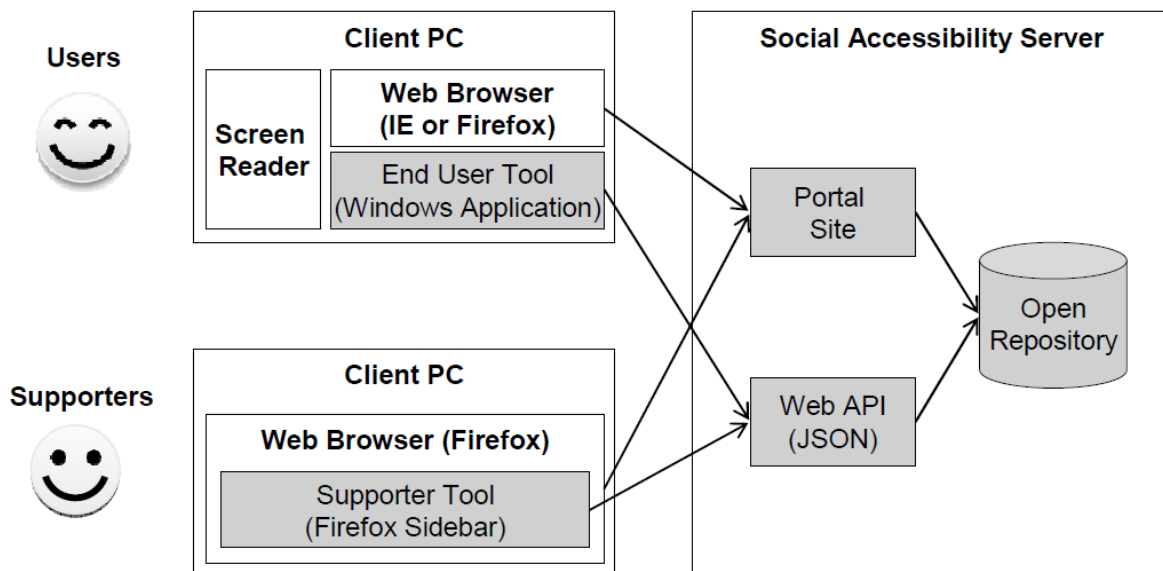


Figure 1. Basic Architecture of Social Accessibility Service

Ex. 1007, at Figure 1.

263. A POSITA would have understood that a Social Accessibility Server discloses one or more computer systems.

264. Therefore, Springer and Takagi thus disclose and render this claim obvious.

11. Independent Claim 11

- a. “A non-transitory electronic storage medium with computer code stored thereon, the computer code configured to programmatically assign a descriptive attribute to an element on a web page to enable an audible description of the element, the web page having an associated document object model (DOM), and the computer code configured to perform, when executed by a computer processor, the steps of:”
(Claim 11[pre])**

265. To the extent the preamble is limiting, Springer and Takagi disclose it.

Springer and Takagi disclose this element for the reasons described above for Ground 1, Claim 1[pre]. A POSITA would have understood that a computer implemented method (as claimed in claim 1) was routinely performed through computer code stored on a non-transitory electronic medium. Additionally, the recital of a “non-transitory computer readable information storage media” was well-known in the art and would have been obvious to a POSITA. Furthermore, a POSITA would have understood that a descriptive attribute to an element on a web page to include an untagged element as described in claim 1[pre].

266. Springer and Takagi thus disclose and render obvious the preamble.

- b. “dynamically analyzing, by a computer system, code associated with the web page, the code comprising at least the DOM or HTML code” (Claim 11[a])**

Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

267. Springer and Takagi disclose this element for the reasons described above with respect to Ground 1, Claim element 1[a]. Accordingly, Springer and Takagi thus disclose or render this element obvious.

- c. **“detecting, by the computer system, one or more compliance issues relating to web accessibility standards in the code, wherein at least one of the one or more compliance issues comprises the element lacking an adequate descriptive attribute;” (Claim 11[b])**

268. Springer and Takagi disclose this element for the reasons described above with respect to Ground 1, Claim element 1[b]. Accordingly, Springer and Takagi thus disclose or render this element obvious.

- d. **“applying, by the computer system, one or more pre-existing remediations to the one or more compliance issues, wherein the one or more pre-existing remediations is generated by a remote server system performing at least:” (Claim 11[c][1])**

269. Springer and Takagi disclose this element for the reasons described above with respect to Ground 1, Claim element 1[c][1]. Accordingly, Springer and Takagi thus disclose or render this element obvious.

- e. **“accessing the code associated with the web page;” (Claim 11[c][2])**

270. Springer and Takagi in disclose this element for the reasons described above with respect to Ground 1, Claim element 1[c][2]. Accordingly, Springer and Takagi thus disclose or render this element obvious.

- f. **“receiving, by the remote server system, via a remediation interface input from a user, the input comprising a remediation action to manually remediate a non-programmatically-fixable compliance issue associated with the web page;” (Claim 11[c][3])**

271. Springer and Takagi disclose this element for the reasons described above with respect to Ground 1, Claim element 1[c][3]. Accordingly, Springer and Takagi thus disclose or render this element obvious.

- g. **“generating, by the remote server system, the one or more pre-existing remediations based on the input comprising the remediation action to manually remediate a non-programmatically-fixable compliance issue;” (Claim 11[c][4])**

272. Springer and Takagi disclose this element for the reasons described above with respect to Ground 1, Claim element 1[c][4]. Accordingly, Springer and Takagi thus disclose or render this element obvious.

- h. **“storing, by the remote server system, the one or more pre-existing remediations in an electronic data storage medium; and” (Claim 11[c][5])**

273. Springer and Takagi disclose this element for the reasons described above with respect to Ground 1, Claim element 1[c][5]. Accordingly, Springer and Takagi thus disclose or render this element obvious.

- i. **“assigning, by the computer system, a descriptive attribute to the element in the code of the web page based on the one or more pre-existing remediations, the descriptive attribute assigned to the element**

**adapted to enable an assistive technology to speak the
descriptive attribute to a user.” (Claim 11[d])**

274. Springer and Takagi disclose this element for the reasons described above with respect to Ground 1, Claim element 1[d]. Accordingly, Springer and Takagi thus disclose or render this element obvious.

12. Claim 12

275. Claim 12 depends on claim 11 and recites the additional requirement of: **“wherein the one or more pre-existing remediations are javascript.”** Springer and Takagi disclose this claim for the reasons described above with respect to Ground 1, Claim 2. Accordingly, Springer and Takagi thus disclose or render this claim obvious.

13. Claim 13

276. Claim 13 depends on claim 11 and recites the additional requirement of: **“wherein the element is an input field.”** Springer and Takagi disclose this claim for the reasons described above with respect to Ground 1, Claim 3. Accordingly, Springer and Takagi thus disclose or render this claim obvious.

14. Claim 14

277. Claim 14 depends on claim 11 and recites the additional requirement of: **“wherein the element is an image.”** Springer and Takagi disclose this claim

Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

for the reasons described above with respect to Ground 1, Claim 4. Accordingly, Springer and Takagi in thus disclose or render this claim obvious.

15. Claim 16

278. Claim 16 depends on claim 11 and recites the additional requirement of: **“wherein the element lacks the adequate descriptive attribute when one or more descriptive attributes of the element are erroneous.”** Springer and Takagi disclose this claim for the reasons described above with respect to Ground 1, Claim 6. Accordingly, Springer and Takagi thus disclose or render this claim obvious.

16. Claim 17

279. Claim 17 depends on claim 11 and recites the additional requirement of: **“wherein the assigning comprises changing the DOM or HTML code associated with the web page.”** Springer and Takagi disclose this claim for the reasons described above with respect to Ground 1, Claim 7. Accordingly, Springer and Takagi thus disclose or render this claim obvious.

17. Claim 18

280. Claim 18 depends on claim 11 and recites the additional requirement of: **“wherein the descriptive attribute is an HTML label.”** Springer and Takagi disclose this claim for the reasons described above with respect to Ground 1, Claim 8. Accordingly, Springer and Takagi thus disclose or render this claim obvious.

18. Claim 19

281. Claim 19 depends on claim 11 and recites the additional requirement of: **“wherein the descriptive attribute is an alt-text label.”** Springer and Takagi disclose this claim for the reasons described above with respect to Ground 1, Claim 9. Accordingly, Springer and Takagi thus disclose or render this claim obvious.

19. Claim 20

282. Claim 20 depends on claim 11 and recites the additional requirement of: **“wherein the computer system comprises one or more computing systems.”** Springer and Takagi disclose this claim for the reasons described above with respect to Ground 1, Claim 20. Accordingly, Springer and Takagi thus disclose or render this claim obvious.

C. GROUND 2: Claims 1-4, 6-14, 16-20 are Unpatentable as Obvious Over Springer and Takagi in view of Hendry.

1. Motivation to Combine

283. A POSITA would have been motivated to combine Springer, and Takagi in view of Hendry in order to improve upon Springer’s compliance retrofitter functions for determining and assigning descriptive attributes for untagged elements, e.g., such as labels for untagged input fields and images lacking alt-text. Springer provides the various tools for checking and remediating web pages, but does not take advantage of collaborative metadata authoring and

Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

the powerful functions of a remote server that were commonplace by the time of the '532 patent.

284. A POSITA would have been motivated to combine Springer and Takagi for the reasons described above with respect to Ground 1.

285. Similar to combining Springer with Takagi, a POSITA would have been motivated to combine Hendry's server machine functionalities in fixing the Web accessibility issues with the systems of Springer and Takagi. In general, Hendry teaches systems and methods for modifying documents (e.g., web pages; *see* Ex. 1009, at ¶42) to improve document accessibility implemented on a client machine or server machine. Ex. 1009, at ¶59. For example, Hendry teaches that its methods can be performed on: (1) a local client computer, (2) a remote server, or (3) a combination of the two. *Id.* Hendry also teaches that its "[e]xamples [that] refer to a specific machine performing the steps [are merely] for the purposes of clarity and understanding" and that any step can be performed on any machine. *Id.* Hendry teaches that the various steps recited can be repeated, omitted, or performed in different order. *Id.*

286. Specifically, Hendry discloses that its "document modifier (120) is executed on the server machine (110) in response to receiving a request for a document from the client machine (102)" and that the "document modifier (120)

Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

on the server machine (110) determines that the particular requested document (114) is not accessibility enabled during an analysis of the particular requested document (114). The document (114) is analyzed to identify tags that are not recognizable by assistive technologies (106). The server machine (110) executes the document modifier (120) to produce a modified document (116) which is an accessibility-enabled version of the particular requested document (114). In response to receiving a request from the client machine (102) for the particular document (114), the server machine (110) sends the modified document (116).”

Ex. 1009, at ¶53. Hendry further discloses that “[t]he server machine (110) includes a data repository (112) storing one or more of: documents (114), modified documents (116) and accessibility features (118). The data repository (112) corresponds to any data storage device (for example, local memory on the server machine (110), web servers connected over the internet, machines within a local area network, a memory on a mobile device, local memory on the client machine (102), etc.).” *Id.*, at ¶41. Further Hendry discloses “[t]hose skilled in the art will appreciate that elements or various portions of data stored in the data repository (112) may be distributed (for example, stored on client machine (102) or at other servers)” where “the data repository (112) includes [a] flat, hierarchical, network

Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

based, relational, dimensional, object modeled, or data files structured otherwise”
and “is maintained as a table of a SQL database.” *Id.*, at ¶41.

287. A POSITA would also have expected this combination to succeed with minimal modification. Indeed, Springer already includes a form fixer tool that analyzes context of a web page to find and extract an appropriate label for a web element, an image fixer for adding alt attributes to images and an image map fixer for “adding alt attributes to AREAs. Ex. 1005 at Tables I(a)-(b), II(a)-(b), ¶¶16, 37-38. Takagi teaches that label extraction can be improved using an input label from a public repository authored by a volunteer, while Hendry teaches modifying a HTML document on the server machine. Ex. 1007, at 196, col. 2, ¶6; Ex. 1009, at ¶15. A POSITA could simply modify Springer’s “fixer” code to also query a database for matching descriptive attributes that had been authored by others. A POSITA could also add an interface (e.g., a quick fix wizard in Takagi) that allows supporters to respond to users’ requests and quickly make alternative text metadata via a remote server with minimal work and the rest of Springer system, such as its FormChecker tool, would work as intended (with improved results from using the Takagi social labeling methods). Ex. 1007, at 197, col. 1, ¶¶3, 5; Ex. 1005, at Tables I(a)-(b), III.

Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

288. Takagi also discloses WYSISYG-style editor, referred to as a “page map,” for metadata authoring but does not elaborate on the details of how untagged elements, e.g., inaccessible images include those without alternative text or with useless alternative text (such as “banner” or “spacer”), are detected. Ex. 1007, at 197, col. 1, ¶4. Thus a POSITA would have understood the benefit of combining the Takagi with the Springer system which discloses how to verify compliance accessibility standards and automatically or manually retrofit the HTML of web pages to ensure compliance. Ex. 1005, at ¶¶8, 15-16, 38. Similar remediation techniques were well known. *See also* Ex. 1015, at ¶¶2, 4, 15, 18; Ex. 1006, at ¶72.

289. Furthermore, a POSITA would have understood that collaborative web accessibility improvements leveraged by the participation of an open community were well known by the time of the '532 patent. Ex. 1007, at 196, col. 1, ¶3. For example, as discussed above in the Background section, various types of social approaches other than Takagi, such as the 1997 ALT-server, Asakawa, WebInSight (Bigham), Teraguchi, and WebVisum, could provide an improvement over the existing automated accessibility enhancements described by Springer. Ex. 1007, 196, col. 1, ¶3; Ex. 1031, at 2-4; Ex. 1032, at 177, col. 2, ¶¶3-4, Figure 4; Ex. 1033; Ex. 1034; Ex. 1035, 1:18-36, Figure 1.

290. Additionally, as I explained in Section IV, it was a commonly known approach to store alternative text in a remote server, where the alternative text was generated with the help manual remediation. Ex. 1008, at 186, col. 2 ¶¶3-5, Figure 4; Ex. 1007, at 186, col. 1, ¶6; *see also* Ex. 1031, at 2-4; Ex. 1032, at 177, col. 2, ¶¶3-4, Figure 4; Ex. 1033; Ex. 1034; Ex. 1035, 1:18-36, Figure 1.

291. A POSITA would have been motivated to further add Hendry in order to combine Hendry's flexible distribution of accessibility remediation functionality with the systems of Springer and Takagi. Therefore, it would have been obvious to a POSITA to combine Springer, which discloses a heuristic method for checking an HTML document of a web page for compliance by running a checker and fixer tool to remediate a section of the HTML document containing the flagged violation by modifying HTML code, with Takagi and Hendry, both of which disclose inputting alternative text from a remote server machine.

2. Independent Claim 1

- a. **“A computer-implemented method of programmatically assigning a descriptive attribute to an untagged element on a web page to enable an audible description of the untagged element, the web page having an associated document object model (DOM), the method comprising:” (Claim 1[pre])**

Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

292. To the extent the preamble is limiting, Springer and Takagi in view of Hendry disclose it. Springer and Takagi disclose this claim for the reasons described above with respect to Ground 1, claim element 1[pre].

293. Hendry also discloses the preamble. Hendry discloses “automatically modifying documents to improve document accessibility.” Ex. 1009, at ¶12. Hendry also discloses “dynamically modifying any set of documents (for example, obtained over a network, a local machine, a server, a peer machine, within a software application, etc.).” *Id.*, at ¶15. Hendry also discloses techniques for automatically adding “accessibility features” to the said documents to improve the accessibility of the documents,” and that “examples of accessibility features [] include, but are not limited to, text alternatives for non-text content.” *Id.*, at ¶¶17, 24.

294. Hendry further discloses these “[d]ocuments []include, but are not limited to, pdfs, [and] web pages.” *Id.*, at ¶42. Hendry also discloses analyzing “HTML documents.” *Id.*, at ¶60. Hendry discloses that the HTML documents are defined by a Document Object Model (DOM). *Id.*, at ¶65. Specifically, Hendry discloses that “a model of a HTML or XML document is defined using a Document Object Model (DOM)” where “a HTML table from a HTML document is used to generate a table object that represents the HTML table.” *Id.*, at ¶65.

Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

Additionally, Hendry also discloses that “[m]odifying a document to improve document accessibility refers to modifications which result in proper or better interpretation by an assistive technology” and that “[e]xamples of assistive technologies include, but are not limited to screen readers, [and] text to speech software.” *Id.*, at ¶¶12, 3.

295. Therefore, Hendry discloses automatically modifying documents to improve document accessibility by adding text alternatives for non-text content (programmatically assigning a descriptive attribute to an untagged element on a web page), providing modifications which result in proper or better interpretation by an assistive technology (to enable an audible description of the untagged element), the document including a webpage or a HTML document defined by a Document Object Model (the web page having an associated document object model (DOM)). Ex. 1009, at ¶¶, 3, 12, 15, 17, 24, 42, 60, 65. Thus Hendry clearly discloses “programmatically assigning a descriptive attribute to an untagged element on a web page to enable an audible description of the untagged element, the web page having an associated document object model (DOM), the method comprising” as claimed.

296. Accordingly, Springer and Takagi in view of Hendry disclose and render obvious the preamble of claim 1.

b. “dynamically analyzing, by a computer system, code associated with the web page, the code comprising at least the DOM or HTML code;” (Claim 1[a])

297. Springer and Takagi in view of Hendry disclose this element.

Springer and Takagi disclose this element for the reasons described above with respect to Ground 1, claim element 1[a].

298. Furthermore, Hendry discloses this element. Hendry describes identifying accessibility compliance issues in a document defined using the DOM, such as untagged input fields, and automatically remediating them. Ex. 1009, at ¶¶12, 28. Hendry specifically discloses “automatically modifying documents to improve document accessibility” and “dynamically modifying any set of documents (for example, obtained over a network, a local machine, a server, a peer machine, within a software application, etc.)” *Id.*, at ¶¶12, 15. Hendry further discloses these “[d]ocuments [i]nclude, but are not limited to, pdfs, [and] web pages.” *Id.*, at ¶42. Hendry also discloses analyzing “HTML documents.” *Id.*, at ¶¶ 58, 60. Hendry also discloses that “a model of a HTML or XML document is defined using a Document Object Model (DOM)” where “a HTML table from a HTML document is used to generate a table object that represents the HTML table.” *Id.*, at ¶65. Therefore, Hendry clearly discloses a document that is defined using the DOM, e.g., the code of the document comprising the DOM. *Id.*

Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

299. Hendry further discloses embodiments where the remediation software is stored on a remote computer (e.g., a computer system such as a server), and where a remediation request is received from a user's browser (i.e., user computer) over the internet (i.e., an electronic network). *Id.*, at ¶58 (“one document modifier (120) is implemented as a server plugin, launched by server (110), to handle an HTTP request from the client machine (102) for a document (114).”) Also see claim 1[pre]. Thus, Hendry also discloses “dynamically analyzing, by a computer system, a code associated with the web page, the code comprising at least HTML or the DOM” as claimed.

300. Therefore, Springer and Takagi in view of Hendry thus disclose and render obvious this element.

- c. **“detecting, by the computer system, one or more compliance issues relating to web accessibility standards in the code, wherein at least one of the one or more compliance issues comprises the untagged element lacking an adequate descriptive attribute;” (Claim 1[b])**

301. Springer and Takagi in view of Hendry disclose this element. Springer and Takagi disclose this element for the reasons described above with respect to Ground 1, claim element 1[b].

302. Hendry also discloses this element because Hendry describes identifying accessibility compliance issues, such as untagged input fields, and

Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

automatically remediating them. Ex. 1009, at ¶¶12, 28. Hendry discloses a system for analyzing a document (e.g., webpages; *see* Ex. 1009, at ¶42) “to identify inaccessible features.” Ex. 1006, at ¶60. Hendry discloses that an “inaccessible feature corresponds to one or more elements that lack a label or a description that can be accessed by an assistive technology.” *Id.*, at ¶28. Hendry thus discloses “detecting, by the computer system, one or more compliance issues relating to web accessibility standards in the code, wherein at least one of the one or more compliance issues comprises an untagged element, the untagged element is an element lacking an adequate descriptive attribute” as claimed.

303. Therefore, Springer and Hendry in view of Takagi thus disclose and render obvious this element.

- d. **“applying, by the computer system, one or more pre-existing remediation code to the one or more compliance issues, wherein the one or more pre-existing remediations is generated by a remote server system performing at least:” (Claim 1[c][1])**

304. Springer and Takagi in view of Hendry disclose this element.
Springer and Takagi disclose this element for the reasons described above with respect to Ground 1, claim element 1[c][1].

305. Hendry discloses this element because Hendry describes identifying accessibility compliance issues, such as untagged input fields, and automatically

Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

remediating them. Ex. 1009, at ¶¶12, 28. Hendry also discloses embodiments where the remediation software is stored on a remote computer (e.g., remote server), and where a remediation request is received from a user's browser (i.e., user computer) over the internet (i.e., an electronic network). *Id.*, at ¶58.

306. Hendry discloses that the document modifier is executed on the server machine in response to receiving a request for a document that is not accessibility enabled from the client machine. In this process, Hendry describes that “[t]he document is analyzed to identify tags that are not recognizable by assistive technologies. The server machine executes the document modifier to produce a modified document, which is an accessibility-enabled version of the requested document. In response to receiving a request from the client machine for the particular document, the server machine sends the modified document.” *Id.*, at ¶53.

307. Furthermore, Hendry discloses a document modifier that runs in part at the server and in part at the client (computer system) such that the client can retrieve at least one JQuery or JavaScript (remediation code) from a remote server, where JQuery or JavaScript is added in a response to an HTTP request that JavaScript is the remediation code, and would be transmitted to the client (e.g., computer system). Ex. 1005, at ¶58. Hendry discloses a server plugin that

Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

“accesses the document (114) from a database at the server (110) and adds JQuery or JavaScript to obtain a modified document (116). The JQuery or JavaScript if executed would produce an accessibility-enabled version of the document (114) that can be properly interpreted by a screen reader (for example, execution of the JavaScript causes adding, removing, or modifying of information based on suitability for a screen reader).” *Id.*

308. A POSITA would have understood Springer and Takagi in view of Hendry to teach “applying, by the computer system, one or more pre-existing remediation code to the one or more compliance issues, wherein the one or more pre-existing remediations is generated by a remote server system performing at least.”

309. Therefore, Springer and Takagi in view of Hendry thus disclose and render obvious this element.

**e. “accessing the code associated with the web page”
(Claim 1[c][2])**

310. Springer and Hendry in view of Takagi disclose this element. Springer and Takagi disclose this element for the reasons described above with respect to Ground 1, claim element 1[c][2].

311. Furthermore, Hendry discloses that the analysis of a document includes defining a model of the document, i.e., the element(s) within the

Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

document for storing in a data structure, which correspond to a data value (for example, displayed data) or a keyword value (for example, a formatting tag).

Hendry states that “[t]he model of the document includes element characteristics, which include, but are not limited to, an element value, the manner in which an element is presented, the manner in which an element is accessed, the manner in which an element is manipulated, or the relationship of an element to other elements.” In an example, a model of a HTML or XML document is defined using a Document Object Model (DOM). Ex. 1009, at ¶65.

312. Therefore, Hendry also discloses the identification of inaccessible features in an embodiment by accessing and searching for text that is known to be misinterpreted by one or more assistive technologies. In an example, Hendry states that “[t]he text to be searched may include particular HTML formatting tags or particular document elements (for example, a layout table) in a HTML document.” *Id.*, at ¶60. Thus, Hendry discloses this claim element.

