IN THE UNITED STATES DISTRICT COURT
FOR THE WESTERN DISTRICT OF TEXAS
WACO DIVISION

CLOUDOFCHANGE, LLC, Plaintiff, v. CLOVER NETWORK, INC., Defendant.	Civil Action No. 6:22-cv-00634-ADA

DEFENDANT CLOVER NETWORK, INC.'S
PRELIMINARY INVALIDITY CONTENTIONS

TABLE OF CONTENTS

Page

I. INTRODUCTION 1
II. RESERVATIONS 2
III. GENERAL COMMENTS APPLICABLE TO ALL ASSERTED PATENTS 3
IV. IDENTIFICATION OF PRIOR ART 4
A. The '640 Patent 5
1. Prior Art Patents / Patent Applications 5
2. Non-Patent Literature 6
3. Additional Prior Art References 6
4. Product Prior Art Under 35 U.S.C. §§ 102(a) and (b) 10
B. The '012 Patent 12
1. Prior Art Patents / Patent Applications 12
2. Non-Patent Literature 13
3. Additional Prior Art References 14
4. Product Prior Art Under 35 U.S.C. §§ 102(a) and (b) 18
C. The '793 Patent 19
1. Prior Art Patents / Patent Applications 19
2. Non-Patent Literature 21
3. Additional Prior Art References 21
4. Product Prior Art Under 35 U.S.C. §§ 102(a) and (b) 25
V. CLAIM CHARTS 27
VI. DISCLOSURE OF INVALIDITY DUE TO ANTICIPATION AND/OR OBVIOUSNESS 27
A. The Claimed Concepts Were Known 28
1. Software as a Subscription / Software as a Service (SaaS) ("SaaS References") 30
2. Point of Sale Builder Tool POS Builder ("POS Builder References") 43
3. Point of Sale Graphical User Interface ("POS GUI References") 81
4. Remote Web Server ("Remote Web Server References") 101
5. Remote Manager that Can Access the Remote Web Server From Anywhere ("Remote Manager References") 130
6. Browser Access to Point of Sale Screens ("Browser References") 148
7. Operating Point of Sale Terminals Independent of Remote Server ("Independent Terminals References") 159

B. Obviousness Combinations....167
1. ’640 Patent....167
a. Woycik Alone or in Combination With One or More SaaS References....168
b. Manno alone or in combination with one or more POS Builder References....172
c. Tengler alone or in combination with one or more POS Builder References and one or more SaaS References....176
d. Cusack alone or in combination with one or more POS Builder References and one or more SaaS References....183
e. Olson alone or in combination with one or more POS Builder References....191
f. Lukkari alone or in combination with one or more POS Builder References and one or more SaaS References....195
g. Flashpoint alone or in combination with one or more Remote Web Server References and one or more POS Builder References and one or more SaaS References and one or more Browser References....202
h. Halo alone or in combination with one or more POS Builder References....213
i. InfoGenesis alone or in combination with one or more POS Builder References and one or more Browser References....217
j. Simphony alone or in combination with one or more POS Builder References....223
k. Squirrel alone or in combination with one or more Remote Manager References and one or more SaaS References and one or more POS Builder References and one or more Browser References....227
2. ’012 Patent....238
a. Manno alone or in combination with one or more POS GUI References....240
b. Tengler alone or in combination with one or more POS GUI References and one or more Browser References....245
c. Cusack alone or in combination with one or more POS GUI References....251
d. Cusack alone or in combination with one or more POS GUI References and one or more Independent Terminals References 256
e. Olson alone or in combination with one or more POS GUI References....264

f. Lukkari alone or in combination with one or more POS GUI References and one or more Independent Terminals References 269

g. Flashpoint alone or in combination with one or more Remote Web Server References and one or more Browser References 276

h. Halo alone or in combination with one or more Independent Terminals References.....281

i. NextPOS alone or in combination with one or more Remote Web Server References and one or more Browser References and one or more Independent Terminals References.....285

j. Point of Success alone or in combination with one or more Remote Web Server References and one or more Browser References and one or more Independent Terminals References 294

k. InfoGenesis alone or in combination with one or more POS GUI References.....303

l. Squirrel alone or in combination with one or more POS GUI References and one or more Remote Web Server References and one or more Browser References and one or more Independent Terminals References.....307