313. Therefore, Springer and Takagi in view of Hendry disclose and render obvious this element.

- f. **“receiving, by the remote server system, via a remediation interface input from a user, the input comprising a remediation action to manually remediate a non-programmatically-fixable compliance issue associated with the web page;”
(Claim 1[c][3])**

Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

314. Springer and Takagi in view of Hendry disclose this element.

Springer and Takagi disclose this claim element for the reasons described above with respect to Ground 1, claim element 1[c][3].

315. Hendry discloses that “[t]he document modifier is activated [on the server machine] in response to a user command.” Ex. 1009, at ¶57. In an embodiment, Hendry discloses that a document modifier is executed on the server machine in response to receiving a request for a document that is not accessibility enabled from the client machine. *Id.*, at ¶53. In this process, Hendry describes that “[t]he document is analyzed to identify tags that are not recognizable by assistive technologies. The server machine executes the document modifier to produce a modified document, which is an accessibility-enabled version of the requested document. In response to receiving a request from the client machine for the particular document, the server machine sends the modified document.” *Id.*

316. Therefore, to the extent Springer and Takagi do not already disclose this limitation, Springer and Takagi in view of Hendry disclose “receiving, by the remote server system, via a remediation interface input from a user, the input comprising a remediation action to manually remediate a non-programmatically-fixable compliance issue associated with the web page” as claimed.

Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

317. Therefore, Springer and Takagi in view of Hendry disclose and render obvious this element.

- g. “generating, by the remote server system, the one or more pre-existing remediations based on the input comprising the remediation action to manually remediate a non-programmatically-fixable compliance issue; ” (Claim 1[c][4])**

318. Springer and Takagi in view of Hendry disclose this element.

Springer and Takagi disclose this element for the reasons described above with respect to Ground 1, claim element 1[c][4].

319. Hendry also discloses this element because Hendry discloses using a document modifier, which is activated on a server machine in response to a user command, to executes a script for adding accessibility features to a document on the server machine for producing a modified document. Hendry discloses using “a user of an assistive technology that results in improving a user’s ability to understand the content of a document.” Ex. 1009, at ¶12. Furthermore, as described above, Hendry discloses that “[t]he document modifier is activated on the server machine in response to a user command.” A document is displayed by a browser executing on the client machine, and the user selects an accessibility option to make the document accessibility-enabled. In response to the user selection, the document modifier executes a script for adding accessibility features

to document on the server machine for producing a modified document. The script further calls a text-to-speech procedure to read out the text from the modified document while document is still being displayed by the browser.” *Id.*, at ¶57.

320. Therefore, Springer and Takagi in view of Hendry disclose and render obvious this element.

h. “storing, by the remote server system, the one or more pre-existing remediations in an electronic data storage medium; and” (Claim 1[c][5])

321. Springer and Takagi in view of Hendry disclose this element. Springer and Takagi disclose this element for the reasons described above with respect to Ground 1, claim element 1[c][5].

322. Hendry also discloses this element because Hendry discloses storing a modified document (one or more pre-existing remediations) in a data repository (an electronic data storage medium). Hendry discloses that its document modifier (120) is executed on the server machine (110) in response to receiving a request for a document from the client machine (102)” and that the “ document modifier (120) on the server machine (110) determines that the particular requested document (114) is not accessibility enabled during an analysis of the particular requested document (114). The document (114) is analyzed to identify tags that are not recognizable by assistive technologies (106). The server machine (110) executes

Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

the document modifier (120) to produce a modified document (116) which is an accessibility-enabled version of the particular requested document (114). In response to receiving a request from the client machine (102) for the particular document (114), the server machine (110) sends the modified document (116).”

Ex. 1009, at ¶53. Hendry discloses a “server machine (110) [that] includes a data repository (112) storing one or more of: documents (114), modified documents (116) and accessibility features (118). The data repository (112) corresponds to any data storage device (for example, local memory on the server machine (110), web servers connected over the internet, machines within a local area network, a memory on a mobile device, local memory on the client machine (102), etc.).” Ex. 1009, at ¶41. Further, Hendry discloses that “elements or various portions of data stored in the data repository (112) may be distributed (for example, stored on client machine (102) or at other servers). *Id.*

323. Therefore, Springer and Takagi in view of Hendry disclose and renders obvious this element.

- i. **“assigning, by the computer system, a descriptive attribute to the untagged element in the code of the web page based on the one or more pre-existing remediations, the descriptive attribute assigned to the untagged element adapted to enable an assistive technology to speak the descriptive attribute to a user.” (Claim 1[d])**

Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

324. Springer and Takagi in view of Hendry disclose this element.

Springer and Takagi disclose this element for the reasons described above with respect to Ground 1, claim element 1[d].

325. Hendry discloses this element. Hendry discloses adding (assigning), e.g., via JavaScript added by the server plugin (by the computer system), one or more accessibility features to a modified document (a descriptive attribute to the untagged element in the code of the web page based on the one or more pre-existing remediations), and using a browser renders the modified document for display to a user via screen reader (the descriptive attribute assigned to the untagged element adapted to enable an assistive technology to speak the descriptive attribute to a user). Ex. 1009, at ¶58 Hendry also discloses that “examples of accessibility features [] include, but are not limited to, text alternatives for non-text content,” e.g., such as a descriptive attribute to the untagged element. *Id.*, at ¶24.

326. Hendry discloses:

The component of the browser (104) parses the modified document (116) and executes the JavaScript added by the server plugin. The execution of the JavaScript results in adding one or more accessibility features (118) to the modified document (116) that were not previously included in the document (114). The browser (104) then renders the modified document (116) with the additional one or more accessibility features (118) for display to a

Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

user. A screen reader reads out content from the displayed document by properly interpreting the HTML and distinguishing the content that is to be read out from the formatting elements that are not be read out.

Ex. 1009, at ¶58.

327. Therefore, Springer and Takagi in view of Hendry disclose and renders obvious this element.

3. Claim 2

328. Claim 2 depends on claim 1 and recites “**wherein the one or more pre-existing remediations are javascript.**” Springer and Takagi in view of Hendry disclose this claim. Springer and Takagi disclose this claim for the reasons described above with respect to Ground 1, Claim 2.

329. Hendry also explicitly discloses that in some embodiments, its accessibility remediation code is JavaScript:

[A]ccessibility features (118) corresponds to code (for example, JavaScript) which when executed adds other accessibility features (118) to the document (114) to produce the modified document (116). For example, accessibility features (118) correspond to code which extracts content from a document with formatting tags that are not properly interpreted by assistive technologies. The accessibility features (118) correspond to code which uses the extracted content to produce a modified document (116) with different formatting tags than the original formatting tags.

Ex. 1009, at ¶25.

330. Therefore, Springer and Takagi in view of Hendry thus disclose and render this claim obvious.

4. Claim 3

331. Claim 3 depends on claim 1 and recites the additional requirement of: **“wherein the untagged element is an input field.”** Springer and Takagi in view of disclose this claim. Springer and Takagi disclose this claim for the reasons described above with respect to Ground 1, Claim 3.

332. Hendry also discloses this claim. Hendry discloses “automatically modifying documents to improve document accessibility.” Ex. 1009, at ¶12. Hendry also states that “modifying a document includes analyzing the contents of [] [an] original document (114) and creating a new document based on the analysis of the contents of the original document (114)” such as “[m]odifying a first document (114) to obtain a second document may be referred to herein as producing the second document.” *Id.*, at ¶47. Furthermore, Hendry discloses that “the analysis of [] [the] first document is used to produce [] [the] second document which includes one or more accessibility features that were not included in the first document.” *Id.*, at ¶68. Hendry also discloses “producing a second document includes modifying web pages to add one or more of: labels” and where the “the document is further modified to include labels.” *Id.*, at ¶¶76, 82. Hendry shows a

Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

document modified to include labels for untagged elements such as input fields

(*id.*):

```
<div class="application" aria-labelledby="application_label0">
<form class="form"><div class="form-basic textbox">
<span class="span_form" id="firstname_span">
<label for="firstname">First Name:</label></span>
<input type="text" id="firstname">
<br></div>
<div class="form-basic textbox">
<span class="span_form" id="lastname_span"><label
for="input1">Last Name:</label></span><input type="text"
name="lastname" id="input1">
<br></div>
<div class="form-basic submit"> <input type="submit"
value="Submit" id="input8"></div>
</form></div>
```

333. Therefore, Hendry discloses “wherein the untagged element is an input field” as claimed.

334. Therefore, Springer and Takagi in view of Hendry thus disclose and render this claim obvious.

5. Claim 4

335. Claim 4 depends on claim 1 and recites the additional requirement of: “**wherein the untagged element is an image.**” Springer and Takagi in view of Hendry disclose this claim. Springer and Takagi disclose this claim for the reasons described above with respect to Ground 1, Claim 4.

336. Hendry discloses this claim. Hendry discloses adding a descriptive attribute, e.g., such as the “alt attribute,” to an untagged element, e.g., such as an image. Hendry discloses that where “an image is used only as a design element (for example, a page divider line), then the “alt” attribute is set to an empty string so that a screen reader does not narrate a file name of the design element.” Ex. 1009, at ¶73. .

337. Therefore, Hendry discloses an image as an untagged element. *Id.*

338. Springer and Takagi in view of Hendry thus disclose and render this claim obvious.

6. Claim 6

339. Claim 6 depends on claim 1 and recites the additional requirement of: **“wherein the untagged element lacks the adequate descriptive attribute when one or more descriptive attributes of the untagged element are erroneous.”** Springer and Takagi in view of Hendry disclose this claim. Springer and Takagi disclose this claim for the reasons described above with respect to Ground 1, Claim 6.

340. Therefore, Springer and Takagi in view of Hendry thus disclose and render this claim obvious.

7. Claim 7

341. Claim 7 depends on claim 1 and recites the additional requirement of:
“wherein the assigning comprises changing the DOM or HTML code associated with the web page.” Springer and Takagi in view of Hendry disclose this claim. Springer and Takagi disclose this claim for the reasons described above with respect to Ground 1, Claim 7.

342. Therefore, Springer and Takagi in view of Takagi thus disclose and render this claim obvious.

8. Claim 8

343. Claim 8 depends on claim 1 and recites the additional requirement of:
“wherein the descriptive attribute is an HTML label.” Springer and Hendry in view of Takagi disclose this claim. Springer and Takagi disclose this claim for the reasons described above with respect to Ground 1, Claim 8.

344. Hendry discloses this claim. In particular, Hendry discloses
“producing a second document includes modifying web pages to add one or more of: labels” and where the “the document is further modified to include labels.” Ex. 1009, at ¶¶76, 82. Hendry shows a document modified to include labels for untagged elements such as input fields (*id.*):

```
<div class="application" aria-labelledby="application_label0">
<form class="form"><div class="form-basic textbox">
<span class="span_form" id="firstname_span">
<label for="firstname">First Name:</label></span>
<input type="text" id="firstname">
<br></div>
<div class="form-basic textbox">
<span class="span_form" id="lastname_span"><label
for="input1">Last Name:</label></span><input type="text"
name="lastname" id="input1">
<br></div>
<div class="form-basic submit"> <input type="submit"
value="Submit" id="input8"></div>
</form></div>
```

345. Therefore, Springer and Takagi in view of Hendry thus disclose and render this claim obvious.

9. Claim 9

346. Claim 9 depends on claim 1 and recites the additional requirement of: **“wherein the descriptive attribute is an alt-text label.”** Springer and Takagi in view of Hendry disclose this claim. Springer and Takagi disclose this claim for the reasons described above with respect to Ground 1, Claim 9.

347. Hendry discloses this claim. Specially, Hendry discloses that “examples of accessibility features [] include, but are not limited to, text alternatives for non-text content.” Ex. 1009, at ¶24. Therefore, Hendry discloses an accessibility feature (descriptive attribute) including a text alternative (alt-text label, e.g., short for: alternative text label).

348. Therefore, Springer and Takagi in view of Hendry thus disclose and render this claim obvious.

10. Claim 10

349. Claim 10 depends on claim 1 and recites the additional requirement of: **“wherein the computer system comprises one or more computing systems.”** Springer and Takagi in view of Hendry disclose this claim. Springer and Takagi disclose this claim for the reasons described above with respect to Ground 1, Claim 10.

350. Furthermore, Hendry discloses a “system (100) [that] includes a client machine (102) and a server machine (110).” Ex. 1009, at ¶34. A POSITA would have understood a “client machine and a server machine” to include one or more computer systems. Therefore, Hendry also discloses this claim.

351. Therefore, Springer and Takagi in view of Hendry thus disclose and render this claim obvious.

11. Independent Claim 11

- a. **“A non-transitory electronic storage medium with computer code stored thereon, the computer code configured to programmatically assign a descriptive attribute to an element on a web page to enable an audible description of the element, the web page having an associated document object model (DOM), and the computer code configured to perform, when**

**executed by a computer processor, the steps of:”
(Claim 11[pre])**

352. To the extent the preamble is limiting, Springer and Takagi in view of Hendry disclose the preamble. Springer and Takagi in view of Hendry disclose this element for the reasons described above for Ground 2, element 1[pre]. A POSITA would have understood that a computer implemented method (as claimed in claim 1) was routinely performed through computer code stored on a non-transitory electronic medium. Additionally, the recital of a “non-transitory computer readable information storage media” was well-known in the art and would have been obvious to a POSITA. Furthermore, a POSITA would have understood that a descriptive attribute to an element on a web page to include an untagged element as described in element 1[pre].

353. Springer and Takagi in view of Hendry thus disclose and render obvious the preamble.

b. “dynamically analyzing, by a computer system, code associated with the web page, the code comprising at least the DOM or HTML code” (Claim 11[a])

354. Springer and Takagi in view of Hendry disclose this element for the reasons described above with respect to Ground 2, Claim element 1[a].

Accordingly, Springer and Takagi in view of Hendry thus disclose or render this element obvious.

- c. **“detecting, by the computer system, one or more compliance issues relating to web accessibility standards in the code, wherein at least one of the one or more compliance issues comprises the element lacking an adequate descriptive attribute;” (Claim 11[b])**

355. Springer and Takagi in view of Hendry disclose this element for the reasons described above with respect to Ground 2, Claim element 1[b].

Accordingly, Springer and Takagi in view of Hendry thus disclose or render this element obvious.

- d. **“applying, by the computer system, one or more pre-existing remediations to the one or more compliance issues, wherein the one or more pre-existing remediations is generated by a remote server system performing at least:” (Claim 11[c][1])**

356. Springer and Takagi in view of Hendry disclose this element for the reasons described above with respect to Ground 2, Claim element 1[c][1].

Accordingly, Springer and Takagi in view of Hendry thus disclose or render this element obvious.

- e. **“accessing the code associated with the web page;” (Claim 11[c][2])**

357. Springer and Takagi in view of Hendry disclose this element for the reasons described above with respect to Ground 2, Claim element 1[c][2].

Accordingly, Springer and Takagi in view of Hendry thus disclose or render this element obvious.

- f. “receiving, by the remote server system, via a remediation interface input from a user, the input comprising a remediation action to manually remediate a non-programmatically-fixable compliance issue associated with the web page;” (Claim 11[c][3])**

358. Springer and Takagi in view of Hendry disclose this element for the reasons described above with respect to Ground 2, Claim element 1[c][3].

Accordingly, Springer and Takagi in view of Hendry thus disclose or render this element obvious.

- g. “generating, by the remote server system, the one or more pre-existing remediations based on the input comprising the remediation action to manually remediate a non-programmatically-fixable compliance issue;” (Claim 11[c][4])**

359. Springer and Takagi in view of Hendry disclose this element for the reasons described above with respect to Ground 2, Claim element 1[c][4].

Accordingly, Springer and Takagi in view of Hendry thus disclose or render this element obvious.

- h. “storing, by the remote server system, the one or more pre-existing remediations in an electronic data storage medium; and” (Claim 11[c][5])**

360. Springer and Takagi in view of Hendry disclose this element for the reasons described above with respect to Ground 2, Claim element 1[c][5].

Accordingly, Springer and Takagi in view of Hendry thus disclose or render this element obvious.

- i. **“assigning, by the computer system, a descriptive attribute to the element in the code of the web page based on the one or more pre-existing remediations, the descriptive attribute assigned to the element adapted to enable an assistive technology to speak the descriptive attribute to a user.” (Claim 11[d])**

361. Springer and Takagi in view of Hendry disclose this element for the reasons described above with respect to Ground 2, Claim element 1[d].

Accordingly, Springer and Takagi in view of Hendry thus disclose or render this element obvious.

12. Claim 12

362. Claim 12 depends on claim 11 and recites the additional requirement of: **“wherein the one or more pre-existing remediations are javascript.”**

Springer and Takagi in view of Hendry disclose this claim for the reasons described above with respect to Ground 2, Claim 2. Accordingly, Springer and Takagi in view of Hendry thus disclose or render this claim obvious.

13. Claim 13

363. Claim 13 depends on claim 11 and recites the additional requirement of: **“wherein the element is an input field.”** Springer and Takagi in view of Hendry disclose this claim for the reasons described above with respect to Ground

2, Claim 3. Accordingly, Springer and Takagi in view of Hendry thus disclose or render this claim obvious.

14. Claim 14

364. Claim 14 depends on claim 11 and recites the additional requirement of: **“wherein the element is an image.”** Springer and Takagi in view of Hendry disclose this claim for the reasons described above with respect to Ground 2, Claim 4. Accordingly, Springer and Takagi in view of Hendry thus disclose or render this claim obvious.

15. Claim 16

365. Claim 16 depends on claim 11 and recites the additional requirement of: **“wherein the element lacks the adequate descriptive attribute when one or more descriptive attributes of the element are erroneous.”** Springer and Takagi in view of Hendry disclose this claim for the reasons described above with respect to Ground 2, Claim 6. Accordingly, Springer and Takagi in view of Hendry thus disclose or render this claim obvious.

16. Claim 17

366. Claim 17 depends on claim 11 and recites the additional requirement of: **“wherein the assigning comprises changing the DOM or HTML code associated with the web page.”** Springer and Takagi in view of Hendry disclose this claim for the reasons described above with respect to Ground 2, Claim 7.

Accordingly, Springer and Takagi in view of Hendry thus disclose or render this claim obvious.

17. Claim 18

367. Claim 18 depends on claim 11 and recites the additional requirement of: **“wherein the descriptive attribute is an HTML label.”** Springer and Takagi in view of Hendry disclose this claim for the reasons described above with respect to Ground 2, Claim 8. Accordingly, Springer and Takagi in view of Hendry thus disclose or render this claim obvious.

18. Claim 19

368. Claim 19 depends on claim 11 and recites the additional requirement of: **“wherein the descriptive attribute is an alt-text label.”** Springer and Takagi in view of Hendry disclose this claim for the reasons described above with respect to Ground 2, Claim 9. Accordingly, Springer and Takagi in view of Hendry thus disclose or render this claim obvious.

19. Claim 20

369. Claim 20 depends on claim 11 and recites the additional requirement of: **“wherein the computer system comprises one or more computing systems.”** Springer and Takagi in view of Hendry disclose this claim for the reasons described above with respect to Ground 2, Claim 20. Accordingly, Springer and Takagi in view of Hendry thus disclose or render this claim obvious.

D. GROUND 3: Claims 1-4, 6-14, 16-20 are Unpatentable as Obvious Over Springer and Bigham in view of Hendry

1. Motivation to Combine

370. A POSITA would have been motivated to combine Springer, Bigham and Hendry to improve upon Springer's compliance retrofitter functions for determining and assigning descriptive attributes for untagged elements, e.g., such as labels for untagged input fields and images lacking alt-text. As described in detail above, Springer provides the various tools for checking and remediating web pages, but does not take advantage of the labeling service and/or mechanism for users to request an untagged element such as an image to be labeled by humans and the powerful functions of using a cached database that allows new labels to be seamlessly added after a web page is downloaded, that were commonplace by the time of the '532 patent.

371. As I described above, Springer describes tools for remediating untagged elements such as unlabeled form fields and elements lacking a text equivalent for a non-text element (e.g., an image or an AREA). For example, Springer discloses a form label checker, form label fixer, image checker, and image fixer. Ex. 1005, at Tables I(a)-(b), II(a)-(b). The form label checker finds inaccessible elements, identifying form fields (e.g., untagged input fields) in a website code that do not have a label (e.g., descriptive attribute) explicitly

Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

associated with it. The form label fixer remediates these elements, by finding a piece of text in the HTML code that is most likely the intended label for the form element, generating a label using that text, and adding a label tag with descriptive text in the code as the previously-untagged input field's label. Ex. 1005, at ¶38.

372. Springer also describes an ImageFixer and ImageMapFixer used to remediate compliance issues stemming from websites lacking a text equivalent for every non-text element such as an image. *See* Ex. 1005, at Tables II(a)-II(b). Springer's ImageFixer adds alt attributes to images, image buttons, and image links, as well as replacing alt attributes with more appropriate (e.g., in length) or meaningful descriptions. *Id.* Springer's ImageMapFixer does the same things for AREAS (described above). *Id.*

373. As described above, Springer describes its form label fixer as using a heuristic method (e.g., a series of rules) to analyze nearby elements and attributes in the website code (e.g., context cues) to identify candidate text for the form control label (e.g., input field). *Id.*, at ¶38. Springer allows for manual remediations, where a human can select which candidate text should be inserted as the label, and also allows for automatic remediations, where the system uses an "Autofix" library to automatically apply corrections based on previously-determined corrections, such as inserting text for labels (for untagged input fields)

Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

as previously-identified by the form label fixer, image fixer or by a human being. *Id.*, at ¶¶18, 21. Springer's compliance retrofitter described above is used to build and maintain compliant HTML. It can be used to retrofit web sites by enabling automated and user-driven accessibility enhancements. It updates HTML code to make it compliant with 36 CFR § 1194.22. *Id.*, at ¶26.

374. Springer also discloses a set of imager fixer tools, e.g., ImageFixer and ImageMapFixer, which remediate compliance issues stemming from websites lacking a text equivalent for every non-text element such as an image. *Id.*, at Tables II(a)-(b). ImageFixer remediates by adding alt attributes to images, image buttons, and image links, as well as replacing alt attributes with more appropriate or meaningful descriptions. *Id.* ImageMapFixer does the same things for AREA elements. *Id.* Springer does not limit how alt attributes for images are determined, thus leaving the system open to numerous means for determining an appropriate alt text for untagged images. *Id.*

375. Springer also discloses how to verify compliance accessibility standards and automatically retrofit the HTML of web pages to ensure compliance, as well as “[t]he fixers, or algorithms that make changes to HTML documents to bring the documents into compliance with the standard include manual fixes. Ex. 1005, at ¶16. For example, the ImageMapFixer adds alt attributes to AREA

Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

elements that currently do not have valid alt attributes.” *Id.*, at ¶15. Springer already discloses that descriptions can be manually input and stored in the autofix library. Further, Springer describes systems, including a compliance label fixer and image fixer, where “various modifications may be made.” *Id.*, at ¶56.

376. To this end, a POSITA would have been motivated to improve Springer’s fixer tools with a human labeling system as provided by Bigham’s WebInSight, which can be implemented via a cached local database, e.g., a remote server. To the extent Springer does not explicitly disclose retrieving manual remediations from a server, it would have been obvious. Takagi, Bigham, and Hendry all disclose remediations originating from a remote server. Ex. 1006, at 196, col. 1, ¶¶3, 6; Ex. 1008, at 184, col. 2, ¶2; Ex. 1009, at ¶52.

377. Furthermore, well before the ’532 patent, it was well known that it would be desirable to request for manual remediations by humans taught by Bigham. Ex. 1008, at 186, col. 2, ¶3-5. A POSITA would have been motivated to apply requested remediation, by humans other than the user, of untagged elements such as images since not all images can be applied for automatic labeling. Ex. 1006, at 186, col. 2, ¶3. A POSITA would have expected the combination to work without undue experimentation because its already operating over the web and methods for pulling data from a wide range of sources over a network, such as

Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

databases, was well known. Ex. 1006, at 196, col. 1, ¶¶3, 6; Ex. 1008, at 184, col. 2, ¶2; Ex. 1009, at ¶¶14, 15, 52; *see also* Ex. 1005, at Tables I(a)-(b), II(a).

Mechanisms for users to request that an untagged element such as an image to be labeled, e.g., by humans via a labeling service, were well known by the time of the '532 patent. Ex. 1008, at 186, col. 2, ¶3.

378. A POSITA would have understood that Bigham teaches how to generate labels for untagged elements such as images since Bigham's WebInSight system provides a mechanism for users to request for labeling by humans. *Id.* For example, Bigham discloses a "[t]o generate labels for these images, the WebInSight system provides a mechanism for users to request that an image be sent to a labeling service for labeling by humans." *Id.* Bigham's WebInSight system allows "users to request alternative text and receive a reply with an acceptable latency." *Id.*, at 186, col. 2 ¶4.

379. Furthermore, a POSITA would have been motivated to use Bigham's WebInsight system combined with Springer's system since the WebInSight system "implements a simple web-based application for humans to label images." *Id.* Also, Bigham discloses the WebInSight "system caches alternative text in a local database and can add new labels seamlessly after a web page is downloaded, resulting in minimal impact to the browsing experience." *Id.*, at Abstract. A

Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

POSITA would therefore have been motivated to combine Bigham's WebInSight system with Springer's fixers in order to improve the accuracy of creating appropriate label text for untagged elements such as images, including caching alternative text via Bigham's WebInSight system database to add untagged elements such as labels to images seamlessly, compared to the only heuristic methods disclosed by Springer. *Id.*, at 186, col. 2, ¶¶3-5, Abstract.

380. Similar to combining Springer with Bigham, a POSITA would have been motivated to combine Hendry's server machine functionalities in fixing the Web accessibility issues with the systems of Springer and Bigham. In general, and as discussed above, Hendry teaches systems and methods for modifying documents (e.g., web pages; *see* Ex. 1009, at ¶42) to improve document accessibility implemented on a client machine or server machine. Ex. 1009, at ¶59. For example, Hendry teaches that its methods can be performed on: (1) a local client computer, (2) a remote server, or (3) a combination of the two. *Id.* Hendry also teaches that its "[e]xamples [that] refer to a specific machine performing the steps [are merely] for the purposes of clarity and understanding" and that any step can be performed on any machine. *Id.* Hendry teaches that the various steps recited can be repeated, omitted, or performed in different order. *Id.*

Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

381. Hendry discloses that its “document modifier (120) is executed on the server machine (110) in response to receiving a request for a document from the client machine (102)” and that the “document modifier (120) on the server machine (110) determines that the particular requested document (114) is not accessibility enabled during an analysis of the particular requested document (114). The document (114) is analyzed to identify tags that are not recognizable by assistive technologies (106). The server machine (110) executes the document modifier (120) to produce a modified document (116) which is an accessibility-enabled version of the particular requested document (114). In response to receiving a request from the client machine (102) for the particular document (114), the server machine (110) sends the modified document (116).” Ex. 1009, at ¶53.

382. Hendry further discloses that “[t]he server machine (110) includes a data repository (112) storing one or more of: documents (114), modified documents (116) and accessibility features (118). The data repository (112) corresponds to any data storage device (for example, local memory on the server machine (110), web servers connected over the internet, machines within a local area network, a memory on a mobile device, local memory on the client machine (102), etc.).” *Id.*, at ¶41. Further Hendry discloses “[t]hose skilled in the art will

Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

appreciate that elements or various portions of data stored in the data repository (112) may be distributed (for example, stored on client machine (102) or at other servers)” where “the data repository (112) includes [a] flat, hierarchical, network based, relational, dimensional, object modeled, or data files structured otherwise” and “is maintained as a table of a SQL database.” *Id.*, at ¶41.

383. A POSITA would also have expected this combination to succeed with minimal modification. Indeed, Springer already includes a form fixer tool that analyzes context of a web page to find and extract an appropriate label for a web element, an image fixer for adding alt attributes to images and an image map fixer for “adding alt attributes to AREAs.” Ex. 1005, at Tables I(a)-(b), II(a)-(b), ¶¶16, 37-38. Bigham teaches that automatic labeling can be improved using humans to generate labels from some images and caching the labels on a local database for seamless access, while Hendry teaches modifying a HTML document on the server machine. Ex. 1008, at 186, col. 2, ¶¶3-5; Ex. 1009, at ¶15.

384. A POSITA could simply modify Springer’s “fixer” code to also query a database for matching descriptive attributes that had been authored by others. A POSITA could also add an interface (e.g., the WebInSight system’s web-based application) that allows humans to respond to users’ requests and quickly make alternative text labels via a cached local database with minimal work and the rest

Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

of Springer system, such as its FormChecker tool, would work as intended (with improved results from using the WebInsight system labeling service). Ex. 1008, at 186, col. 2, ¶¶3-5, Abstract; Ex. 1005, at Tables I (a)-(b), III.

385. Bigham also discloses the benefit of using simple web-based application for humans to label untagged elements such as images, e.g., providing a mechanism for users to request an image to be sent to a labeling service by humans, since automatic labeling (such as that performed by the Springer system) does not apply to all web images. Ex. 1008, at 186, col. 2 ¶¶3-5. Thus, a POSITA would have understood the benefit of combining the Bigham with the Springer system which discloses how to verify compliance accessibility standards and automatically or manually retrofit the HTML of web pages to ensure compliance Ex. 1005, at ¶¶8, 15-16, 38. Similar remediation techniques were well known. *See, e.g.*, Ex. 1015, at ¶¶2, 4, 15, 18; Ex. 1006, at ¶72.

386. Furthermore, a POSITA would have understood that collaborative web accessibility improvements leveraged by the participation of an open community were well known by the time of the '532 patent. Ex. 1007, at 196, col. 1, ¶3. For example, various types of social approaches other than Bigham, such as the 1997 ALT-server, Takagi, Teraguchi, Asakawa, and WebVisum, could provide an improvement over the existing automated accessibility enhancements described

by Springer by automatically inserting descriptive attribute generated by other users and stored on a remote database. *Id.*; Ex. 1031, at 2-4; Ex. 1032, at 177, col. 2, ¶¶3-4, Figure 4; Ex. 1033; Ex. 1034; Ex. 1035, 1:18-36, Figure 1. As I explained in Section IV, it was a commonly known approach to store alternative text in a remote server, where the alternative text was generated with the help manual remediation. Ex. 1008, at 186, col. 2 ¶¶3-5, Figure 4; Ex. 1007, at 186, col. 1, ¶6; *see also*, Ex. 1031, at 2-4; Ex. 1032, at 177, col. 2, ¶¶3-4, Figure 4; Ex. 1033; Ex. 1034; Ex. 1035, at 1:18-36, Figure 1.

387. Therefore, it would have been obvious to a POSITA to combine Springer, which discloses a heuristic method for checking an HTML document of a web page for compliance by running a checker and fixer tool to remediate a section of the HTML document containing the flagged violation by modifying HTML code, with Bigham and Hendry, both of which disclose inputting alternative text from a remote server machine.

2. Independent Claim 1

- a. **“A computer-implemented method of programmatically assigning a descriptive attribute to an untagged element on a web page to enable an audible description of the untagged element, the web page having an associated document object model (DOM), the method comprising:” (Claim 1[pre])**

Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

388. To the extent the preamble is limiting, Springer and Bigham in view of Hendry disclose this element. Springer discloses this claim for the reasons described above for Ground 1, Claim 1[pre]. Hendry discloses the preamble for the reasons described above for Ground 2, Claim 1[pre].

389. Furthermore, Bigham discloses the preamble because Bigham discloses a “system that automatically creates and inserts alternative text into web pages on-the-fly” where the “system caches alternative text in a local database and can add new labels seamlessly after a web page is downloaded.” Ex. 1008, at 181, Abstract. Bigham also discloses “[u]sing the Document Object Model (DOM), [] [to] extract all relevant information from each image on the web page, including those images that have been dynamically loaded. This method provides more accurate data than simply parsing the HTML text of a web page as we also have access to images loaded using dynamic scripting.” Ex. 1008, at 183, col. 2, ¶5.

390. Accordingly, Springer and Bigham in view of Hendry disclose and render obvious the preamble of claim 1.

b. “dynamically analyzing, by a computer system, code associated with the web page, the code comprising at least the DOM or HTML code;” (Claim 1[a])

391. Springer and Bigham in view of Hendry disclose this element, because Springer discloses using a computer system to dynamically analyze the

Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

data model associated with a web page (the DOM) to check for accessibility compliance issues. Springer discloses this element for the reasons described above for Ground 1, Claim 1[a]. Hendry discloses this element for the reasons described above for Ground 2, Claim 1[a].

392. Similarly, Bigham discloses this element because Bigham discloses a “system that automatically creates and inserts alternative text into web pages on-the-fly” where the “system caches alternative text in a local database and can add new labels seamlessly after a web page is downloaded.” Ex. 1008, at 181, Abstract. Furthermore, Bigham discloses that “[t]o formulate alternative text for images, we present three labeling modules based on web context analysis, enhanced optical character recognition (OCR) and human labeling.” Ex. 1008, at 181, Abstract. Additionally, Bigham discloses “[u]sing the Document Object Model (DOM), [] [to] extract all relevant information from each image on the web page, including those images that have been dynamically loaded. This method provides more accurate data than simply parsing the HTML text of a web page as we also have access to images loaded using dynamic scripting.” Ex. 1008, at 183, col. 2, ¶5. A POSITA would have understood that Bigham’s WebInSight system discloses this element because the WebInSight system uses “three labeling modules based on web context analysis” and uses the “Document Object Model

Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

(DOM), [] [to] extract all relevant information from each image on the web page to disclose. Ex. 1008, at 181, Abstract, at 183, col. 2, ¶5. Thus, Bigham also discloses this element.

393. Therefore, Springer and Bigham in view of Hendry thus disclose and render obvious this element.

- c. **“detecting, by the computer system, one or more compliance issues relating to web accessibility standards in the code, wherein at least one of the one or more compliance issues comprises the untagged element lacking an adequate descriptive attribute;”
(Claim 1[b])**

394. Springer and Bigham in view of Hendry disclose this element. Springer discloses this element for the reasons described above for Ground 1, Claim 1[b]. Hendry discloses this element for the reasons described above for Ground 2, Claim 1[b].

395. Bigham discloses this element. Bigham discloses that its WebInSight system includes “two modules which target requests for HTML and image content. The first [module] captures web pages and alters them to include both cached alternative text and the code allowing pages to be dynamically updated.” Ex. 1008, at 185 col. 1, ¶4. Bigham further discloses that “the second module calculates the MD5 hash of each observed image and records the mapping between this hash and the image’s URL in a local database.” Ex. 1008, at 185, col. 2, ¶1.

Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

Bigham also discloses that “MD5 hash serves as a unique identifier for each image, irrespective of its location on the web.” *Id.* Bigham discloses that “[t]his allows [] to detect when the image at a given URL changes” *Id.* Thus, Bigham’s WebInSight system detects if an image in its cache has changed based on its MD5 hash, and a POSITA would have understood that detection of a changed image would be a compliance issue because the “cached alternative text” will be out-of-date and therefore incorrect. Ex. 1008, at 185, col. 1, ¶4, col. 2, ¶1.

396. Furthermore, Bigham also discloses this element because Bigham’s WebInSight system retrieves alternative text from its database for an image, e.g., an untagged element, and uses three novel underlying labeling modules: enhanced web context analysis, optical character recognition (OCR) and human labeling, for automatically inserting alternative text for the untagged element. Ex. 1008, at Abstract, at 181, col. 1, ¶1, 183, col. 2, ¶5, 184 col. 2 ¶2, 183. Thus, Bigham also discloses this element.

397. Bigham discloses that “[t]he WebInSight system presented in this paper targets such significant images and provides a mechanism for automatically inserting appropriate alternative text.” Ex. 1008, at 181, col. 2, ¶4.

398. Also, Bigham discloses:

[T]he system retrieves alternative text from its database
for each image when it is present and requests that

Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

alternative text be calculated for any image not already in its database. It then dynamically inserts alternative text into the web page, instead of relying upon web authors to have added accessibility information. To calculate alternative text automatically the system utilizes OCR and web context labeling. While many images lend themselves to these automatic labeling, images that cannot be labeled automatically can be sent to human labeling services if the user desires.

Ex. 1008, at 184, col. 2, ¶2.

399. Furthermore, Bigham discloses:

To gather the image data for this portion of the study, we created a web crawler that loads each web page with Internet Explorer. Using the Document Object Model (DOM), we extract all relevant information from each image on the web page, including those images that have been dynamically loaded. This method provides more accurate data than simply parsing the HTML text of a web page as we also have access to images loaded using dynamic scripting.

Ex. 1008, at 183, col. 2, ¶5.

400. Bigham thus discloses “detecting, by the computer system, one or more compliance issues relating to web accessibility standards in the code, wherein at least one of the one or more compliance issues comprises an untagged element, the untagged element is an element lacking an adequate descriptive attribute” as claimed.

401. Therefore, Springer and Bigham in view of Hendry thus disclose and render obvious this element.

- d. **“applying, by the computer system, one or more pre-existing remediation code to the one or more compliance issues, wherein the one or more pre-existing remediations is generated by a remote server system performing at least:” (Claim 1[c][1])**

402. Springer and Bigham in view of Hendry disclose this element.

Springer discloses this element for the reasons described above for Ground 1, Claim 1[c][1]. Thus, Springer also discloses this element. Hendry discloses this element for the reasons described above for Ground 2, Claim 1[c][1]. Thus, Hendry also discloses this element.

403. Bigham also discloses this element because it discloses applying, such as by human labeling, one or more pre-existing remediation code to the one or more untagged elements such as images lacking appropriate alt-text (compliance issues), wherein the one or more pre-existing remediations is generated by a cached database (remote server). For example, Bigham discloses “[t]o generate labels for these images, the WebInSight system provides a mechanism for users to request that an image be sent to a labeling service for labeling by humans.” Ex. 1008 at 186, col. 2, ¶3. Bigham discloses the WebInSight “system caches alternative text in a local database and can add new labels seamlessly after a web

page is downloaded, resulting in minimal impact to the browsing experience.” *Id.*,
at Abstract.

404. Therefore, Springer and Bigham in view of Hendry thus disclose and
render obvious this element.

**e. “accessing the code associated with the web page”
(Claim 1[c][2])**

405. Springer and Bigham in view of Hendry disclose this element.
Springer discloses this element for the reasons described above for Ground 1,
Claim 1[c][2]. Thus, Springer also discloses this element. Hendry discloses this
element for the reasons described above for Ground 2, Claim 1[c][2]. Thus,
Hendry also discloses this element.

406. Bigham discloses allowing users to access the web through a proxy,
where the proxy arranges images to be labeled on his or her behalf. Ex. 1008, at
183, Section 4.1 ¶1. For example, Bigham discloses “[w]hen alternative text is
present in the database for an image viewed by a user of the system, the proxy
automatically appends the alternative text and the name of the module that
generated it to the value of the alt attribute of the associated img tag. When users
reach the image on the page, the alternative text is then available.” Ex. 1008, at
183, Section 4.1 ¶1. A POSITA would have understood that by using a proxy,

Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

Bigham's WebInSight system would need to access the code of whichever website the user had requested from the proxy.

407. Furthermore, Bigham discloses

A blind user accesses the web through the proxy, which arranges for images to be labeled on his or her behalf. When alternative text is present in the database for an image viewed by a user of the system, the proxy automatically appends the alternative text and the name of the module that generated it to the value of the alt attribute of the associated img tag. When users reach the image on the page, the alternative text is then available.

Ex. 1008, at 184, Section 4.1 ¶1.

408. Furthermore, Bigham discloses using the DOM (code) to extract (access) relevant information from an image on a webpage. Ex. 1008, at 183, ¶5.

409. Specifically Bigham discloses:

Using the Document Object Model (DOM), we extract all relevant information from each image on the web page, including those images that have been dynamically loaded. This method provides more accurate data than simply parsing the HTML text of a web page as we also have access to images loaded using dynamic scripting.

Ex. 1008, at 183, ¶5

410. Therefore, Springer and Bigham in view of Hendry disclose and render obvious this element.

- f. **“receiving, by the remote server system, via a remediation interface input from a user, the input**

**comprising a remediation action to manually
remediate a non-programmatically-fixable
compliance issue associated with the web page;”
(Claim 1[c][3])**

411. Springer and Bigham in view of Hendry disclose this element.

Hendry discloses this element for the reasons described above for Ground 2, Claim 1[c][3].

412. Bigham further describes receiving, by a cached database (remote server), via simple web-based application for humans to label untagged elements such as images (remediation interface input), comprising a label of an untagged element such as an image that may not apply for an automatic labeling method (a remediation action to manually remediate a non-programmatically-fixable compliance issue associated with the web page).

413. Specifically Bigham discloses that “automatic labeling methods mentioned earlier work well for the large subset of significant images, but they do not apply to all web images” and “[t]o generate labels for these images, the WebInSight system provides a mechanism for users to request that an image be sent to a labeling service for labeling by humans.” Ex. 1006, at 186, col. 2, ¶3. Bigham’s WebInSight system allows “users to request alternative text and receive a reply with an acceptable latency.” *Id.*, at 186, col. 2, ¶4. Bigham also discloses that “WebInSight currently implements a simple web-based application for humans

Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

to label images.” *Id.* Furthermore, Bigham discloses the WebInSight “system caches alternative text in a local database and can add new labels seamlessly after a web page is downloaded, resulting in minimal impact to the browsing experience.” *Id.*, at Abstract.

414. Therefore, Springer and Bigham in view of Hendry disclose and render obvious this element.

- g. “generating, by the remote server system, the one or more pre-existing remediations based on the input comprising the remediation action to manually remediate a non-programmatically-fixable compliance issue; ” (Claim 1[c][4])**

415. Springer and Bigham in view of Hendry disclose this element. Hendry discloses this element for the reasons described above for Ground 2, Claim 1[c][4].

416. Bigham further discloses generating labels form untagged elements such as images, by a cached database (remote server), the generated label of an untagged element such as an image that may not apply for an automatic labeling method (one or more pre-existing remediations based on the input comprising the remediation action to manually remediate a non-programmatically-fixable compliance issue).

Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

417. Bigham discloses that “automatic labeling methods mentioned earlier work well for the large subset of significant images, but they do not apply to all web images” and “[t]o generate labels for these images, the WebInSight system provides a mechanism for users to request that an image be sent to a labeling service for labeling by humans.” Ex. 1008, at 186, col. 2, ¶3. Bigham’s WebInSight system allows “users to request alternative text and receive a reply with an acceptable latency.” *Id.*, at 186, col. 2, ¶4. Bigham also discloses that “WebInSight currently implements a simple web-based application for humans to label images.” *Id.* Furthermore, Bigham discloses the WebInSight “system caches alternative text in a local database and can add new labels seamlessly after a web page is downloaded, resulting in minimal impact to the browsing experience.” *Id.*, at Abstract.

418. Therefore, Springer and Bigham in view of Hendry discloses and render obvious the element “generating, by the remote server system, the one or more pre-existing remediations based on the input comprising the remediation action to manually remediate a non-programmatically-fixable compliance issue.”

- h. “storing, by the remote server system, the one or more pre-existing remediations in an electronic data storage medium; and” (Claim 1[c][5])**

Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

419. Springer and Bigham in view of Hendry disclose this element.

Hendry discloses this element for the reasons described above for Ground 2, Claim 1[c][5]. Thus, Hendry also discloses this element.

420. Bigham further discloses using a database (a remote server, an electronic storage medium) to cache alternative text (one or more preexisting remediations). Ex. 1008, at 181, Abstract. Bigham’s WebInSight system “caches alternative text in a local database and can add new labels seamlessly after a web page is downloaded, resulting in minimal impact to the browsing experience.” *Id.*

421. Therefore, Springer and Bigham in view of Hendry discloses and renders obvious this element.

- i. **“assigning, by the computer system, a descriptive attribute to the untagged element in the code of the web page based on the one or more pre-existing remediations, the descriptive attribute assigned to the untagged element adapted to enable an assistive technology to speak the descriptive attribute to a user.” (Claim 1[d])**

422. Springer and Bigham and Hendry disclose this element. Springer discloses this element for the reasons described above for Ground 1, Claim 1[d]. Hendry discloses this element for the reasons described above for Ground 2, Claim 1[d].

Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

423. Bigham discloses also discloses element. Bigham discloses a “system that automatically creates and inserts alternative text into web pages on-the-fly” where the “system caches alternative text in a local database and can add new labels seamlessly after a web page is downloaded.” Ex. 1008, at 181, Abstract. Furthermore, Bigham discloses that “[t]o formulate alternative text for images, we present three labeling modules based on web context analysis, enhanced optical character recognition (OCR) and human labeling.” *Id.* Bigham also discloses that its WebInSight system includes “two modules which target requests for HTML and image content. The first [module] captures web pages and alters them to include both cached alternative text and the code allowing pages to be dynamically updated.” Ex. 1008, at 185 col. 1, ¶4. Bigham also stresses that screen readers depend on proper alternative text for images. *Id.*, at Abstract, 181, col. 2, ¶3, 183, col. 1, ¶6, col. 2, ¶1. Thus, Bigham discloses capturing and altering (assigning), by the computer system, alternative text (a descriptive attribute to the untagged element in the code of the web page based on the one or more pre-existing remediations), the alternative text captured and altered to the untagged element adapted to recognized by a screen reader for a blind user (assistive technology to speak the descriptive attribute to a user).

424. Therefore, Springer and Bigham in view of Hendry discloses and renders obvious this element.

3. Claim 2

425. Claim 2 depends on claim 1 and recites “**wherein the one or more pre-existing remediations are javascript.**” Springer and Bigham in view of Hendry disclose this claim. Springer discloses this element for the reasons described above for Ground 1, Claim 2. Hendry discloses this claim for the reasons described above for Ground 2, Claim 2.

426. Bigham also explicitly discloses using a “proxy [that] inserts Javascript code into web pages that allows them to dynamically query the labeling framework after the main content has loaded.” Ex. 1008, at 184, col. 2, ¶3. Thus, Bigham also discloses this claim.

427. Therefore, Springer and Bigham in view of Hendry thus disclose and render this claim obvious.

4. Claim 3

428. Claim 3 depends on claim 1 and recites the additional requirement of: “**wherein the untagged element is an input field.**” Springer and Bigham in view of Hendry disclose this claim. Springer discloses this claim for the reasons

Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

described above for Ground 1, Claim 3. Hendry also discloses this claim for the reasons described above for Ground 2, Claim 3.