3. '793 Patent 320

a. Tengler alone or in combination with one or more Browser References.....321

b. Olson alone or in combination with one or more Browser References.....323

c. Flashpoint alone or in combination with one or more Remote Web Server References and one or more Browser References.....326

d. Halo alone or in combination with one or more Remote Web Server References 333

e. Phoenix alone or in combination with one or more POS Builder References and one or more Remote Web Server References 337

f. InfoGenesis alone or in combination with one or more Remote Web Server References and one or more Browser References ...344

g. Simphony alone or in combination with one or more POS Builder References and one or more Remote Web Server References and one or more Browser References.....350

h. Squirrel alone or in combination with one or more Remote Web Server References and one or more Browser References.....360

VII. SECONDARY CONSIDERATIONS 366

VIII. ON-SALE BAR 368

IX. CONTENTIONS UNDER 35 U.S.C. § 112 368

A. Legal Standard 369

1. Written Description and Enablement 369
2. Indefiniteness 371
B. The Asserted Claims 372
1. The Asserted Claims of the ’640 Patent Lack Written Description and/or Enablement under 35 U.S.C. § 112, ¶ 1 372
a. Claim 1 372
b. Claim 3 373
c. Claim 5 374
d. Claim 6 376
2. The Asserted Claims of the ’012 Patent Lack Written Description and/or Enablement under 35 U.S.C. § 112, ¶ 1 376
a. Claim 1 376
3. The Asserted Claims of the ’793 Patent Lack Written Description and/or Enablement under 35 U.S.C. § 112, ¶ 1 377
a. Claim 1 377
b. Claim 4 378
4. The Asserted Claims of the ’640 Patent are Indefinite under 35 U.S.C. § 112, ¶ 2 378
a. Claim 1 379
b. Claim 3 380
c. Claim 5 381
d. Claim 6 382
e. Claim 8 382
f. Claim 9 383
g. Claim 10 383
h. Claim 11 383
i. Claim 12 383
5. The Asserted Claims of the ’012 Patent are Indefinite under 35 U.S.C. § 112, ¶ 2 384
a. Claim 1 384
b. Claim 2 385
c. Claim 4 386
6. The Asserted Claims of the ’793 Patent are Indefinite under 35 U.S.C. § 112, ¶ 2 386
a. Claims 1, 42, 44 386
b. Claim 3 387

X. CONTENTIONS UNDER 35 U.S.C. § 101 388
A. The Asserted Claims Are Abstract Ideas Under *Alice* Step One 390
B. The Asserted Claims, Alone or In Combination, Do Not Include An Inventive Concept Sufficient to Transform the Claimed Abstract Idea Into a Patent-Eligible Application Under *Alice* Step Two 392

I. INTRODUCTION

Pursuant to the Court's October 24, 2022 Scheduling Order (D.I. 17), Defendant Clover Network, LLC[1] ("Clover" or "Defendant") hereby serves its Preliminary Invalidity Contentions ("Preliminary Invalidity Contentions") with respect to the asserted claims of U.S. Patent No. 9,400,640 ("the '640 patent"), U.S. Patent No. 10,083,012 ("the '012 patent"), and U.S. Patent No. 11,226,793 ("the '793 patent") (the "Asserted Patents"), identified by CloudofChange, LLC ("CloudofChange") in CloudofChange's Preliminary Infringement Contentions, served on September 30, 2022 ("Preliminary Infringement Contentions").