429. Furthermore, it would have been obvious to apply the teachings of Bigham to Springer and Hendry. Bigham discloses “a system that automatically creates and inserts alternative text into web pages on-the-fly.” Ex. 1008, at 181, Abstract. Bigham teaches generating alternative text for untagged elements and further discloses a mechanism for users to request an untagged element such as an image to be sent for labeling by humans. Ex. 1006, at 186, col. 2, ¶3. For example, Bigham’s WebInSight system allows “users to request alternative text and receive a reply with an acceptable latency.” *Id.*, at 186, col. 2, ¶4. Both Springer and Hendry both already teach manual human labeling (Ex. 1005, at ¶¶8, 21; Ex. 1009, at ¶85), corresponding to the same role of Bigham’s “simple web-based application for humans to label images.” Ex. 1008, at 186, col. 2, ¶4. Springer and Hendry also both teach storing remediation code in an electronic storage medium such as a database (Ex. 1005, at ¶18; Ex. 1009, at ¶41), corresponding to the same role of Bigham’s database. For example, Bigham’s Figure 4 below shows a database:

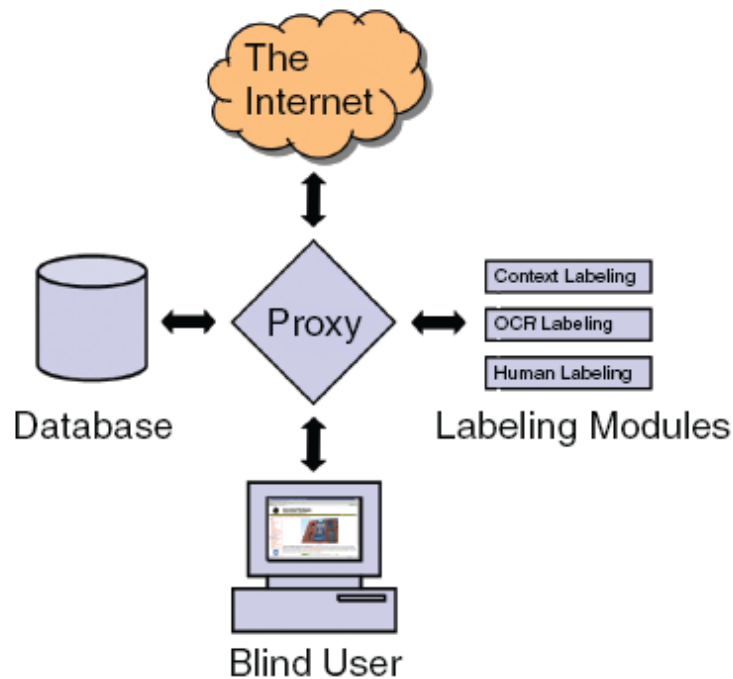


Figure 4: The architecture of WebInSight.

Ex. 1008, Figure 4. “When a user loads a web page, the system retrieves alternative text from its database for each image when it is present and requests that alternative text be calculated for any image not already in its database.” Ex. 1008, at 184, col. 2, ¶2.

430. Because both Springer and Hendry already teach Bigham Figure 4’s key components of human labeling and an associated database, a POSITA would have been motivated to apply the same approach as Bigham to other types of untagged elements in addition to images, e.g., such as an input field, as taught by Springer and Hendry.

431. Therefore, Springer and Bigham in view of Hendry thus disclose and render this claim obvious.

5. Claim 4

432. Claim 4 depends on claim 1 and recites the additional requirement of: **“wherein the untagged element is an image.”** Springer and Bigham in view of Hendry disclose this claim. Springer discloses this claim for the reasons described above for Ground 1, Claim 4. Hendry also discloses this claim for the reasons described above for Ground 2, Claim 4.

433. Similarly, Bigham’s WebInSight system automatically creates and inserts alternative text for images. Specifically Bigham discloses “automatically creates and inserts alternative text into web pages on-the-fly. To formulate alternative text for images, we present three labeling modules based on web con-text analysis, enhanced optical character recognition (OCR) and human labeling.” Ex. 1008, at 181, Abstract. Thus Bigham also discloses this claim.

434. Springer and Bigham in view of Hendry thus disclose and render this claim obvious.

6. Claim 6

435. Claim 6 depends on claim 1 and recites the additional requirement of: **“wherein the untagged element lacks the adequate descriptive attribute when**

one or more descriptive attributes of the untagged element are erroneous.”

Springer and Bigham in view of Hendry disclose this claim. Springer discloses this claim for the reasons described above for Ground 1, Claim 6.

436. Bigham discloses this because Bigham discloses remediating untagged elements such as images which are significant but not labeled, or improperly labeled (e.g., with a zero-length alt attribute), or images that are insignificant but improperly labeled (e.g., with a non-zero-length alt attribute). Ex. 1008, at 183, col. 1, ¶¶2-4, col. 2, ¶¶1-6.

437. Bigham discloses:

A significant image was properly labeled if its associated img tag contains a non-zero length alt, title or longdesc attribute, or if the image is contained in the same anchor (link) tag as additional text. An insignificant image was properly labeled if it contained a zero-length alt attribute. In our studies, we first focused on university, government and high-traffic websites. Next we looked at the traffic generated by members of the University of Washington Computer Science Department.

Ex. 1008, at 183, col. 2, ¶3.

438. Bigham also discloses:

Images containing more than one color and having both width and height greater than 10 pixels are classified as significant. Images that are clickable, either because they are contained within anchor tags or because of defined script events, are also considered significant. Our criteria for classifying an image as significant for this portion of

Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

the study was determined from manual observation and we
have found it to be quite accurate.

Ex. 1008, at 183, col. 2, ¶6.

439. Bigham further discloses:

Once the significant images are identified, we examine
each for alternative text according to the labeling criteria
described in the introduction to this section. We then
compute the aggregate image statistics for each
homepage to determine the percentage of significant and
insignificant images that are properly labeled. Table 1
summarizes this information for each group.

Ex. 1008, at 183, col. 2, ¶7.

440. Bigham further discloses that its WebInSight system includes “two
modules which target requests for HTML and image content. The first [module]
captures web pages and alters them to include both cached alternative text and the
code allowing pages to be dynamically updated.” Ex. 1008, at 185, col. 1, ¶4.

Bigham further discloses that “the second module calculates the MD5 hash of each
observed image and records the mapping between this hash and the image’s URL
in a local database.” Ex. 1008, at 185, col. 2, ¶1. Bigham also discloses that
“MD5 hash serves as a unique identifier for each image, irrespective of its location
on the web.” *Id.* Bigham discloses that “[t]his allows [] to detect when the image
at a given URL changes” *Id.* Thus, Bigham’s WebInSight system detects if an
image in its cache has changed based on its MD5 hash, and a POSITA would have

understood that detection of a changed image would be a compliance issue because the “cached alternative text” will be out-of-date and therefore “erroneous.” Ex. 1008, at 185, col. 1, ¶4, col. 2, ¶1.

441. Therefore, Springer and Bigham in view of Hendry thus disclose and render this claim obvious.

7. Claim 7

442. Claim 7 depends on claim 1 and recites the additional requirement of: **“wherein the assigning comprises changing the DOM or HTML code associated with the web page.”** Springer and Bigham in view of Hendry disclose this claim. Springer discloses this claim for the reasons described above for Ground 1, Claim 7.

443. Bigham discloses this claim because Bigham discloses dynamically inserting alternative text into a webpage. Ex. 1008, at 184, Section 4 ¶1. A POSITA would have understood that dynamically inserting alternative text into a webpage includes changing the DOM or HTML code associated with the webpage.

444. Specifically, Bigham discloses

When a user loads a web page, the system retrieves alternative text from its database for each image when it is present and re-requests that alternative text be calculated for any image not already in its database. It then dynamically inserts alter-native text into the web page, instead of

Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

relying upon web authors to have added accessibility
information.

Ex. 1008, at 184, Section 4 ¶1.

445. Therefore, Springer and Bigham in view of Bigham thus disclose and
render this claim obvious.

8. Claim 8

446. Claim 8 depends on claim 1 and recites the additional requirement of:
“**wherein the descriptive attribute is an HTML label.**” Springer and Hendry in
view of Bigham disclose this claim. Springer discloses this claim for the reasons
described above for Ground 1, Claim 8. Hendry discloses this claim for the
reasons described above for Ground 2, Claim 8.

447. Furthermore, it would have been obvious to apply the teachings of
Bigham to Springer and Hendry. Bigham discloses “a system that automatically
creates and inserts alternative text into web pages on-the-fly.” Ex. 1008, at 181,
Abstract. Bigham teaches generating alternative text for untagged elements and
further discloses a mechanism for users to request an untagged element such as an
image to be sent for labeling by humans. Ex. 1006, at 186, col. 2 ¶3. For example,
Bigham’s WebInSight system allows “users to request alternative text and receive
a reply with an acceptable latency.” *Id.*, at 186, col. 2, ¶4. Both Springer and
Hendry both already teach manual human labeling (Ex. 1005, at ¶¶8, 21, Ex. 1009,

at ¶85), corresponding to the same role of Bigham’s “simple web-based application for humans to label images.” Ex. 1008, at 186 col. 2, ¶4. Springer and Hendry also both teach storing remediation code in a electronic storage medium such as a database (Ex. 1005, at ¶18, Ex. 1009, at ¶41), corresponding to the same role of Bigham’s database. For example, Bigham’s Figure 4 below shows a database:

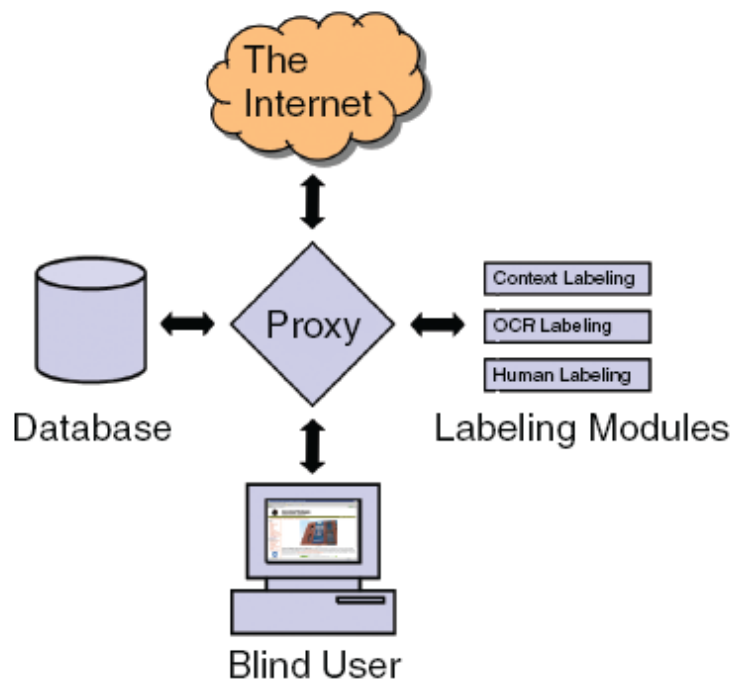


Figure 4: The architecture of WebInSight.

Ex. 1008, at Figure 4. “When a user loads a web page, the system retrieves alternative text from its database for each image when it is present and requests

that alternative text be calculated for any image not already in its database.” Ex. 1008, at 184, col. 2, ¶2.

448. Because both Springer and Hendry already teach Bigham Figure 4’s key components of human labeling and an associated database, a POSITA would have been motivated to apply the same approach as Bigham to other types of untagged elements in addition to images, e.g., such as for an HTML label, as taught by Springer and Hendry.

449. Therefore, Springer and Bigham in view of Hendry thus disclose and render this claim obvious.

9. Claim 9

450. Claim 9 depends on claim 1 and recites the additional requirement of: **“wherein the descriptive attribute is an alt-text label.”** Springer and Bigham in view of Hendry disclose this claim. Springer discloses this claim for the reasons described above for Ground 1, Claim 9. Hendry also discloses this claim for the reasons described above for Ground 2, Claim 9.

451. Bigham further discloses this claim because Bigham discloses automatically creating and inserting alternative text for images. Ex. 1008, at 181, Abstract. Bigham discloses “automatically creates and inserts alternative text into web pages on-the-fly. To formulate alternative text for images, we present three

labeling modules based on web con-text analysis, enhanced optical character recognition (OCR) and human labeling.” Ex. 1008, at 181, Abstract. Thus Bigham also discloses this claim.

452. Therefore, Springer and Bigham in view of Hendry thus disclose and render this claim obvious.

10. Claim 10

453. Claim 10 depends on claim 1 and recites the additional requirement of: **“wherein the computer system comprises one or more computing systems.”** Springer and Bigham in view of Hendry disclose this claim. Springer discloses this claim for the reasons described above for Ground 1, Claim 10. Hendry discloses this claim for the reasons described above for Ground 2, Claim 10. Bigham also discloses this claim limitation because every computer system comprises one or more computing system.

454. Therefore, Springer and Bigham in view of Hendry thus disclose and render this claim obvious.

11. Independent Claim 11

- a. **“A non-transitory electronic storage medium with computer code stored thereon, the computer code configured to programmatically assign a descriptive attribute to an element on a web page to enable an audible description of the element, the web page having an associated document object model (DOM),**

**and the computer code configured to perform, when
executed by a computer processor, the steps of:”
(Claim 11[pre])**

455. To the extent the preamble is limiting, Springer and Bigham in view of Hendry disclose the preamble. Springer and Bigham in view of Hendry disclose this element for the reasons described above for Ground 3, Claim 1[pre]. A POSITA would have understood that a computer implemented method (as claimed in claim 1) was routinely performed through computer code stored on a non-transitory electronic medium. Additionally, the recital of a “non-transitory computer readable information storage media” was well-known in the art and would have been obvious to a POSITA. Furthermore, a POSITA would have understood that a descriptive attribute to an element on a web page to include an untagged element as described in claim 1[pre].

456. Springer and Bigham in view of Hendry thus disclose and render obvious the preamble.

**b. “dynamically analyzing, by a computer system, code
associated with the web page, the code comprising at
least the DOM or HTML code” (Claim 11[a])**

457. Springer and Bigham in view of Hendry disclose this element for the reasons described above with respect to Ground 3, Claim element 1[a].

Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

Accordingly, Springer and Bigham in view of Hendry thus disclose or render this element obvious.

- c. **“detecting, by the computer system, one or more compliance issues relating to web accessibility standards in the code, wherein at least one of the one or more compliance issues comprises the element lacking an adequate descriptive attribute;” (Claim 11[b])**

458. Springer and Bigham in view of Hendry disclose this element for the reasons described above with respect to Ground 3, Claim element 1[b].

Accordingly, Springer and Bigham in view of Hendry thus disclose or render this element obvious.

- d. **“applying, by the computer system, one or more pre-existing remediations to the one or more compliance issues, wherein the one or more pre-existing remediations is generated by a remote server system performing at least:” (Claim 11[c][1])**

459. Springer and Bigham in view of Hendry disclose this element for the reasons described above with respect to Ground 3, Claim element 1[c][1].

Accordingly, Springer and Bigham in view of Hendry thus disclose or render this element obvious.

- e. **“accessing the code associated with the web page;” (Claim 11[c][2])**

460. Springer and Bigham in view of Hendry disclose this element for the reasons described above with respect to Ground 3, Claim element 1[c][2].

Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

Accordingly, Springer and Bigham in view of Hendry thus disclose or render this element obvious.

- f. **“receiving, by the remote server system, via a remediation interface input from a user, the input comprising a remediation action to manually remediate a non-programmatically-fixable compliance issue associated with the web page;” (Claim 11[c][3])**

461. Springer and Bigham in view of Hendry disclose this element for the reasons described above with respect to Ground 3, Claim element 1[c][3].

Accordingly, Springer and Bigham in view of Hendry thus disclose or render this element obvious.

- g. **“generating, by the remote server system, the one or more pre-existing remediations based on the input comprising the remediation action to manually remediate a non-programmatically-fixable compliance issue;” (Claim 11[c][4])**

462. Springer and Bigham in view of Hendry disclose this element for the reasons described above with respect to Ground 3, Claim element 1[c][4].

Accordingly, Springer and Bigham in view of Hendry thus disclose or render this element obvious.

- h. **“storing, by the remote server system, the one or more pre-existing remediations in an electronic data storage medium; and” (Claim 11[c][5])**

463. Springer and Bigham in view of Hendry disclose this element for the reasons described above with respect to Ground 3, Claim element 1[c][5].

Accordingly, Springer and Bigham in view of Hendry thus disclose or render this element obvious.

- i. **“assigning, by the computer system, a descriptive attribute to the element in the code of the web page based on the one or more pre-existing remediations, the descriptive attribute assigned to the element adapted to enable an assistive technology to speak the descriptive attribute to a user.” (Claim 11[d])**

464. Springer and Bigham in view of Hendry disclose this element for the reasons described above with respect to Ground 3, Claim element 1[d].

Accordingly, Springer and Bigham in view of Hendry thus disclose or render this element obvious.

12. Claim 12

465. Claim 12 depends on claim 11 and recites the additional requirement of: **“wherein the one or more pre-existing remediations are javascript.”**

Springer and Bigham in view of Hendry disclose this claim for the reasons described above with respect to Ground 3, Claim 2. Accordingly, Springer and Bigham in view of Hendry thus disclose or render this claim obvious.

13. Claim 13

Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

466. Claim 13 depends on claim 11 and recites the additional requirement of: **“wherein the element is an input field.”** Springer and Bigham in view of Hendry disclose this claim for the reasons described above with respect to Ground 3, Claim 3. Accordingly, Springer and Bigham in view of Hendry thus disclose or render this claim obvious.

14. Claim 14

467. Claim 14 depends on claim 11 and recites the additional requirement of: **“wherein the element is an image.”** Springer and Bigham in view of Hendry disclose this claim for the reasons described above with respect to Ground 3, Claim 4. Accordingly, Springer and Bigham in view of Hendry thus disclose or render this claim obvious.

15. Claim 16

468. Claim 16 depends on claim 11 and recites the additional requirement of: **“wherein the element lacks the adequate descriptive attribute when one or more descriptive attributes of the element are erroneous.”** Springer and Bigham in view of Hendry disclose this claim for the reasons described above with respect to Ground 3, Claim 6. Accordingly, Springer and Bigham in view of Hendry thus disclose or render this claim obvious.

16. Claim 17

469. Claim 17 depends on claim 11 and recites the additional requirement of: **“wherein the assigning comprises changing the DOM or HTML code associated with the web page.”** Springer and Bigham in view of Hendry disclose this claim for the reasons described above with respect to Ground 3, Claim 7. Accordingly, Springer and Bigham in view of Hendry thus disclose or render this claim obvious.

17. Claim 18

470. Claim 18 depends on claim 11 and recites the additional requirement of: **“wherein the descriptive attribute is an HTML label.”** Springer and Bigham in view of Hendry disclose this claim for the reasons described above with respect to Ground 3, Claim 8. Accordingly, Springer and Bigham in view of Hendry thus disclose or render this claim obvious.

18. Claim 19

471. Claim 19 depends on claim 11 and recites the additional requirement of: **“wherein the descriptive attribute is an alt-text label.”** Springer and Bigham in view of Hendry disclose this claim for the reasons described above with respect to Ground 3, Claim 9. Accordingly, Springer and Bigham in view of Hendry thus disclose or render this claim obvious.

19. Claim 20

472. Claim 20 depends on claim 11 and recites the additional requirement of: **“wherein the computer system comprises one or more computing systems.”** Springer and Bigham in view of Hendry disclose this claim for the reasons described above with respect to Ground 3, Claim 20. Accordingly, Springer and Bigham in view of Hendry thus disclose or render this claim obvious.

E. GROUND 4: Claims 1-4, 6-14, 16-20 are Unpatentable as Obvious Over Markel and Takagi.

1. Motivation to Combine

473. A POSITA would have been motivated to combine Markel and Takagi in order to improve upon Markel’s modules for diagnosing, evaluating, reporting, and retrofitting code violations existing in websites, e.g., such as missing HTML tags, alt tag for images, a title tag, and labels for form fields. Markel provides several modules for automatically, semi-automatically, or manually remediating web pages, and can be further benefit from, and take advantage of, collaborative metadata authoring and the powerful functions of a remote server that were commonplace by the time of the ’532 patent.

474. Markel describes several modules for correcting code violations on a website including missing HTML tags, alt tag for images, a title tag, and labels for form fields. Ex. 1006, at Abstract, ¶¶72, 83, 88. Markel discloses an assessment module, a reporting module, and a correcting module, together used for evaluating

Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

and correcting websites. Ex. 1006, at Abstract. Specifically, Markel discloses an assessment module for parsing and analyzing a website based on a function of a ruleset of design guideline. *Id.* Markel's ruleset of design guidelines comprise a set of standards, such as the Section 508 standard and a usability standard such as the W3C Web Content Accessibility Guideline. *Id.*, at Abstract, ¶73. Markel's assessment module identifies standards violations that can include missing HTML tags, alt tag for images, a title tag, and labels for form fields (e.g., input fields). *Id.*, at Abstract, ¶¶72, 83, 88. Markel's reporting module is used to list these items, e.g., the design errors and standards violations, that can be automatically corrected, that can use semi-automatic assisted correction, and that are for manual correction. Ex. 1006, at Abstract. Markel also discloses a correcting module for automatically, semi-automatically, or manually correcting the listed items. *Id.*, at Abstract.

475. Markel allows for "semi-automated wizard-assisted manual implementation, and support for manual correction of reported ruleset violation." Ex. 1006, at ¶78. For example, Markel's system "provides additional functionality including fixing minor alignment problems or font specifications, or adding ALT tags to images after querying the user for appropriate labels." *Id.* Thus, Markel's

Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

system can be used for automatic, semi-automatic, or manual remediation of ruleset violations. *Id.*, at Abstract ¶¶72, 78, 83, 88.

476. Furthermore, Markel also discloses that access to Markel’s system “is provided over the internet and the graphical user interface 112 is embodied in a web page accessed via a web browser 108.” Ex. 1006, ¶104. A POSITA would have understood that access over the internet to disclose that Markel’s system resides on a remote server. Furthermore, referring to Figure 1 of Markel, the computer of Markel’s system is clearly accessed by a user through a network 106, e.g., such as the internet, and is separate from the user’s computer 105. To this end, a POSITA would have been motivated to improve Markel’s system for evaluating and correcting websites, which can access over the internet. To the extent Markel does not explicitly disclose retrieving manual remediations from a server, it would have been obvious. Takagi, Bigham, and Hendry all disclose remediations originating from a remote server. Ex. 1006, at 196, col. 1, ¶¶3, 6; Ex. 1008, at 184, col. 2, ¶2; Ex. 1009, ¶52.

477. Furthermore, well before the ’532 patent, it was well known that it would be desirable to share manual fixes across a community as taught by Takagi. Ex. 1006, at 195, col. 2, ¶2, 196, col. 1, ¶6. A POSITA would have been motivated to combine Markel with Takagi to take advantage of Takagi’s teaching share

Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

manual fixes across a community in order to address errors, such as elements lacking a descriptive attribute, which a user or administrator of a website may find costly or time-consuming to fix on their own. Ex. 1006, at ¶102; Ex. 1007, at 195, ¶5, 196, ¶6.

478. A POSITA would also have been motivated to do so because it would save the number of manual remediations that would have to be done by any one individual in a community, and cut down on time it took to wait for a new manual remediation (if one already existed), which would then not delay other remediation requests for images that had no description. A POSITA would have expected the combination to work without undue experimentation because its already operating over the web and methods for pulling data from a wide range of sources over a network, such as databases, was well known. Ex. 1007, at 196, col. 1 ¶¶3, 6; Ex. 1008, at 184, col. 2, ¶2; Ex. 1009, ¶¶14-15, 52; *see also* Ex. 1005 at Tables I(a)-(b), II(a). For example, collaborative web accessibility improvements leveraged by the participation of an open community were well known by the time of the '532 patent. Ex. 1007, at 196, col. 1, ¶3. Also, various types of social approaches, such as the 1997 ALT-server, WebInSight, and WebVisum, could provide an improvement over the automatically, semi-automatically, or manually remediation described by Markel. *Id.*; Ex. 1006, at Abstract. Other such references, such as

Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

Takagi, Teraguchi, and Asakawa are also discussed in the background section above.

479. Furthermore, a POSITA would have understood that collaborative web accessibility improvements leveraged by the participation of an open community were well known by the time of the '532 patent. Ex. 1007, at 196, col. 1, ¶3. For example, various types of social approaches, such as the 1997 ALT-server, Takagi, WebInSight (Bigham), Teraguchi, Asakawa, and WebVisum, could provide an improvement over the existing techniques for evaluating and correcting websites described by Markel. *Id.*; Ex. 1031, at 2-4; Ex. 1032, at 177, col. 2, ¶¶3-4, Figure 4.; Ex. 1033; Ex. 1034; Ex. 1035, 1:18-36, Figure 1.

480. A POSITA would have understood that Takagi teaches how to implement a social labeling system, including at the end user level, the supporting community tasks and using a remote public server as a metadata repository. Ex. 1007, at 196, col. 1, ¶¶6, 10. Indeed, Takagi discloses a “system [that] consists of the three components[,] . . . end-user tools for problem reporting and client-side webpage transcoding based on external metadata; volunteer tools for metadata authoring; and a public server as the metadata repository.” Ex. 1007, at 196, col. 1, ¶6. Takagi’s approach also results in improved and effective generation of alternative text labels: “[t]he supporters resolved 85.1% of the 323 submitted

Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

requests and only 5.9% remained unresolved as of May 1, 2009. The requests are first labeled as “unanswered” and supporters can change the status with the authoring tool for supporters.” *Id.*, at 198, col. 1, ¶2. A POSITA would thus have been motivated to combine Takagi’s social labeling approach with Markel’s system, to extend Markel’s system for evaluating and correcting websites based on a ruleset which can be accessed over the internet via Takagi’s social labeling pilot system which makes use of a public server as a metadata repository.

481. A POSITA would also have expected this combination to succeed with minimal modification. Indeed, Markel already includes an several modules that analyze a website as a function of a ruleset of design, list items for automatic, semi-automatic and manual correction, and automatically, semi-automatically and manually correct the listed items. Ex. 1006, at Abstract. Markel also discloses using a “wizard” tool which provides “prompts and methods to resolve conflicts or violations.” *Id.*, at ¶99. A POSITA would have been motivated to combine the wizard of Markel with one of the interfaces, e.g., the quick fix wizard, of Takagi. The combination of the wizard of Markel and the quick fix wizard of Takagi, for example, would allow supporters to respond to users’ requests, quickly make alternative text metadata via a remote server with minimal work, and allow access to the rest of Markel system, such as its assessment, reporting and correction

Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

module, among other modules, which would work as intended (with improved results from using the Takagi's social labeling methods). Ex. 1007, at 197, col. 1, ¶¶3, 5; Ex. 1006, at Abstract, ¶¶71-76, 99.

482. Takagi also discloses WYSISYG-style editor, referred to as a “page map,” for metadata authoring but does not elaborate on the details of how untagged elements, e.g., inaccessible images include those without alternative text or with useless alternative text (such as “banner” or “spacer”), are detected. Ex. 1007 at 197, col. 1, ¶4. Thus a POSITA would have understood the benefit of combining the Takagi with the Markel system which discloses correcting code violations on a website including missing HTML tags, alt tag for images, a title tag, and labels for form fields. Ex. 1006, at Abstract, ¶¶72, 83, 88. As explained in Section IV above, similar remediation techniques were well known. *See also* Ex. 1005, at Tables I(a)-(b), II(a)-(b); Ex. 1015 at ¶¶2, 4, 15, 18.

483. Additionally, as I explained in Section IV, it was a commonly known approach to store alternative text in a remote server, where the alternative text was generated with the help manual remediation. Ex. 1008, at 186, col. 2 ¶¶3-5, Figure 4; Ex. 1007, at 186, col. 1, ¶6; *see also* Ex. 1031, at 2-4; Ex. 1032, at 177, col. 2, ¶¶3-4, Figure 4; Ex. 1033; Ex. 1034; Ex. 1035, at 1:18-36, Figure 1.

484. Therefore, it would have been obvious to a POSITA to combine Markel, which discloses a several modules for evaluating and correcting websites based on a design ruleset, with Takagi, which discloses inputting alternative text from a remote server machine.

2. Independent Claim 1

- a. **“A computer-implemented method of programmatically assigning a descriptive attribute to an untagged element on a web page to enable an audible description of the untagged element, the web page having an associated document object model (DOM), the method comprising:” (Claim 1[pre])**

485. To the extent the preamble is limiting, Markel and Takagi disclose this element. Takagi discloses the preamble for the reasons described above with respect to Ground 1, Claim 1[pre].

486. Markel also teaches one or more modules to correct (to programmatically assign) standards violations (a website including missing HTML tags, alt tag for images, a title tag, and labels for form fields) on a web page having an associated document object model. For example, Markel discloses an assessment module for parsing and analyzing a website based on a function of a ruleset of design guidelines. Ex. 1006, at Abstract. Markel’s ruleset of design guidelines comprise a set of standards, such as the Section 508 standard and a usability standard such as the W3C Web Content Accessibility Guideline. *Id.*, at

Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

Abstract, ¶73. The assessment module 114 identifies design errors and standards violations that can include missing HTML tags, alt tag for images, a title tag, and labels for form fields (e.g., input fields). *Id.*, at Abstract, ¶¶72, 83, 88. Markel's reporting module is used to list these items, e.g., the design errors and standards violations, that can be automatically corrected, that can use semi-automatic assisted correction, and that are for manual correction. *Id.*, at Abstract. Markel also discloses a correcting module for automatically, semi-automatically, or manually correcting the listed items. *Id.*

487. Furthermore, Markel's correction module includes:

[C]omponents for functional site recoding, error correcting, and validation. The primary function of the correction module 118 is automated execution or semi-automated wizard-assisted implementation, and support for manual correction of reported ruleset violations. The system 100 further provides additional functionality including fixing minor alignment problems or font specifications, or adding ALT tags to images after querying the user for appropriate label.

Ex. 1006, at ¶78.

488. Markel also discloses using a “web crawler 108 [which] communicates with the web server 110 and transmits a request thereto for the information located at the URL entered by the user,” where “the information sent represents a blueprint of the website requested and includes code such as HTML,

Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

Java Script or any other suitable programming language. The blueprint describes what the text and graphics of the website will look like and their positions on the website.” Ex. 1006, at ¶68. Furthermore, Markel’s system “executes a uniform process of diagnosis, reporting correction, and validation of source website code” where “[t]he system 100 removes standards violations in a rapid, reliable, consistent, and cost-effective manner.” *Id.*, at ¶70.

489. A POSITA would have understood a “blueprint of a website” to be a document object model that “describes what the text and graphics of the website will look like and their positions on the website.” At the time of Markel, the DOM was already the standard approach to “parse and analyze [a] website” that “includes code such as HTML, Java Script,” (Ex. 1006, at ¶¶50, 68) as “The DOM originated as a specification to allow JavaScript scripts and Java programs to be portable among Web browsers” and “Anything found in an HTML or XML document can be accessed, changed, deleted, or added using the Document Object Model.” *Id.*, at ¶¶50, 68; Ex. 1036, at ¶¶2, 10.

490. From at least the above disclosures, a person of ordinary skill in the art would have understood that Markel could be used to check and fix Document Object Models (DOM) associated with a webpage as well as the HTML code itself. To the extent Markel does not explicitly disclose Document Object Models (DOM)

Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

associated with a webpage, a person of ordinary skill in the art would have understood that Markel can be used to check and fix Document Object Models (DOM) associated with a webpage, e.g., “blueprint of a website,” as well as the HTML code, e.g., “source website code,” itself. Ex. 1006, at ¶¶68, 70. Thus, Markel discloses “[a] computer-implemented method of programmatically assigning a descriptive attribute to an untagged element on a web page to enable an audible description of the untagged element, the web page having an associated document object model (DOM)” as claimed.

491. Accordingly, Markel and Takagi disclose and render obvious the preamble of claim 1.

b. “dynamically analyzing, by a computer system, code associated with the web page, the code comprising at least the DOM or HTML code;” (Claim 1[a])

492. Markel and Takagi disclose this element. Takagi discloses this claim element for the reasons described above with respect to Ground 1, Claim 1[a].

493. Markel also discloses using a computer system to parsing, analyzing and diagnosing a source website code (the DOM) as a function of a ruleset of design guidelines. Ex. 1006, at Abstract, ¶70. Markel discloses “pars[ing] and analyz[ing] [] [a] website as a function of a ruleset of design guidelines.” Ex. 1006, at Abstract. Furthermore, Markel’s system “executes a uniform process of

Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

diagnosis, reporting correction, and validation of source website code” where “[t]he system 100 removes standards violations in a rapid, reliable, consistent, and cost-effective manner.” Ex. 1006, at ¶70. A POSITA would have understood “source website code” to refer to a website’s HTML code. As discussed in the claim 1[pre] above, to the extent Markel does not explicitly disclose Document Object Models (DOM) associated with a webpage, a person of ordinary skill in the art would have understood that Markel can be used to check and fix Document Object Models (DOM) associated with a webpage as well as the HTML code, e.g., and “source website code,” itself. *Id.*

494. Markel clearly also discloses using a computer system for analyzing and diagnosing a source website code, as shown at computer 102 in Figure 1 below:

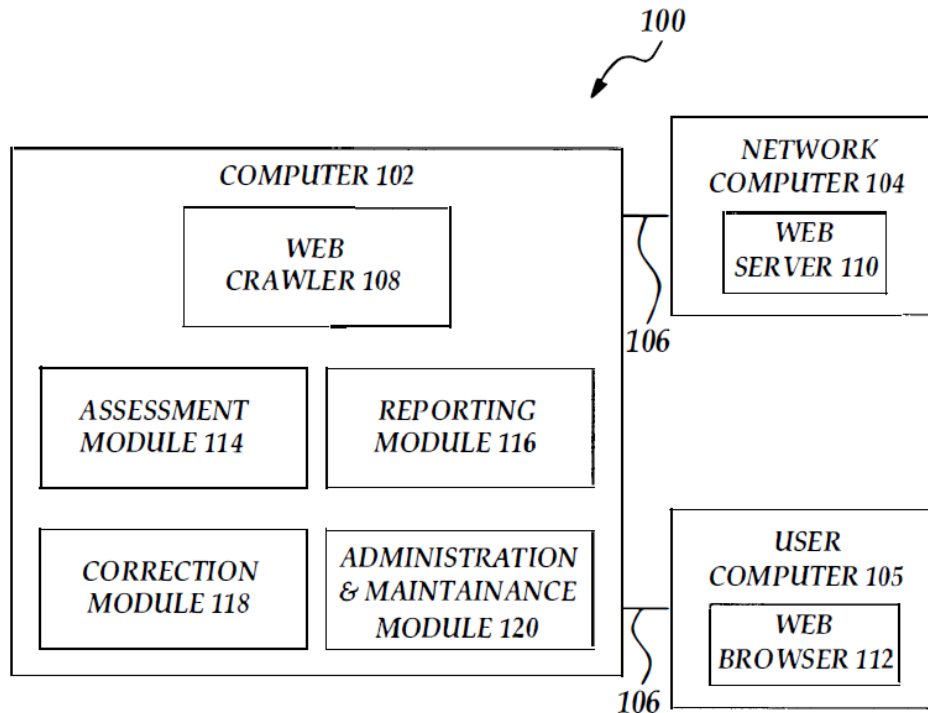


Figure 1

Ex. 1006, at Figure 1. Markel also describes its system as “preferably embodied in a computer program running on a general purpose computer or workstation 102.”

Ex. 1006, at ¶66.

495. Furthermore, Markel discloses:

[T]he computer 102 is connected to a network computer 104 or computers via a first network 106, such as the internet. The website or websites to be evaluated are located on the network computer 104. While the present invention is embodied in a piece of online software, it could be embodied in a stand alone piece of software, on a network, wide area network or any other suitable application

Ex. 1006, at ¶76.

Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

496. Additionally, Markel's system "evaluates and corrects dynamically generated web pages, which are generated real-time as a function of input from the user." Ex. 1006, at ¶105.

497. Therefore, Markel and Takagi thus disclose and render obvious this element.

- c. **"detecting, by the computer system, one or more compliance issues relating to web accessibility standards in the code, wherein at least one of the one or more compliance issues comprises the untagged element lacking an adequate descriptive attribute;" (Claim 1[b])**

498. Markel and Takagi disclose this element. Takagi discloses this claim element for the reasons described above with respect to Ground 1, Claim 1[b].

499. Markel further discloses identifying (detecting), by the computer system, standard violations (one or more compliance issues relating to web accessibility standards in the code), wherein the standard violations can include at least one of as missing HTML tags, alt tag for images, a title tag, and labels for form fields (untagged element lacking an adequate descriptive attribute). Markel discloses an assessment module which "identifies standards violations including . . . missing HTML tags" where "[t]he module 114 parses and analyzes the website as a function of a ruleset of design guidelines." Ex. 1006, at ¶72. Markel's assessment module identifies standards violations that can include missing HTML

Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

tags, alt tag for images, a title tag, and labels for form fields (e.g., input fields). *Id.*, at Abstract, ¶¶72, 76, 83, 88.

500. Furthermore, Markel’s assessment module also detects rule violations including for a title tag and labels for form fields. Ex. 1006, at Abstract, ¶87, 88. Markel discloses an “assessment module 114 [that] includes a method of selecting rules to apply.” *Id.*, at ¶87. This includes “rules that apply across the entire webpage [which] are detected, such as the presence of a title tag and labels for form fields.” *Id.*, at ¶88.

501. Markel’s system is “preferably embodied in a computer program running on a general purpose computer or workstation 102.” *Id.*, at ¶66. A POSITA would have understood that a computer system to include “computer program running on a general purpose computer or workstation.”

502. Thus, Markel discloses “detecting, by the computer system, one or more compliance issues relating to web accessibility standards in the code, wherein at least one of the one or more compliance issues comprises an untagged element, the untagged element is an element lacking an adequate descriptive attribute” as claimed.

503. Therefore, Markel and Takagi thus disclose and render obvious this element.

- d. “applying, by the computer system, one or more pre-existing remediation code to the one or more compliance issues, wherein the one or more pre-existing remediations is generated by a remote server system performing at least:” (Claim 1[c][1])**

504. Markel and Takagi disclose this element. Takagi discloses this claim element for the reasons described above with respect to Ground 1, Claim 1[c][1].

505. Markel also describes several modules for correcting code violations on a website including missing HTML tags, alt tag for images, a title tag, and labels for form fields. Ex. 1006, at Abstract, ¶¶72, 83, 88. Markel discloses an assessment module, a reporting module, and a correcting module, together used for evaluating and correcting websites. *Id.*, at Abstract. Specifically, Markel discloses an assessment module for parsing and analyzing a website based on a function of a ruleset of design guideline. *Id.* Markel’s ruleset of design guidelines comprise a set of standards, such as the Section 508 standard and a usability standard such as the W3C Web Content Accessibility Guideline. *Id.*, at Abstract, ¶73. Markel’s assessment module identifies standards violations that can include missing HTML tags, alt tag for images, a title tag, and labels for form fields (e.g., input fields). *Id.*, at Abstract, ¶¶72, 83, 88. Markel’s reporting module is used to list these items, e.g., the design errors and standards violations, that can be automatically corrected, that can use semi-automatic assisted correction, and that are for manual correction.

Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

Ex. 1006, at Abstract. Markel also discloses a correcting module for automatically, semi-automatically, or manually correcting the listed items. *Id.*

506. Markel allows for “semi-automated wizard-assisted manual implementation, and support for manual correction of reported ruleset violation.” Ex. 1006, at ¶78. For example, Markel’s system “provides additional functionality including fixing minor alignment problems or font specifications, or adding ALT tags to images after querying the user for appropriate labels.” *Id.* Thus, Markel’s system can be used for automatic, semi-automatic, or manual remediation of ruleset violations. *Id.*, at Abstract ¶¶72, 78, 83, 88.

507. Markel also discloses that access to Markel’s system “is provided over the internet and the graphical user interface 112 is embodied in a web page accessed via a web browser 108.” Ex. 1006, ¶104. A POSITA would have understood that access over the internet to disclose that Markel’s system resides on a remote server. Furthermore, referring to Figure 1 of Markel, the computer of Markel’s system is clearly accessed by a user through a network, e.g., such as the internet, and is separate from the user’s computer. To this end, a POSITA would have been motivated to improve Markel’s system for evaluating and correcting websites, which can be accessed over the internet with that of Takagi, to take

advantage of Takagi's social labeling pilot system which makes use of a public server as a metadata repository.

508. As discussed above, a POSITA would have been motivated to combine Takagi with Markel's system, to extend Markel's system for evaluating and correcting websites based on a ruleset which can be accessed over the internet via Takagi's social labeling pilot system which makes use of a public server as a metadata repository, and can therefore reduce the time and cost associated with remediating non-programmatically fixable compliance issues.

509. Thus, a POSITA would have understood Markel and Takagi to teach "applying, by the computer system, one or more pre-existing remediation code to the one or more compliance issues, wherein the one or more pre-existing remediations is generated by a remote server system performing at least."

510. Therefore, Markel and Takagi thus disclose and render obvious this element.

**e. "accessing the code associated with the web page"
(Claim 1[c][2])**

511. Markel and Takagi disclose this element. Takagi discloses this claim element for the reasons described above with respect to Ground 1, Claim 1[c][2].

512. Markel also discloses this element. Markel discloses using a "web crawler 108 [which] communicates with the web server 110 and transmits a

Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

request thereto for the information located at the URL entered by the user,” where “the information sent represents a blueprint of the website requested and includes code such as HTML, Java Script or any other suitable programming language. The blueprint describes what the text and graphics of the website will look like and their positions on the website.” Ex. 1006, at ¶¶68. A POSITA would have understood that requesting and sending “a blueprint of the website requested and includes code such as HTML, Java Script or any other suitable programming language” to disclose accessing the code associated with the web page.

513. Markel also discloses parsing and analyzing the code of a website. Ex. 1006, at Abstract, ¶¶50, 90. A POSITA would have understood parsing and analyzing to include accessing the code of website to include accessing the code of a webpage.

514. Moreover, like Takagi, Markel also discloses using a “wizard” tool which provides “prompts and methods to resolve conflicts or violations.” Ex. 1006, ¶¶99. A POSITA would have been motivated to combine the wizard of Markel with one of the interfaces, e.g., the WYSISYG-style editor or quick fix wizard, of Takagi. Ex. 1007, at 197, col. 1, ¶¶4, 5. In addition, a POSITA would have been motivated to extend Markel based using the social accessibility pilot service of Takagi. Specifically, a POSITA would have understood that Markel’s

Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

wizard tool could have easily been extended using Takagi's metadata authoring tool, e.g., the page map WYSISYG-style editor, to take advantage of the benefits provided by social accessibility pilot service as disclosed by Takagi. Ex. 1007, at 197, col. 1, ¶¶4, 5, Figure 2, 3.

515. Therefore, Markel and Takagi disclose and render obvious this element.

- f. **“receiving, by the remote server system, via a remediation interface input from a user, the input comprising a remediation action to manually remediate a non-programmatically-fixable compliance issue associated with the web page;”
(Claim 1[c][3])**

516. Markel and Takagi disclose this element. Takagi discloses this claim element for the reasons described above with respect to Ground 1, Claim 1[c][3].

517. Markel also discloses this element. Markel discloses a system that includes several modules for reporting, correcting, administration, maintenance for evaluating and correcting websites. Ex. 1006, ¶71, Abstract. Markel's system also includes web crawler and an interface embodied in a webpage. Ex. 1006, ¶71.

518. Markel discloses:

[T]he system: 100 includes an assessment module 114, a reporting module 116, a correction module 118 and an administration and maintenance module 120. In one embodiment, the web crawler 108, and the modules 114, 116,

Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

118, 120 are embodied in software and the graphical user interface 112 is embodied in a web page.
Ex. 1006, ¶71.

519. As shown in Figure 9 of Markel, Markel discloses an interface having a fix input text box 906 [that] allows the user to provide input for semi-automatic correction items” such as “an image is missing an ALT tag,” and in that instance, “the user is prompted to enter the appropriate information associated with the ALT tag.” Ex. 1006, ¶113. Markel also discloses that its system “identifies which items can be corrected automatically.” Ex. 1006, ¶91.

520. Furthermore, Markel also discloses using a “wizard” tool which provides “prompts and methods to resolve conflicts or violations.” Ex. 1006, ¶99.

521. Markel also discloses that access to Markel’s system “is provided over the internet and the graphical user interface 112 is embodied in a web page accessed via a web browser 108.” Ex. 1006, ¶104. A POSITA would have understood that access over the internet to disclose that Markel’s system resides on a remote server. Furthermore, referring to Figure 1 of Markel, the computer of Markel’s system is clearly accessed by a user through a network, e.g., such as the internet, and is separate from the user’s computer. To this end, a POSITA would have been motivated to improve Markel’s system for evaluating and correcting websites, which can be accessed over the internet with that of Takagi, to take

advantage of Takagi's social labeling pilot system which makes use of a public server as a metadata repository.

522. Thus, Markel discloses using an interface and wizard (a remediation interface) for receiving, via a remote server system and from a user, input for semi-automatic (manually remediated) correction items such as an image is missing an ALT tag including items that cannot be corrected automatically (a non-programmatically-fixable compliance issue associated with the web page).

523. Therefore, Markel and Takagi disclose "receiving, by the remote server system, via a remediation interface input from a user, the input comprising a remediation action to manually remediate a non-programmatically-fixable compliance issue associated with the web page" as claimed.

524. Therefore, Markel and Takagi thus disclose and render obvious this element.

- g. "generating, by the remote server system, the one or more pre-existing remediations based on the input comprising the remediation action to manually remediate a non-programmatically-fixable compliance issue; " (Claim 1[c][4])**

525. Markel and Takagi disclose this element. Takagi discloses this claim element for the reasons described above with respect to Ground 1, Claim 1[c][4].

Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

526. Markel also discloses this element. Markel discloses a system that includes several modules for reporting, correcting, administration, maintenance for evaluating and correcting websites. Ex. 1006, at ¶71, Abstract. Markel’s system also includes web crawler and an interface embodied in a webpage. *Id.*, at ¶71.

527. Markel discloses:

[T]he system: 100 includes an assessment module 114, a reporting module 116, a correction module 118 and an administration and maintenance module 120. In one embodiment, the web crawler 108, and the modules 114, 116, 118, 120 are embodied in software and the graphical user interface 112 is embodied in a web page.

Ex. 1006, at ¶71.

528. As shown in Figure 9 of Markel, Markel discloses an interface having a fix input text box 906 [that] allows the user to provide input for semi-automatic correction items” such as “an image is missing an ALT tag,” and in that instance, “the user is prompted to enter the appropriate information associated with the ALT tag.” *Id.*, at ¶113. Markel also discloses that its system “identifies which items can be corrected automatically.” *Id.*, at ¶91.

529. Furthermore, Markel also discloses using a “wizard” tool which provides “prompts and methods to resolve conflicts or violations.” *Id.*, at ¶99.

530. Markel also discloses that access to Markel’s system “is provided over the internet and the graphical user interface 112 is embodied in a web page

Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

accessed via a web browser 108.” Ex. 1006, at ¶104. A POSITA would have understood that access over the internet to disclose that Markel’s system resides on a remote server. Furthermore, referring to Figure 1 of Markel, the computer of Markel’s system is clearly accessed by a user through a network, e.g., such as the internet, and is separate from the user’s computer. To this end, a POSITA would have been motivated to improve Markel’s system for evaluating and correcting websites, which can be accessed over the internet with that of Takagi, to take advantage of Takagi’s social labeling pilot system which makes use of a public server as a metadata repository.

531. Thus, Markel discloses semi-automatically generating, by the remote server system, correction items such as an image missing an ALT tag including items that cannot be corrected automatically (one or more pre-existing remediations based on the input comprising the remediation action to manually remediate a non-programmatically-fixable compliance issue).

532. Therefore, Markel and Takagi disclose and render obvious this element.

- h. “storing, by the remote server system, the one or more pre-existing remediations in an electronic data storage medium; and” (Claim 1[c][5])**

Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

533. Markel and Takagi disclose this element. Takagi discloses this claim element for the reasons described above with respect to Ground 1, Claim 1[c][5].

534. Markel discloses this element. Markel's system includes an "administration and maintenance module 120 [which] contains components to support the storage, history, and long-term support of the website" and upon correcting the code, Markel's system provides that "appropriate information is saved in the code and the documentation." Ex. 1006, at ¶¶101, 114.

535. Therefore, Markel and Takagi thus disclose and renders obvious this element.

- i. **"assigning, by the computer system, a descriptive attribute to the untagged element in the code of the web page based on the one or more pre-existing remediations, the descriptive attribute assigned to the untagged element adapted to enable an assistive technology to speak the descriptive attribute to a user." (Claim 1[d])**

536. Markel and Takagi disclose this element. Takagi discloses this claim element for the reasons described above with respect to Ground 1, Claim 1[d].

537. Markel also discloses fixing using a correction module (assigning), by the computer system, labels for untagged elements such as adding ALT tags to images (a descriptive attribute to the untagged element in the code of the web page) based on a ruleset of design guidelines (one or more pre-existing

Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

remediations), to enable website providers to diagnose, evaluate, report, and retrofit code violations existing in websites meet both in-house and industry design standards addressing accessibility issues providing benefits to all users (descriptive attribute assigned to the untagged element adapted to enable an assistive technology to speak the descriptive attribute to a user).

538. As described above, Markel describes several modules for correcting code violations on a website including missing HTML tags, alt tag for images, a title tag, and labels for form fields. Ex. 1006, at Abstract, ¶¶72, 83, 88. For example, Markel discloses an assessment module, a reporting module, and a correcting module, together used for evaluating and correcting websites. In particular, Markel discloses a correcting module for automatically, semi-automatically, or manually correcting and/or fixing identified standards violations that can include missing HTML tags, alt tag for images, a title tag, and labels for form fields. *Id.* Markel's system "provides additional functionality including fixing minor alignment problems or font specifications, or adding ALT tags to images after querying the user for appropriate labels." Ex. 1006, at ¶78. A POSITA would have understood correcting and/or fixing identified standards violations to disclose assigning a descriptive attribute to the untagged element in the code of the web page based on the one or more pre-existing remediations.

Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

539. Markel also recognized that sightless users must employ dedicated browsers which can speak text or a Braille reader to access the web and consume information. *Id.*, at ¶31. From this disclosure, and a POSITA's understanding of web accessibility generally, a POSITA would have recognized that the descriptive attribute Markel assigned to the untagged element was adapted to enable an assistive technology to speak the descriptive attribute to a user.

540. Also, Markel's system is "preferably embodied in a computer program running on a general purpose computer or workstation 102." Ex. 1006, at ¶66. *See also* Ex. 1006, at Figure 1, ¶76. A POSITA would have understood that "a computer program running on a general purpose computer or workstation" to disclose a "computer system." *Id.*

541. As discussed above, Markel describes several modules for correcting code violations on a website including missing HTML tags, alt tag for images, a title tag, and labels for form fields. Ex. 1006, at Abstract, ¶¶72, 83, 88.

Furthermore, Markel's system "evaluates and corrects dynamically generated web pages, which are generated real-time as a function of input from the user." *Id.*, at ¶105. A POSITA would have understood that evaluating and correcting code violations in dynamically generated web pages as assigning a descriptive attribute to the untagged element in the code of the web page.

542. Therefore Markel and Takagi disclose “assigning, by the computer system, a descriptive attribute to the untagged element in the code of the web page based on the one or more pre-existing remediations, the descriptive attribute assigned to the untagged element adapted to enable an assistive technology to speak the descriptive attribute to a user” as claimed.

3. Claim 2

543. Claim 2 depends on claim 1 and recites “**wherein the one or more pre-existing remediations are javascript.**” Markel and Takagi disclose this claim. Takagi discloses this claim for the reasons described above with respect to Ground 1, Claim 2.

4. Claim 3

544. Claim 3 depends on claim 1 and recites the additional requirement of: “**wherein the untagged element is an input field.**” Markel and Takagi disclose this claim. Takagi discloses this claim for the reasons described above with respect to Ground 1, Claim 3.

545. Markel further discloses this claim because Markel’s assessment module parses and analyzes rule violations including labels for form fields (e.g., an input field is a form field). Ex. 1006, at Abstract, ¶¶87-88. Markel discloses an “assessment module 114 [that] includes a method of selecting rules to apply.” *Id.*,

at ¶87. This includes “rules that apply across the entire webpage [which] are detected, such as the presence of a title tag and labels for form fields.” *Id.*, at ¶88. A POSITA would have understood from Markel’s detecting the presence of labels for form fields meant that it could detect untagged input fields (i.e., form fields that lack a corresponding label).

546. Therefore, Markel and Takagi thus disclose and render this claim obvious.

5. Claim 4

547. Claim 4 depends on claim 1 and recites the additional requirement of: **“wherein the untagged element is an image.”** Markel and Takagi disclose this claim. Takagi discloses this claim for the reasons described above with respect to Ground 1, Claim 4.

548. Markel further discloses this element. Markel describes several modules for correcting code violations on a website including missing HTML tags, alt tag for images, a title tag, and labels for form fields. Ex. 1006, at Abstract, ¶¶72, 83, 88. Markel discloses an “assessment module 114 [which] applies [] [a] ruleset of design guidelines.” *Id.*, at ¶81. The ruleset of design guidelines that includes “correction items [which] may [or may not] be automatically identified and corrected” including “the absence of an HTML tag such as an ALT tag for

images.” Ex. 1006, at Abstract, ¶83. A POSITA would have understood an image having an absent ALT tag to be an untagged element.

549. Markel and Takagi thus disclose and render this claim obvious.

6. Claim 6

550. Claim 6 depends on claim 1 and recites the additional requirement of:
“wherein the untagged element lacks the adequate descriptive attribute when one or more descriptive attributes of the untagged element are erroneous.”

Markel and Takagi disclose this claim. Takagi discloses this claim for the reasons described above with respect to Ground 1, Claim 6.

551. Markel discloses this claim. Markel’s system “diagnosis, reporting corrections, and validates of source website code,” where the system “focuses on code [that] has design errors.” Ex. 1006, at ¶70. Markel discloses an “assessment module 114 [that] identifies design errors and standards violations including syntax errors or the misuse of quotes, poor contrast, and missing HTML tags.” *Id.*, at ¶72. Furthermore, Markel’s system provides “error attributes” where “the source code may be annotated through a technique whereby the code is tagged to assist and guide the automated evaluation.” *Id.* Furthermore, Markel’s Figure 9 shows an image has an alt tag which is erroneously empty. *Id.*, at Figure 9. A POSITA

would have understood that an image having an alt tag which is erroneously empty
to be erroneous. Annotated Figure 9 of Markel is shown below:

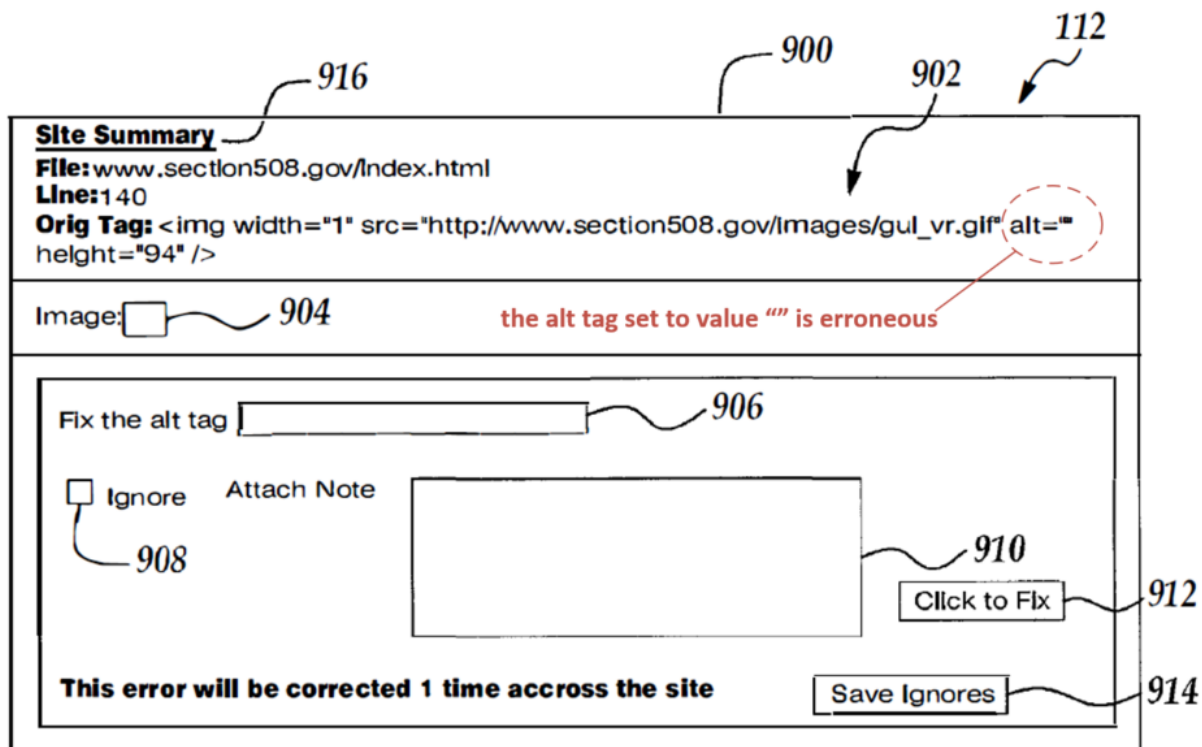


Figure 9

Ex. 1006, at Figure 9 (annotated).

552. Therefore, Markel and Takagi thus disclose and render this claim obvious.

7. Claim 7

Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

553. Claim 7 depends on claim 1 and recites the additional requirement of:
“wherein the assigning comprises changing the DOM or HTML code associated with the web page.” Markel and Takagi disclose this claim. Takagi discloses this claim for the reasons described above with respect to Ground 1, Claim 7.

554. Markel further discloses this element. Markel discloses that through a correction screen in its interface, a user can select a “fix button,” after which “the code is corrected.” Ex. 1007, at ¶¶113-114, Figure 9. Markel further discloses that “[o]nce the correction tool has been applied to a site, a new version of the site files is generated with the problems rectified.” *Id.*, ¶96. A POSITA would have understood that correcting the code and providing a new version of the web site which is generated with accessibility problems rectified, to disclose changing the DOM or HTML code associated with the web page.

555. Furthermore, Markel discloses “enable[ing] website providers to . . . retrofit [source] code violations existing in websites.” Ex. 1006, at Abstract. A POSITA would have understood that retrofitting source code violations to a website to disclose changing the DOM or HTML code associated with the web page.

556. Therefore, Markel and Takagi thus disclose and render this claim obvious.

8. Claim 8

557. Claim 8 depends on claim 1 and recites the additional requirement of: **“wherein the descriptive attribute is an HTML label.”** Markel and Takagi disclose this claim. Takagi discloses this claim for the reasons described above with respect to Ground 1, Claim 8.

558. Markel further discloses this claim. Markel discloses this claim because Markel’s assessment module parses and analyzes rule violations including labels for form fields (e.g., an input field is a form field). Ex. 1006, at Abstract, ¶¶87-88. Markel discloses an “assessment module 114 [that] includes a method of selecting rules to apply.” *Id.*, at ¶87. This includes “rules that apply across the entire webpage [which] are detected, such as the presence of a title tag and labels for form fields.” *Id.*, at ¶88. A POSITA would have understood that labels for form fields include an HTML label.

559. Therefore, Markel and Takagi thus disclose and render this claim obvious.

9. Claim 9

560. Claim 9 depends on claim 1 and recites the additional requirement of:
“wherein the descriptive attribute is an alt-text label.” Markel and Takagi disclose this claim. Takagi discloses this claim for the reasons described above with respect to Ground 1, Claim 9.

561. Markel further discloses this claim. Markel describes several modules for correcting code violations on a website including missing HTML tags, alt tag for images, a title tag, and labels for form fields. Ex. 1006, at Abstract, ¶¶72, 83, 88.

562. Markel discloses an “assessment module 114 [which] applies [] [a] ruleset of design guidelines.” *Id.*, at ¶81. The ruleset of design guidelines that includes “correction items [which] may [or may not] be automatically identified and corrected” including “the absence of an HTML tag such as an ALT tag for images.” *Id.*, at Abstract, ¶83. A POSITA would have understood that an alt-text label includes ALT tag for images.

563. Therefore, Markel and Takagi thus disclose and render this claim obvious.

10. Claim 10

564. Claim 10 depends on claim 1 and recites the additional requirement of: **“wherein the computer system comprises one or more computing systems.”**

Markel and Takagi disclose this claim. Takagi discloses this claim for the reasons described above with respect to Ground 1, Claim 10.

565. Markel further discloses this claim. Markel discloses this element because Markel's computer system includes at least one computer system, as clearly shown in Markel Figure 1 below (Ex. 1006, at Figure 1):

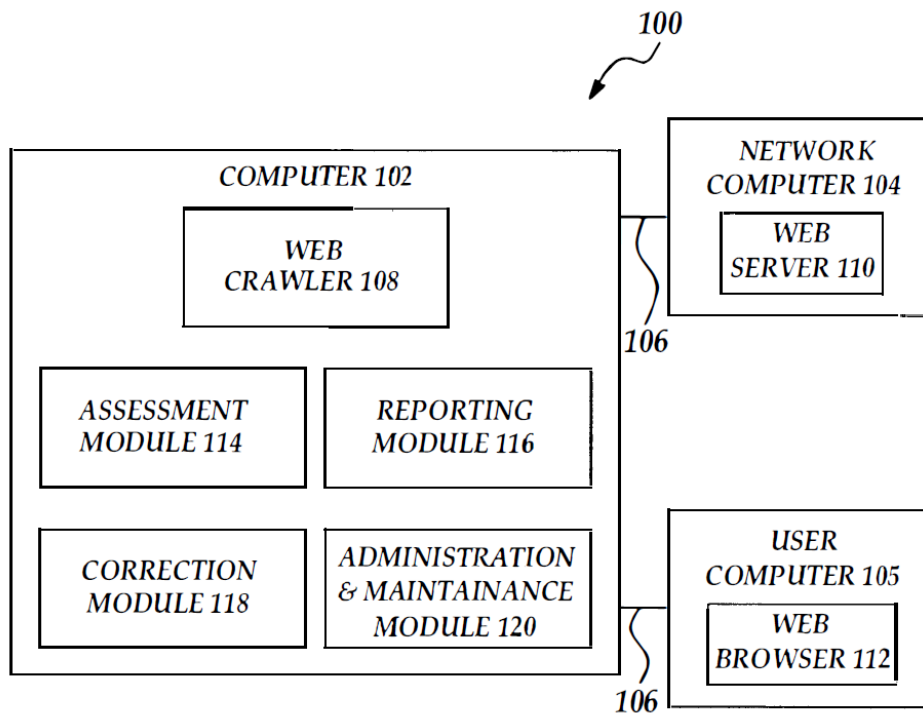


Figure 1

Ex. 1006, at Figure 1.

566. Therefore, Markel and Takagi thus disclose and render this claim obvious.

11. Independent Claim 11

- a. **“A non-transitory electronic storage medium with computer code stored thereon, the computer code configured to programmatically assign a descriptive attribute to an element on a web page to enable an audible description of the element, the web page having an associated document object model (DOM), and the computer code configured to perform, when executed by a computer processor, the steps of:”
(Claim 11[pre])**

567. To the extent the preamble is limiting, Markel and Takagi disclose this element. Markel and Takagi disclose this element for the reasons described above for Ground 4, Claim 1[pre]. A POSITA would have understood that a computer implemented method (as claimed in claim 1) was routinely performed through computer code stored on a non-transitory electronic medium. Additionally, the recital of a “non-transitory computer readable information storage media” was well-known in the art and would have been obvious to a POSITA. Furthermore, a POSITA would have understood that a descriptive attribute to an element on a web page to include an untagged element as described in claim 1[pre].

568. Markel and Takagi thus disclose and render obvious the preamble.

- b. **“dynamically analyzing, by a computer system, code associated with the web page, the code comprising at least the DOM or HTML code” (Claim 11[a])**

569. Markel and Takagi disclose this element for the reasons described above with respect to Ground 4, Claim element 1[a]. Accordingly, Markel and Takagi thus disclose or render this element obvious.

- c. **“detecting, by the computer system, one or more compliance issues relating to web accessibility standards in the code, wherein at least one of the one or more compliance issues comprises the element lacking an adequate descriptive attribute;” (Claim 11[b])**

570. Markel and Takagi disclose this element for the reasons described above with respect to Ground 4, Claim element 1[b]. Accordingly, Markel and Takagi thus disclose or render this element obvious.

- d. **“applying, by the computer system, one or more pre-existing remediations to the one or more compliance issues, wherein the one or more pre-existing remediations is generated by a remote server system performing at least:” (Claim 11[c][1])**

571. Markel and Takagi disclose this element for the reasons described above with respect to Ground 4, Claim element 1[c][1]. Accordingly, Markel and Takagi thus disclose or render this element obvious.

- e. **“accessing the code associated with the web page;” (Claim 11[c][2])**

572. Markel and Takagi in disclose this element for the reasons described above with respect to Ground 4, Claim element 1[c][2]. Accordingly, Markel and Takagi thus disclose or render this element obvious.

- f. “receiving, by the remote server system, via a remediation interface input from a user, the input comprising a remediation action to manually remediate a non-programmatically-fixable compliance issue associated with the web page;” (Claim 11[c][3])**

573. Markel and Takagi disclose this element for the reasons described above with respect to Ground 4, Claim element 1[c][3]. Accordingly, Markel and Takagi thus disclose or render this element obvious.

- g. “generating, by the remote server system, the one or more pre-existing remediations based on the input comprising the remediation action to manually remediate a non-programmatically-fixable compliance issue;” (Claim 11[c][4])**

574. Markel and Takagi disclose this claim for the reasons described above with respect to Ground 4, Claim element 1[c][4]. Accordingly, Markel and Takagi thus disclose or render this element obvious.

- h. “storing, by the remote server system, the one or more pre-existing remediations in an electronic data storage medium; and” (Claim 11[c][5])**

575. Markel and Takagi disclose this claim for the reasons described above with respect to Ground 4, Claim element 1[c][5]. Accordingly, Markel and Takagi thus disclose or render this element obvious.

- i. “assigning, by the computer system, a descriptive attribute to the element in the code of the web page based on the one or more pre-existing remediations, the descriptive attribute assigned to the element**

**adapted to enable an assistive technology to speak the
descriptive attribute to a user.” (Claim 11[d])**

576. Markel and Takagi disclose this claim for the reasons described above with respect to Ground 4, Claim element 1[d]. Accordingly, Markel and Takagi thus disclose or render this element obvious.

12. Claim 12

577. Claim 12 depends on claim 11 and recites the additional requirement of: **“wherein the one or more pre-existing remediations are javascript.”** Markel and Takagi disclose this claim for the reasons described above with respect to Ground 4, Claim 2. Accordingly, Markel and Takagi thus disclose or render this claim obvious.

13. Claim 13

578. Claim 13 depends on claim 11 and recites the additional requirement of: **“wherein the element is an input field.”** Markel and Takagi disclose this claim for the reasons described above with respect to Ground 4, Claim 3. Accordingly, Markel and Takagi thus disclose or render this claim obvious.

14. Claim 14

579. Claim 14 depends on claim 11 and recites the additional requirement of: **“wherein the element is an image.”** Markel and Takagi disclose this claim for

Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

the reasons described above with respect to Ground 4, Claim 4. Accordingly,
Markel and Takagi in thus disclose or render this claim obvious.

15. Claim 16

580. Claim 16 depends on claim 11 and recites the additional requirement of: **“wherein the element lacks the adequate descriptive attribute when one or more descriptive attributes of the element are erroneous.”** Markel and Takagi disclose this claim for the reasons described above with respect to Ground 4, Claim 6. Accordingly, Markel and Takagi thus disclose or render this claim obvious.

16. Claim 17

581. Claim 17 depends on claim 11 and recites the additional requirement of: **“wherein the assigning comprises changing the DOM or HTML code associated with the web page.”** Markel and Takagi disclose this claim for the reasons described above with respect to Ground 4, Claim 7. Accordingly, Markel and Takagi thus disclose or render this claim obvious.

17. Claim 18

582. Claim 18 depends on claim 11 and recites the additional requirement of: **“wherein the descriptive attribute is an HTML label.”** Markel and Takagi disclose this claim for the reasons described above with respect to Ground 4, Claim 8. Accordingly, Markel and Takagi thus disclose or render this claim obvious.

18. Claim 19

583. Claim 19 depends on claim 11 and recites the additional requirement of: **“wherein the descriptive attribute is an alt-text label.”** Markel and Takagi disclose this claim for the reasons described above with respect to Ground 4, Claim 9. Accordingly, Markel and Takagi thus disclose or render this claim obvious.

19. Claim 20

584. Claim 20 depends on claim 11 and recites the additional requirement of: **“wherein the computer system comprises one or more computing systems.”** Markel and Takagi disclose this claim for the reasons described above with respect to Ground 4, Claim 10. Accordingly, Markel and Takagi thus disclose or render this claim obvious.

F. GROUND 5: Claims 1-4, 6-14, 16-20 are Unpatentable as Obvious Over Markel and Takagi in view of Hendry.

1. Motivation to Combine

585. A POSITA would have been motivated to combine Markel, and Takagi in view of Hendry in order to improve upon Markel’s modules for diagnosing, evaluating, reporting, and retrofitting code violations existing in websites, e.g., such as missing HTML tags, alt tag for images, a title tag, and labels for form fields. Markel provides several modules for automatically, semi-automatically, or manually remediating web pages, and can be further benefit

Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

from, and take advantage of, collaborative metadata authoring and the powerful functions of a remote server that were commonplace by the time of the '532 patent.

586. A POSITA would have been motivated to combine Markel and Takagi for the reasons described above with respect to Ground 4.

587. Similar to combining Markel with Takagi, a POSITA would have been motivated to combine Hendry's server machine functionalities in fixing the Web accessibility issues with the systems of Markel and Takagi. In general, Hendry teaches systems and methods for modifying documents (e.g., web pages; *see* Ex. 1009, at ¶42) to improve document accessibility implemented on a client machine or server machine. *Id.* For example, Hendry teaches that its methods can be performed on: (1) a local client computer, (2) a remote server, or (3) a combination of the two. *Id.* Hendry also teaches that its "[e]xamples [that] refer to a specific machine performing the steps [are merely] for the purposes of clarity and understanding" and that any step can be performed on any machine. *Id.* Hendry teaches that the various steps recited can be repeated, omitted, or performed in different order. *Id.*

588. Specifically, Hendry discloses that its "document modifier (120) is executed on the server machine (110) in response to receiving a request for a document from the client machine (102)" and that the "document modifier (120)

Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

on the server machine (110) determines that the particular requested document (114) is not accessibility enabled during an analysis of the particular requested document (114). The document (114) is analyzed to identify tags that are not recognizable by assistive technologies (106). The server machine (110) executes the document modifier (120) to produce a modified document (116) which is an accessibility-enabled version of the particular requested document (114). In response to receiving a request from the client machine (102) for the particular document (114), the server machine (110) sends the modified document (116).” Ex. 1009, at ¶53.

589. Hendry further discloses that “[t]he server machine (110) includes a data repository (112) storing one or more of: documents (114), modified documents (116) and accessibility features (118). The data repository (112) corresponds to any data storage device (for example, local memory on the server machine (110), web servers connected over the internet, machines within a local area network, a memory on a mobile device, local memory on the client machine (102), etc.).” *Id.*, at ¶41. Further Hendry discloses “[t]hose skilled in the art will appreciate that elements or various portions of data stored in the data repository (112) may be distributed (for example, stored on client machine (102) or at other servers)” where “the data repository (112) includes [a] flat, hierarchical, network

Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

based, relational, dimensional, object modeled, or data files structured otherwise”
and “is maintained as a table of a SQL database.” *Id.*, at ¶41.

590. A POSITA would also have expected this combination to succeed with minimal modification. Indeed, Markel discloses correcting the code of a website using its correction module (e.g., a software tool), and once the correction has been applied by the correction module, a new version of the website is generated with accessibility problems rectified. Ex. 1007, at ¶¶96, 113-114, Figure 9. Specifically, Markel discloses that through a correction screen in its interface, a user can select a “fix button,” after which “the code is corrected.” Ex. 1007, at ¶¶113-114, Figure 9. Markel further discloses that “[o]nce the correction tool has been applied to a site, a new version of the site files is generated with the problems rectified.” *Id.*, at ¶96. A POSITA would have understood that the modified documents that Hendry discloses, are the same as the new website files that Markel describes, and thus the techniques of Markel and Hendry are compatible. Therefore, as stated above, a POSITA expect minimal modification to combine the teachings of Hendry to Markel, to get the benefit of improving document accessibility implemented via a combination of client and server machines that Hendry discloses.

Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

591. Furthermore, Markel already includes an several modules that analyze a website as a function of a ruleset of design, list items for automatic, semi-automatic and manually correction, and automatically, semi-automatically and manually correct the listed items. A POSITA could simply modify Markel's code to also query a database for matching descriptive attributes that had been authored by others. A POSITA could also add an interface (e.g., a quick fix wizard in Takagi) that allows supporters to respond to users' requests and quickly make alternative text metadata via a remote server with minimal work and the rest of Markel system, such as its assessment, reporting and correction module, among other modules, that would work as intended (with improved results from using the Takagi social labeling methods). Ex. 1007, at 197, ¶¶3, 5; Ex. 1006, at Abstract, ¶¶71-76.

592. Takagi also discloses WYSISYG-style editor, referred to as a "page map," for metadata authoring but does not elaborate on the details of how untagged elements, e.g., inaccessible images include those without alternative text or with useless alternative text (such as "banner" or "spacer"), are detected. Ex. 1007, at 197, ¶4. Thus a POSITA would have understood the benefit of combining the Takagi with the Markel system which discloses correcting code violations on a website including missing HTML tags, alt tag for images, a title tag, and labels for

Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

form fields. Ex. 1006, at Abstract, ¶¶72, 83, 88. Similar remediation techniques were well known, *see also* Ex. 1005, at Tables I (a)-(b), II(a)-(b); Ex. 1015, at ¶¶2, 4, 15, 18.

593. Furthermore, a POSITA would have understood that collaborative web accessibility improvements leveraged by the participation of an open community were well known by the time of the '532 patent. Ex. 1007, at 196, col. 1, ¶6. For example, as I explained in Section IV above, various types of social approaches in addition to Takagi, such as the 1997 ALT-server, Teraguchi, Asakawa, WebInSight (Bigham), and WebVisum, could provide an improvement over the existing automated accessibility enhancements described by Markel. Ex. 1007, at 196, col. 1, ¶6; Ex. 1031, at 2-4; Ex. 1032, at 177, col. 2, ¶¶3-4, Figure 4; Ex. 1033; Ex. 1034; Ex. 1035, at 1:18-36, Figure 1.

594. Additionally, as I explained in Section IV, it was a commonly known approach to store alternative text in a remote server, where the alternative text was generated with the help manual remediation. Ex. 1016, at 4, Ex. 1008, at 186, col. 2, ¶¶1-3, Figure 4; Ex. 1032, at 177, col. 2, ¶¶3-4; Ex. 1007, at 196, col. 1, ¶6.

595. Therefore, it would have been obvious to a POSITA to combine Markel, which discloses a several modules for evaluating and correcting websites

based on a design ruleset, with Takagi and Hendry, both of which disclose
inputting alternative text from a remote server machine.

2. Independent Claim 1

- a. **“A computer-implemented method of programmatically assigning a descriptive attribute to an untagged element on a web page to enable an audible description of the untagged element, the web page having an associated document object model (DOM), the method comprising:” (Claim 1[pre])**

596. To the extent the preamble is limiting, Markel and Takagi in view of Hendry disclose this element. Markel and Takagi disclose this claim for the reasons described above with respect to Ground 4, claim element 1[pre]. Hendry discloses the preamble for the reasons described above for Ground 2, Claim 1[pre].

- b. **“dynamically analyzing, by a computer system, code associated with the web page, the code comprising at least the DOM or HTML code;” (Claim 1[a])**

597. Markel and Takagi in view of Hendry disclose this element. Markel and Takagi disclose this element for the reasons described above with respect to Ground 4, claim element 1[a]. Hendry discloses this claim element for the reasons described above for Ground 2, Claim 1[a].

- c. **“detecting, by the computer system, one or more compliance issues relating to web accessibility standards in the code, wherein at least one of the one or more compliance issues comprises the untagged element lacking an adequate descriptive attribute;” (Claim 1[b])**

Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

598. Markel and Takagi in view of Hendry disclose this element. Markel and Takagi disclose this element for the reasons described above with respect to Ground 4, claim element 1[b]. Hendry discloses this claim element for the reasons described above for Ground 2, Claim 1[b].

- d. **“applying, by the computer system, one or more pre-existing remediation code to the one or more compliance issues, wherein the one or more pre-existing remediations is generated by a remote server system performing at least:” (Claim 1[c][1])**

599. Markel and Takagi in view of Hendry disclose this element. Markel and Takagi disclose this element for the reasons described above with respect to Ground 4, claim element 1[c][1]. Hendry discloses this claim element for the reasons described above for Ground 2, Claim 1[c][1].

- e. **“accessing the code associated with the web page” (Claim 1[c][2])**

600. Markel and Hendry in view of Takagi disclose this element. Markel and Takagi disclose this element for the reasons described above with respect to Ground 4, claim element 1[c][2]. Hendry discloses this claim element for the reasons described above for Ground 2, Claim 1[c][2].

- f. **“receiving, by the remote server system, via a remediation interface input from a user, the input comprising a remediation action to manually remediate a non-programmatically-fixable**

**compliance issue associated with the web page;”
(Claim 1[c][3])**

601. Markel and Takagi in view of Hendry disclose this element. Markel and Takagi disclose this element for the reasons described above with respect to Ground 4, claim element 1[c][3]. Hendry discloses this claim element for the reasons described above for Ground 2, Claim 1[c][3].

- g. “generating, by the remote server system, the one or more pre-existing remediations based on the input comprising the remediation action to manually remediate a non-programmatically-fixable compliance issue; ” (Claim 1[c][4])**

602. Markel and Takagi in view of Hendry disclose this element. Markel and Takagi disclose this element for the reasons described above with respect to Ground 4, claim element 1[c][4]. Hendry discloses this claim element for the reasons described above for Ground 2, Claim 1[c][4].

- h. “storing, by the remote server system, the one or more pre-existing remediations in an electronic data storage medium; and” (Claim 1[c][5])**

603. Markel and Takagi in view of Hendry disclose this element. Markel and Takagi disclose this element for the reasons described above with respect to Ground 4, claim element 1[c][5]. Hendry discloses this claim element for the reasons described above for Ground 2, Claim 1[c][5].

- i. “assigning, by the computer system, a descriptive attribute to the untagged element in the code of the**

web page based on the one or more pre-existing remediations, the descriptive attribute assigned to the untagged element adapted to enable an assistive technology to speak the descriptive attribute to a user.” (Claim 1[d])

604. Markel and Takagi in view of Hendry disclose this element. Markel and Takagi disclose this element for the reasons described above with respect to Ground 4, claim element 1[d]. Hendry discloses this claim element for the reasons described above for Ground 2, Claim 1[d].

3. Claim 2

605. Claim 2 depends on claim 1 and recites “**wherein the one or more pre-existing remediations are javascript.**” Markel and Takagi in view of Hendry disclose this claim. Markel and Takagi disclose this claim for the reasons described above with respect to Ground 4, Claim 2. Hendry discloses this claim for the reasons described above for Ground 2, Claim 2.

4. Claim 3

606. Claim 3 depends on claim 1 and recites the additional requirement of: “**wherein the untagged element is an input field.**” Markel and Takagi in view of disclose this claim. Markel and Takagi disclose this claim for the reasons described above with respect to Ground 4, Claim 3. Hendry discloses this claim for the reasons described above for Ground 2, Claim 3.

5. Claim 4

607. Claim 4 depends on claim 1 and recites the additional requirement of:
“wherein the untagged element is an image.” Markel and Takagi in view of
Hendry disclose this claim. Markel and Takagi disclose this claim for the reasons
described above with respect to Ground 4, Claim 4. Hendry discloses this claim
for the reasons described above for Ground 2, Claim 4.

6. Claim 6

608. Claim 6 depends on claim 1 and recites the additional requirement of:
**“wherein the untagged element lacks the adequate descriptive attribute when
one or more descriptive attributes of the untagged element are erroneous.”**
Markel and Takagi in view of Hendry disclose this claim. Markel and Takagi
disclose this claim for the reasons described above with respect to Ground 4, Claim
6.

7. Claim 7

609. Claim 7 depends on claim 1 and recites the additional requirement of:
**“wherein the assigning comprises changing the DOM or HTML code
associated with the web page.”** Markel and Takagi in view of Hendry disclose
this claim. Markel and Takagi disclose this claim for the reasons described above
with respect to Ground 4, Claim 7.

8. Claim 8

Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

610. Claim 8 depends on claim 1 and recites the additional requirement of:
“wherein the descriptive attribute is an HTML label.” Markel and Hendry in view of Takagi disclose this claim. Markel and Takagi disclose this claim for the reasons described above with respect to Ground 4, Claim 8. Hendry discloses this claim for the reasons described above for Ground 2, Claim 8.

9. Claim 9

611. Claim 9 depends on claim 1 and recites the additional requirement of:
“wherein the descriptive attribute is an alt-text label.” Markel and Takagi in view of Hendry disclose this claim. Markel and Takagi disclose this claim for the reasons described above with respect to Ground 4, Claim 9. Hendry discloses this claim for the reasons described above for Ground 2, Claim 9.

10. Claim 10

612. Claim 10 depends on claim 1 and recites the additional requirement of: **“wherein the computer system comprises one or more computing systems.”** Markel and Takagi in view of Hendry disclose this claim. Markel and Takagi disclose this claim for the reasons described above with respect to Ground 4, Claim 10. Hendry discloses this claim for the reasons described above for Ground 2, Claim 10.

11. Independent Claim 11

- a. **“A non-transitory electronic storage medium with computer code stored thereon, the computer code configured to programmatically assign a descriptive attribute to an element on a web page to enable an audible description of the element, the web page having an associated document object model (DOM), and the computer code configured to perform, when executed by a computer processor, the steps of:”
(Claim 11[pre])**

613. To the extent the preamble is limiting, Markel and Takagi disclose this element. Markel and Takagi disclose this element for the reasons described above for Ground 5, Claim 1[pre]. A POSITA would have understood that a computer implemented method (as claimed in claim 1) was routinely performed through computer code stored on a non-transitory electronic medium. Additionally, the recital of a “non-transitory computer readable information storage media” was well-known in the art and would have been obvious to a POSITA. Furthermore, a POSITA would have understood that a descriptive attribute to an element on a web page to include an untagged element as described in element 1[pre].

614. Markel and Takagi in view of Hendry thus disclose and render obvious the preamble.

- b. **“dynamically analyzing, by a computer system, code associated with the web page, the code comprising at least the DOM or HTML code” (Claim 11[a])**

Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

615. Markel and Takagi in view of Hendry disclose this element for the reasons described above with respect to Ground 5, Claim element 1[a].

Accordingly, Markel and Takagi in view of Hendry thus disclose or render this element obvious.

- c. **“detecting, by the computer system, one or more compliance issues relating to web accessibility standards in the code, wherein at least one of the one or more compliance issues comprises the element lacking an adequate descriptive attribute;” (Claim 11[b])**

616. Markel and Takagi in view of Hendry disclose this element for the reasons described above with respect to Ground 5, Claim element 1[b].

Accordingly, Markel and Takagi in view of Hendry thus disclose or render this element obvious.

- d. **“applying, by the computer system, one or more pre-existing remediations to the one or more compliance issues, wherein the one or more pre-existing remediations is generated by a remote server system performing at least:” (Claim 11[c][1])**

617. Markel and Takagi in view of Hendry disclose this element for the reasons described above with respect to Ground 5, Claim element 1[c][1].

Accordingly, Markel and Takagi in view of Hendry thus disclose or render this element obvious.

**e. “accessing the code associated with the web page;”
(Claim 11[c][2])**

618. Markel and Takagi in view of Hendry disclose this element for the reasons described above with respect to Ground 5, Claim element 1[c][2].

Accordingly, Markel and Takagi in view of Hendry thus disclose or render this element obvious.

**f. “receiving, by the remote server system, via a remediation interface input from a user, the input comprising a remediation action to manually remediate a non-programmatically-fixable compliance issue associated with the web page;”
(Claim 11[c][3])**

619. Markel and Takagi in view of Hendry disclose this element for the reasons described above with respect to Ground 5, Claim element 1[c][3].

Accordingly, Markel and Takagi in view of Hendry thus disclose or render this element obvious.

g. “generating, by the remote server system, the one or more pre-existing remediations based on the input comprising the remediation action to manually remediate a non-programmatically-fixable compliance issue;” (Claim 11[c][4])

620. Markel and Takagi in view of Hendry disclose this element for the reasons described above with respect to Ground 5, Claim element 1[c][4].

Accordingly, Markel and Takagi in view of Hendry thus disclose or render this element obvious.

- h. **“storing, by the remote server system, the one or more pre-existing remediations in an electronic data storage medium; and” (Claim 11[c][5])**

621. Markel and Takagi in view of Hendry disclose this element for the reasons described above with respect to Ground 5, Claim element 1[c][5].

Accordingly, Markel and Takagi in view of Hendry thus disclose or render this element obvious.

- i. **“assigning, by the computer system, a descriptive attribute to the element in the code of the web page based on the one or more pre-existing remediations, the descriptive attribute assigned to the element adapted to enable an assistive technology to speak the descriptive attribute to a user.” (Claim 11[d])**

622. Markel and Takagi in view of Hendry disclose this element for the reasons described above with respect to Ground 5, Claim element 1[d].

Accordingly, Markel and Takagi in view of Hendry thus disclose or render this element obvious.

12. Claim 12

623. Claim 12 depends on claim 11 and recites the additional requirement of: **“wherein the one or more pre-existing remediations are javascript.”**

Markel and Takagi in view of Hendry disclose this claim for the reasons described above with respect to Ground 5, Claim 2. Accordingly, Markel and Takagi in view of Hendry thus disclose or render this claim obvious.

13. Claim 13

624. Claim 13 depends on claim 11 and recites the additional requirement of: **“wherein the element is an input field.”** Markel and Takagi in view of Hendry disclose this claim for the reasons described above with respect to Ground 5, Claim 3. Accordingly, Markel and Takagi in view of Hendry thus disclose or render this claim obvious.

14. Claim 14

625. Claim 14 depends on claim 11 and recites the additional requirement of: **“wherein the element is an image.”** Markel and Takagi in view of Hendry disclose this claim for the reasons described above with respect to Ground 5, Claim 4. Accordingly, Markel and Takagi in view of Hendry thus disclose or render this claim obvious.

15. Claim 16

626. Claim 16 depends on claim 11 and recites the additional requirement of: **“wherein the element lacks the adequate descriptive attribute when one or more descriptive attributes of the element are erroneous.”** Markel and Takagi in view of Hendry disclose this claim for the reasons described above with respect to Ground 5, Claim 6. Accordingly, Markel and Takagi in view of Hendry thus disclose or render this claim obvious.

16. Claim 17

627. Claim 17 depends on claim 11 and recites the additional requirement of: **“wherein the assigning comprises changing the DOM or HTML code associated with the web page.”** Markel and Takagi in view of Hendry disclose this claim for the reasons described above with respect to Ground 5, Claim 7. Accordingly, Markel and Takagi in view of Hendry thus disclose or render this claim obvious.

17. Claim 18

628. Claim 18 depends on claim 11 and recites the additional requirement of: **“wherein the descriptive attribute is an HTML label.”** Markel and Takagi in view of Hendry disclose this claim for the reasons described above with respect to Ground 5, Claim 8. Accordingly, Markel and Takagi in view of Hendry thus disclose or render this claim obvious.

18. Claim 19

629. Claim 19 depends on claim 11 and recites the additional requirement of: **“wherein the descriptive attribute is an alt-text label.”** Markel and Takagi in view of Hendry disclose this claim for the reasons described above with respect to Ground 5, Claim 9. Accordingly, Markel and Takagi in view of Hendry thus disclose or render this claim obvious.

19. Claim 20

630. Claim 20 depends on claim 11 and recites the additional requirement of: **“wherein the computer system comprises one or more computing systems.”** Markel and Takagi in view of Hendry disclose this claim for the reasons described above with respect to Ground 5, Claim 10. Accordingly, Markel and Takagi in view of Hendry thus disclose or render this claim obvious.

G. GROUND 6: Claims 1-4, 6-14, 16-20 are Unpatentable as Obvious Over Markel and Bigham in view of Hendry

1. Motivation to Combine

631. A POSITA would have been motivated to combine Markel, Bigham and Hendry in order to improve upon Markel’s modules for diagnosing, evaluating, reporting, and retrofitting code violations existing in websites, e.g., such as missing HTML tags, alt tag for images, a title tag, and labels for form fields. Markel provides several modules for automatically, semi-automatically, or manually remediating web pages, and can be further benefit from, and take advantage of, the labeling service and/or mechanism for users to request an untagged element such as an image to be labeled by humans and the powerful functions of using a cached database that allows new labels to be seamlessly added after a web page is downloaded, that were commonplace by the time of the ’532 patent.

632. As I described above, Markel describes several modules for correcting code violations on a website including missing HTML tags, alt tag for images, a

Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

title tag, and labels for form fields. Ex. 1006, at Abstract, ¶¶72, 83, 88. Markel discloses an assessment module, a reporting module, and a correcting module, together used for evaluating and correcting websites. *Id.*, at Abstract. Specifically, Markel discloses an assessment module for parsing and analyzing a website based on a function of a ruleset of design guideline. Ex. 1006, at Abstract. Markel's ruleset of design guidelines comprise a set of standards, such as the Section 508 standard and a usability standard such as the W3C Web Content Accessibility Guideline. *Id.*, at Abstract, ¶73. Markel's assessment module identifies standards violations that can include missing HTML tags, alt tag for images, a title tag, and labels for form fields (e.g., input fields). *Id.*, at Abstract, ¶¶72, 83, 88. Markel's reporting module is used to list these items, e.g., the design errors and standards violations, that can be automatically corrected, that can use semi-automatic assisted correction, and that are for manual correction. *Id.*, at Abstract. Markel also discloses a correcting module for automatically, semi-automatically, or manually correcting the listed items. *Id.*

633. Markel allows for "semi-automated wizard-assisted manual implementation, and support for manual correction of reported ruleset violation." Ex. 1006, at ¶78. For example, Markel's system "provides additional functionality including fixing minor alignment problems or font specifications, or adding ALT

Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

tags to images after querying the user for appropriate labels.” *Id.* Thus, Markel’s system can be used for automatic, semi-automatic, or manual remediation of ruleset violations. *Id.*, at Abstract ¶¶72, 78, 83, 88.

634. Furthermore, Markel also discloses that access to Markel’s system “is provided over the internet and the graphical user interface 112 is embodied in a web page accessed via a web browser 108.” Ex. 1006, at ¶104. A POSITA would have understood that access over the internet to disclose that Markel’s system resides on a remote server. Furthermore, referring to Figure 1 of Markel, the computer of Markel’s system is clearly accessed by a user through a network 106, e.g., such as the internet, and is separate from the user’s computer 105. To this end, a POSITA would have been motivated to improve Markel’s system for evaluating and correcting websites, which can access over the internet. To the extent Markel does not explicitly disclose retrieving manual remediations from a server, it would have been obvious. Takagi, Bigham, and Hendry all disclose remediations originating from a remote server. Ex. 1006, at 196, col. 1, ¶¶3, 6; Ex. 1008, at 184, col. 2, ¶2; Ex. 1009, at ¶52.

635. Furthermore, well before the ’532 patent, it was well known that it would be desirable to request for manual remediations by humans taught by Bigham. Ex. 1008, at 186, col. 2, ¶¶3-5. A POSITA would have been motivated

Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

to apply requested remediation, by humans other than the user, of untagged elements such as images since not all images can be applied for automatic labeling.

Ex. 1008, at 186, col. 2, ¶3. A POSITA would have expected the combination to work without undue experimentation because its already operating over the web and methods for pulling data from a wide range of sources over a network, such as databases, was well known. Ex. 1006, at 196, col. 1, ¶¶3, 6; Ex. 1008, at 184, col. 2, ¶3; Ex. 1009, ¶¶14, 15, 52. *See also* Ex. 1005, at Tables I(a)-(b), II(a).

Mechanisms for users to request that an untagged element such as an image to be labeled, e.g., by humans via a labeling service, were well known by the time of the '532 patent. Ex. 1008, at 186, col. 2, ¶3.

636. A POSITA would have understood that Bigham teaches how to generate labels for untagged elements such as images since Bigham's WebInSight system provides a mechanism for users to request for labeling by humans. *Id.* Specifically, Bigham discloses a "[t]o generate labels for these images, the WebInSight system provides a mechanism for users to request that an image be sent to a labeling service for labeling by humans." *Id.* Bigham's WebInSight system allows "users to request alternative text and receive a reply with an acceptable latency." *Id.*, at 186, col. 2 ¶4.

Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

637. Furthermore, a POSITA would have been motivated to use Bigham's WebInSight system combined with Markel's system since the WebInSight system "implements a simple web-based application for humans to label images." *Id.*, at 186, col. 2, ¶5. Also, Bigham discloses the WebInSight "system caches alternative text in a local database and can add new labels seamlessly after a web page is downloaded, resulting in minimal impact to the browsing experience." *Id.*, at Abstract. A POSITA would therefore have been motivated to combine Bigham's WebInSight system with Markel's system, in order to improve the accuracy of creating appropriate label text for untagged elements such as images, and to allow new labels to be seamlessly added to webpages after being download as Bigham discloses. *Id.*, at 186, col. 2, ¶¶3-5, Abstract.

638. Similar to combining Markel with Bigham, a POSITA would have been motivated to combine Hendry's server machine functionalities in fixing the Web accessibility issues with the systems of Markel and Bigham. In general, and as discussed above, Hendry teaches systems and methods for modifying documents (e.g., web pages; *see* Ex. 1009, at ¶42) to improve document accessibility implemented on a client machine or server machine. Ex. 1009, at ¶59. For example, Hendry teaches that its methods can be performed on: (1) a local client computer, (2) a remote server, or (3) a combination of the two. *Id.* Hendry also

Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

teaches that its “[e]xamples [that] refer to a specific machine performing the steps [are merely] for the purposes of clarity and understanding” and that any step can be performed on any machine. *Id.* Hendry teaches that the various steps recited can be repeated, omitted, or performed in different order. *Id.*

639. Hendry discloses that its “document modifier (120) is executed on the server machine (110) in response to receiving a request for a document from the client machine (102)” and that the “document modifier (120) on the server machine (110) determines that the particular requested document (114) is not accessibility enabled during an analysis of the particular requested document (114). The document (114) is analyzed to identify tags that are not recognizable by assistive technologies (106). The server machine (110) executes the document modifier (120) to produce a modified document (116) which is an accessibility-enabled version of the particular requested document (114). In response to receiving a request from the client machine (102) for the particular document (114), the server machine (110) sends the modified document (116).” *Id.*, at ¶53.

640. Hendry further discloses that “[t]he server machine (110) includes a data repository (112) storing one or more of: documents (114), modified documents (116) and accessibility features (118). The data repository (112) corresponds to any data storage device (for example, local memory on the server

Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

machine (110), web servers connected over the internet, machines within a local area network, a memory on a mobile device, local memory on the client machine (102), etc.).” *Id.*, at ¶41. Further, Hendry discloses “[t]hose skilled in the art will appreciate that elements or various portions of data stored in the data repository (112) may be distributed (for example, stored on client machine (102) or at other servers)” where “the data repository (112) includes [a] flat, hierarchical, network based, relational, dimensional, object modeled, or data files structured otherwise” and “is maintained as a table of a SQL database.” *Id.*

641. A POSITA would also have expected this combination to succeed with minimal modification. Indeed, Markel already includes an several modules that analyze a website as a function of a ruleset of design, list items for automatic, semi-automatic and manual correction, and automatically, semi-automatically and manually correct the listed items. Ex. 1006, at Abstract. Bigham teaches that automatic labeling can be improved using humans to generate labels from some images and caching the labels on a local database for seamless access, while Hendry teaches modifying a HTML document on the server machine. Ex. 1008, at 186, col. 2, ¶¶3-5; Ex. 1009, at ¶15.

642. A POSITA could simply modify Markel’s code to also query a database for matching descriptive attributes that had been authored by others. A

Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

POSITA could also add an interface (e.g., the WebInSight system's web-based application) that allows humans to respond to users' requests and quickly make alternative text labels via a cached local database with minimal work and the rest of Markel system, such as its assessment, reporting and correction module, among other modules, that would work as intended (with improved results from using the WebInsight system labeling service). Ex. 1008, at 186, col. 2, ¶¶3-5, Abstract; Ex. 1006, at Abstract, ¶¶71-76.

643. Bigham also discloses the benefit of using simple web-based application for humans to label untagged elements such as images, e.g., providing a mechanism for users to request an image to be sent to a labeling service by humans, since automatic labeling (such as that performed by the Markel system) does not apply to all web images. Ex. 1008, at 186, Section 5.3 ¶¶1-3. Thus, a POSITA would have understood the benefit of combining the Bigham with the Markel system for evaluating and correcting websites based on a ruleset. Ex. 1015 at ¶¶2, 4, 15, 18; Ex. 1006, at ¶72.

644. Furthermore, a POSITA would have understood that collaborative web accessibility improvements leveraged by the participation of an open community were well known by the time of the '532 patent. Ex. 1007, at 196, col. 1, ¶3.

Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

645. For example, as I explained in Section IV above, various types of social approaches in addition to Bigham, such as the 1997 ALT-server, Teraguchi, Asakawa, Takagi, and WebVisum, could provide an improvement over the existing automated accessibility enhancements described by Markel. Ex. 1007, at 196, col. 1, ¶6; Ex. 1031, at 2-4; Ex. 1032, at 177, col. 2, ¶¶3-4, Figure 4; Ex. 1033; Ex. 1034; Ex. 1035, at 1:18-36, Figure 1.

646. Additionally, as I explained in Section IV, it was a commonly known approach to store alternative text in a remote server, where the alternative text was generated with the help manual remediation. Ex. 1016, at 4, Ex. 1008, at 186, col. 2, ¶¶1-3, Figure 4; Ex. 1032, at 177, col. 2, ¶¶3-4; Ex. 1007, at 196, col. 1, ¶6.

647. Therefore, it would have been obvious to a POSITA to combine Markel, which discloses a several modules for evaluating and correcting websites based on a design ruleset, with Bigham and Hendry, both of which disclose inputting alternative text from a remote server machine.

2. Independent Claim 1

- a. “A computer-implemented method of programmatically assigning a descriptive attribute to an untagged element on a web page to enable an audible description of the untagged element, the web page having an associated document object model (DOM), the method comprising:” (Claim 1[pre])**

648. To the extent the preamble is limiting, Markel and Bigham in view of Hendry disclose this element. Markel discloses this claim for the reasons described above for Ground 4, Claim 1[pre]. Bigham discloses this claim for the reasons described above for Ground 3, Claim 1[pre]. And Hendry discloses the preamble for the reasons described above for Ground 2, Claim 1[pre].

- b. **“dynamically analyzing, by a computer system, code associated with the web page, the code comprising at least the DOM or HTML code;” (Claim 1[a])**

649. Markel and Bigham in view of Hendry disclose this element. Markel discloses this element for the reasons described above for Ground 4, Claim 1[a]. Bigham discloses this claim for the reasons described above for Ground 3, Claim 1[a]. And Hendry discloses this element for the reasons described above for Ground 2, Claim 1[a].

- c. **“detecting, by the computer system, one or more compliance issues relating to web accessibility standards in the code, wherein at least one of the one or more compliance issues comprises the untagged element lacking an adequate descriptive attribute;” (Claim 1[b])**

650. Markel and Bigham in view of Hendry disclose this element. Markel discloses this element for the reasons described above for Ground 4, Claim 1[b]. Bigham discloses this claim for the reasons described above for Ground 3, Claim

Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

1[b]. And Hendry also discloses this element for the reasons described above for Ground 2, Claim 1[b].

- d. **“applying, by the computer system, one or more pre-existing remediation code to the one or more compliance issues, wherein the one or more pre-existing remediations is generated by a remote server system performing at least:” (Claim 1[c][1])**

651. Markel and Bigham in view of Hendry disclose this element. Markel discloses this element for the reasons described above for Ground 4, Claim 1[c][1]. Bigham discloses this claim for the reasons described above for Ground 3, Claim 1[c][1]. And Hendry discloses this element for the reasons described above for Ground 2, Claim 1[c][1].

- e. **“accessing the code associated with the web page” (Claim 1[c][2])**

652. Markel and Bigham in view of Hendry disclose this element. Markel discloses this element for the reasons described above for Ground 4, Claim 1[c][2]. Bigham discloses this claim for the reasons described above for Ground 3, Claim 1[c][2]. And Hendry discloses this element for the reasons described above for Ground 2, Claim 1[c][2].

- f. **“receiving, by the remote server system, via a remediation interface input from a user, the input comprising a remediation action to manually remediate a non-programmatically-fixable**

**compliance issue associated with the web page;”
(Claim 1[c][3])**

653. Markel and Bigham in view of Hendry disclose this element. Markel discloses this element for the reasons described above for Ground 4, Claim 1[c][3]. Bigham discloses this claim for the reasons described above for Ground 3, Claim 1[c][3]. And Hendry discloses this element for the reasons described above for Ground 2, Claim 1[c][3].

- g. “generating, by the remote server system, the one or more pre-existing remediations based on the input comprising the remediation action to manually remediate a non-programmatically-fixable compliance issue; ” (Claim 1[c][4])**

654. Markel and Bigham in view of Hendry disclose this element. Markel discloses this element for the reasons described above for Ground 4, Claim 1[c][4]. Bigham discloses this claim for the reasons described above for Ground 3, Claim 1[c][4]. And Hendry discloses this element for the reasons described above for Ground 2, Claim 1[c][4].

- h. “storing, by the remote server system, the one or more pre-existing remediations in an electronic data storage medium; and” (Claim 1[c][5])**

655. Markel and Bigham in view of Hendry disclose this element. Markel discloses this element for the reasons described above for Ground 4, Claim 1[c][5]. Bigham discloses this claim for the reasons described above for Ground 3, Claim

Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

1[c][5]. And Hendry discloses this element for the reasons described above for Ground 2, Claim 1[c][5].

656. Bigham further discloses using a database (a remote server, an electronic storage medium) to cache alternative text (one or more preexisting remediations). Ex. 1008, at 181, Abstract. Bigham’s WebInSight system “caches alternative text in a local database and can add new labels seamlessly after a web page is downloaded, resulting in minimal impact to the browsing experience.” *Id.*

657. Therefore, Markel and Bigham in view of Hendry discloses and renders obvious this element.

- i. **“assigning, by the computer system, a descriptive attribute to the untagged element in the code of the web page based on the one or more pre-existing remediations, the descriptive attribute assigned to the untagged element adapted to enable an assistive technology to speak the descriptive attribute to a user.” (Claim 1[d])**

658. Markel and Bigham and Hendry disclose this element. Markel discloses this element for the reasons described above for Ground 4, Claim 1[d]. Bigham discloses this claim for the reasons described above for Ground 3, Claim 1[d]. And Hendry discloses this element for the reasons described above for Ground 2, Claim 1[d].

3. Claim 2

659. Claim 2 depends on claim 1 and recites “**wherein the one or more pre-existing remediations are javascript.**” Markel and Bigham in view of Hendry disclose this claim. Markel discloses this element for the reasons described above for Ground 4, Claim 2. Bigham discloses this claim for the reasons described above for Ground 3, Claim 2. And Hendry discloses this claim for the reasons described above for Ground 2, Claim 2.

4. Claim 3

660. Claim 3 depends on claim 1 and recites the additional requirement of: “**wherein the untagged element is an input field.**” Markel and Bigham in view of disclose this claim. Markel discloses this claim for the reasons described above for Ground 4, Claim 3. Bigham discloses this claim for the reasons described above for Ground 3, Claim 3. And Hendry discloses this claim for the reasons described above for Ground 2, Claim 3.

661. Furthermore, it would have been obvious to apply the teachings of Bigham to Markel and Hendry. Bigham discloses “a system that automatically creates and inserts alternative text into web pages on-the-fly.” Ex. 1008, at 181, Abstract. Bigham teaches generating alternative text for untagged elements and further discloses a mechanism for users to request an untagged element such as an image to be sent for labeling by humans. Ex. 1006, at 186, col. 2 ¶3. For example,

Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

Bigham's WebInSight system allows "users to request alternative text and receive a reply with an acceptable latency." *Id.*, at 186, col. 2 ¶4. Both Markel and Hendry both already teach manual human labeling (Ex. 1006, at ¶¶91, 113; Ex. 1009, at ¶85), corresponding to the same role of Bigham's "simple web-based application for humans to label images." Ex. 1008, at 186, col. 2, ¶4. Markel and Hendry also both teach storing remediation code in an electronic storage medium such as a database (Ex. 1006, at ¶¶101, 114; Ex. 1009, at ¶41), corresponding to the same role of Bigham's database. For example, Bigham's Figure 4 below shows a database:

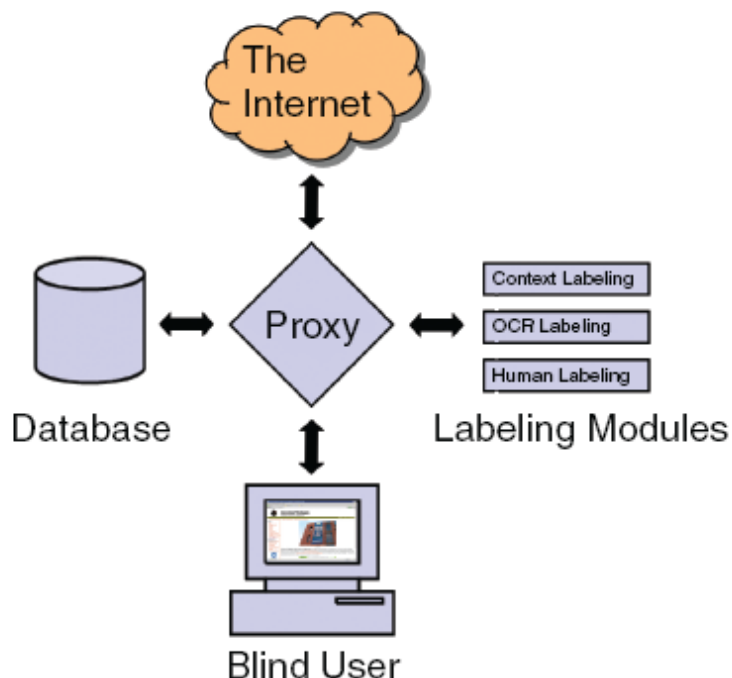


Figure 4: The architecture of WebInSight.

Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

Ex. 1008, at Figure 4. “When a user loads a web page, the system retrieves alternative text from its database for each image when it is present and requests that alternative text be calculated for any image not already in its database.” Ex. 1008, at 184, col. 2, ¶2.

662. Because both Markel and Hendry already teach Bigham Figure 4’s key components of human labeling and an associated database, a POSITA would have been motivated to apply the same approach as Bigham to other types of untagged elements in addition to images, e.g., such an input field, as taught by Markel and Hendry.

663. Therefore, Markel and Bigham in view of Hendry thus disclose and render this claim obvious.

5. Claim 4

664. Claim 4 depends on claim 1 and recites the additional requirement of: **“wherein the untagged element is an image.”** Markel and Bigham in view of Hendry disclose this claim. Markel discloses this claim for the reasons described above for Ground 4, Claim 4. Bigham discloses this claim for the reasons described above for Ground 3, Claim 4. And Hendry discloses this claim for the reasons described above for Ground 2, Claim 4.

665. Similarly, Bigham’s WebInSight system automatically creates and inserts alternative text for images. Specifically Bigham discloses “automatically creates and inserts alternative text into web pages on-the-fly. To formulate alternative text for images, we present three labeling modules based on web con-text analysis, enhanced optical character recognition (OCR) and human labeling.” Ex. 1008, at 181, Abstract. Thus Bigham also discloses this claim.

666. Markel and Bigham in view of Hendry thus disclose and render this claim obvious.

6. Claim 6

667. Claim 6 depends on claim 1 and recites the additional requirement of: **“wherein the untagged element lacks the adequate descriptive attribute when one or more descriptive attributes of the untagged element are erroneous.”** Markel and Bigham in view of Hendry disclose this claim. Markel discloses this claim for the reasons described above for Ground 4, Claim 6. Bigham discloses this claim for the reasons described above for Ground 3, Claim 6.

7. Claim 7

668. Claim 7 depends on claim 1 and recites the additional requirement of: **“wherein the assigning comprises changing the DOM or HTML code associated with the web page.”** Markel and Bigham in view of Hendry disclose

this claim. Markel discloses this claim for the reasons described above for Ground 4, Claim 7. Bigham discloses this claim for the reasons described above for Ground 3, Claim 7.

8. Claim 8

669. Claim 8 depends on claim 1 and recites the additional requirement of: **“wherein the descriptive attribute is an HTML label.”** Markel and Hendry in view of Bigham disclose this claim. Markel discloses this claim for the reasons described above for Ground 4, Claim 8. Bigham discloses this claim for the reasons described above for Ground 3, Claim 8. And Hendry discloses this claim for the reasons described above for Ground 2, Claim 8.

670. Furthermore, to the extent this limitation is not explicitly disclosed by Bigham, for the reasons set forth above with regard to Ground 6, claim 3, it would have been obvious to apply the same approach as Bigham to other types of labels for untagged elements in addition to images, e.g., such as for label of an input field, as taught by Markel and Hendry.

9. Claim 9

671. Claim 9 depends on claim 1 and recites the additional requirement of: **“wherein the descriptive attribute is an alt-text label.”** Markel and Bigham in view of Hendry disclose this claim. Markel discloses this claim for the reasons

described above for Ground 4, Claim 9. Bigham discloses this claim for the reasons described above for Ground 3, Claim 9. And Hendry discloses this claim for the reasons described above for Ground 2, Claim 9.

10. Claim 10

672. Claim 10 depends on claim 1 and recites the additional requirement of: **“wherein the computer system comprises one or more computing systems.”** Markel and Bigham in view of Hendry disclose this claim. Markel discloses this claim for the reasons described above for Ground 4, Claim 10. Hendry discloses this claim for the reasons described above for Ground 2, Claim 10.

673. Therefore, Markel and Bigham in view of Hendry thus disclose and render this claim obvious.

11. Independent Claim 11

- a. **“A non-transitory electronic storage medium with computer code stored thereon, the computer code configured to programmatically assign a descriptive attribute to an element on a web page to enable an audible description of the element, the web page having an associated document object model (DOM), and the computer code configured to perform, when executed by a computer processor, the steps of:”
(Claim 11[pre])**

674. To the extent the preamble is limiting, Markel and Bigham in view of Hendry disclose this element. Markel and Bigham in view of Hendry discloses this element for the reasons described above for Ground 6, Claim 1[pre]. A

Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

POSITA would have understood that a computer implemented method (as claimed in claim 1) was routinely performed through computer code stored on a non-transitory electronic medium. Additionally, the recital of a “non-transitory computer readable information storage media” was well-known in the art and would have been obvious to a POSITA. Furthermore, a POSITA would have understood that a descriptive attribute to an element on a web page to include an untagged element as described in claim 1[pre].

675. Markel and Bigham in view of Hendry thus disclose and render obvious the preamble.

- b. “dynamically analyzing, by a computer system, code associated with the web page, the code comprising at least the DOM or HTML code” (Claim 11[a])**

676. Markel and Bigham in view of Hendry discloses this element for the reasons described above with respect to Ground 6, Claim element 1[a].

Accordingly, Markel and Bigham in view of Hendry thus disclose or render this element obvious.

- c. “detecting, by the computer system, one or more compliance issues relating to web accessibility standards in the code, wherein at least one of the one or more compliance issues comprises the element lacking an adequate descriptive attribute;” (Claim 11[b])**

677. Markel and Bigham in view of Hendry discloses this element for the reasons described above with respect to Ground 6, Claim element 1[b].

Accordingly, Markel and Bigham in view of Hendry thus disclose or render this element obvious.

- d. **“applying, by the computer system, one or more pre-existing remediations to the one or more compliance issues, wherein the one or more pre-existing remediations is generated by a remote server system performing at least:” (Claim 11[c][1])**

678. Markel and Bigham in view of Hendry discloses this element for the reasons described above with respect to Ground 6, Claim element 1[c][1].

Accordingly, Markel and Bigham in view of Hendry thus disclose or render this element obvious.

- e. **“accessing the code associated with the web page;” (Claim 11[c][2])**

679. Markel and Bigham in view of Hendry discloses this element for the reasons described above with respect to Ground 6, Claim element 1[c][2].

Accordingly, Markel and Bigham in view of Hendry thus disclose or render this element obvious.

- f. **“receiving, by the remote server system, via a remediation interface input from a user, the input comprising a remediation action to manually remediate a non-programmatically-fixable**

**compliance issue associated with the web page;”
(Claim 11[c][3])**

680. Markel and Bigham in view of Hendry discloses this element for the reasons described above with respect to Ground 6, Claim element 1[c][3].

Accordingly, Markel and Bigham in view of Hendry thus disclose or render this element obvious.

- g. “generating, by the remote server system, the one or more pre-existing remediations based on the input comprising the remediation action to manually remediate a non-programmatically-fixable compliance issue;” (Claim 11[c][4])**

681. Markel and Bigham in view of Hendry discloses this element for the reasons described above with respect to Ground 6, Claim element 1[c][4].

Accordingly, Markel and Bigham in view of Hendry thus disclose or render this element obvious.

- h. “storing, by the remote server system, the one or more pre-existing remediations in an electronic data storage medium; and” (Claim 11[c][5])**

682. Markel and Bigham in view of Hendry discloses this element for the reasons described above with respect to Ground 6, Claim element 1[c][5].

Accordingly, Markel and Bigham in view of Hendry thus disclose or render this element obvious.

- i. “assigning, by the computer system, a descriptive attribute to the element in the code of the web page**

**based on the one or more pre-existing remediations,
the descriptive attribute assigned to the element
adapted to enable an assistive technology to speak the
descriptive attribute to a user.” (Claim 11[d])**

683. Markel and Bigham in view of Hendry discloses this element for the reasons described above with respect to Ground 6, Claim element 1[d].

Accordingly, Markel and Bigham in view of Hendry thus disclose or render this element obvious.

12. Claim 12

684. Claim 12 depends on claim 11 and recites the additional requirement of: **“wherein the one or more pre-existing remediations are javascript.”**

Markel and Bigham in view of Hendry discloses this claim for the reasons described above with respect to Ground 6, Claim 2. Accordingly, Markel and Bigham in view of Hendry thus disclose or render this claim obvious.

13. Claim 13

685. Claim 13 depends on claim 11 and recites the additional requirement of: **“wherein the element is an input field.”** Markel and Bigham in view of Hendry discloses this claim for the reasons described above with respect to Ground 6, Claim 3. Accordingly, Markel and Bigham in view of Hendry thus disclose or render this claim obvious.

14. Claim 14

Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

686. Claim 14 depends on claim 11 and recites the additional requirement of: **“wherein the element is an image.”** Markel and Bigham in view of Hendry discloses this claim for the reasons described above with respect to Ground 6, Claim 4. Accordingly, Markel and Bigham in view of Hendry thus disclose or render this claim obvious.

15. Claim 16

687. Claim 16 depends on claim 11 and recites the additional requirement of: **“wherein the element lacks the adequate descriptive attribute when one or more descriptive attributes of the element are erroneous.”** Markel and Bigham in view of Hendry discloses this claim for the reasons described above with respect to Ground 6, Claim 6. Accordingly, Markel and Bigham in view of Hendry thus disclose or render this claim obvious.

16. Claim 17

688. Claim 17 depends on claim 11 and recites the additional requirement of: **“wherein the assigning comprises changing the DOM or HTML code associated with the web page.”** Markel and Bigham in view of Hendry discloses this claim for the reasons described above with respect to Ground 6, Claim 7. Accordingly, Markel and Bigham in view of Hendry thus disclose or render this claim obvious.

17. Claim 18

689. Claim 18 depends on claim 11 and recites the additional requirement of: **“wherein the descriptive attribute is an HTML label.”** Markel and Bigham in view of Hendry discloses this claim for the reasons described above with respect to Ground 6, Claim 8. Accordingly, Markel and Bigham in view of Hendry thus disclose or render this claim obvious.

18. Claim 19

690. Claim 19 depends on claim 11 and recites the additional requirement of: **“wherein the descriptive attribute is an alt-text label.”** Markel and Bigham in view of Hendry discloses this claim for the reasons described above with respect to Ground 6, Claim 9. Accordingly, Markel and Bigham in view of Hendry thus disclose or render this claim obvious.

19. Claim 20

691. Claim 20 depends on claim 11 and recites the additional requirement of: **“wherein the computer system comprises one or more computing systems.”** Markel and Bigham in view of Hendry discloses this claim for the reasons described above with respect to Ground 6, Claim 20.

692. Accordingly, Markel and Bigham in view of Hendry thus disclose or render this claim obvious.

H. No Secondary Considerations Support the Patentability of Claims 1-5, 7-13, 16-20, and 22-28 of the '532 patent

693. I understand that certain “secondary considerations” or indicia of non-obviousness can be raised to rebut an argument of obviousness. I understand that at this time, the Patent Owner has not identified any such secondary considerations, and thus I expressly reserve the right to respond to any secondary considerations that may be asserted by Patent Owner in this proceeding.

VIII. CONCLUSION

694. In my opinion, based on my review of the '532 patent, the materials referenced herein, and my knowledge of what a person of ordinary skill in the art would have known at and before the '532 patent's priority date about the technology at issue, a person of ordinary skill in the art would have understood all of the limitations of the challenged claims to be rendered obvious by the prior art cited herein.

695. I reserve the right to supplement my opinions in the future to respond to any arguments or positions that the Patent Owner may raise, taking account of new information as it becomes available to me.

696. I hereby declare that all statements made herein of my own knowledge are true and that all statements made on information and belief are believed to be true; and further that these statements were made with the

Declaration of James Fogarty, Ph.D. in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 11,061,532

knowledge that willful false statements and the like so made are punishable by fine
or imprisonment, or both, under Section 1001 of Title 18 of the United States
Code.

Respectfully submitted,

Dated: September 7, 2022


James Fogarty, Ph.D.