In its Preliminary Infringement Contentions, CloudofChange has asserted the following claims (the "Asserted Claims") against Clover:

1. **'640 patent:** Claims 1, 3, 5-6, and 8-14
2. **'012 patent:** Claims 1-2, 4, and 6-13
3. **'793 patent:** Claims 1-4, 9, 11, 42, and 44

With respect to each asserted claim and based on its investigation to date, Clover: (a) identifies each item of prior art that anticipates each asserted claim or renders it obvious; (b) explains why the prior art anticipates and/or renders obvious each asserted claim, and identifies any combinations of prior art showing obviousness; (c) submits charts specifically identifying where and how in each item of prior art each limitation of each asserted claim is found;[2] (d) identifies the grounds of invalidity based on indefiniteness or lacking written

[1] Clover Network, Inc. converted to Clover Network, LLC in 2021.
[2] These Preliminary Invalidity Contentions cite to the pre-AIA versions of 35 U.S.C. §§ 101, 102, 103, and 112.

description under 35 U.S.C. § 112; and (e) identifies the grounds of invalidity based on 35 U.S.C. § 101.[3]

II. RESERVATIONS

Clover reserves the right to revise its contentions depending upon the Court's constructions of the asserted claims. Clover reserves to revise its contentions depending upon CloudofChange's final infringement contentions, and/or any subsequent infringement contentions CloudofChange is allowed to serve in this case. Clover notes that discovery has not yet commenced in this case, and reserves the right to supplement its Preliminary Invalidity Contentions based on any information identified during discovery. Clover also expressly reserves the right to supplement its Preliminary Invalidity Contentions with third-party documents produced after the Preliminary Invalidity Contentions deadline. Clover further reserves the right to supplement its Preliminary Invalidity Contentions with third-party deposition testimony taken and third-party declarations provided after the Preliminary Invalidity Contentions deadline.

The aforementioned reservations and general comments apply to all charts referenced in Clover's Preliminary Invalidity Contentions, including the attached Exhibits.

[3] The Asserted Patents all share the same specification. A citation to one Asserted Patent should be understood as also including a citation to the analogous portion of any other Asserted Patent.

III. GENERAL COMMENTS APPLICABLE TO ALL ASSERTED PATENTS

The Supreme Court has held that "[t]he combination of familiar elements according to known methods is likely to be obvious when it does no more than yield predictable results." *KSR Int'l Co. v. Teleflex Inc.*, 550 U.S. 398, 416 (2007). "When a work is available in one field of endeavor, design incentives and other market forces can prompt variations of it, either in the same field or a different one." *Id.* at 417. As the Supreme Court made clear, "[f]or the same reason, if a technique has been used to improve one device, and a POSITA would have recognized that it would improve similar devices in the same way, using the technique is obvious unless its actual application is beyond his or her skill." *Id.*

To determine whether there is an apparent reason to combine the known elements in the fashion claimed by the patent at issue, a court can "look to interrelated teachings of multiple patents; the effects of demands known to the design community or present in the marketplace; and the background knowledge possessed by a person having ordinary skill in the art." *Id.* at 418. For example, obviousness can be demonstrated by showing "there existed at the time of invention a known problem for which there was an obvious solution encompassed by the patent's claims." *Id.* at 420. "[A]ny need or problem known in the field of endeavor at the time of invention and addressed by the patent can provide a reason for combining the elements in the manner claimed." *Id.* Common sense also teaches that "familiar items may have obvious uses beyond their primary purposes, and in many cases a person of ordinary skill will be able to fit the teachings of multiple patents together like pieces of a puzzle." *Id.*

Thus, the motivation to combine the teachings of the prior art references disclosed herein is found in the references themselves and/or: (1) the nature of the problem being solved; (2) the express, implied and inherent teachings of the prior art; (3) the knowledge of persons of ordinary skill in the art; (4) the fact that the prior art is generally directed towards the same or similar

problems; and/or (5) the predictable results obtained in combining the different elements of the prior art.

* * *

The foregoing general comments are relevant to how a person of ordinary skill in the art ("POSITA") at the time of the alleged invention of the Asserted Patents would have understood prior art references in the field of the Asserted Patents, and also relevant to which concepts a POSITA would have deemed obvious—both for single-reference obviousness and obviousness combinations.

IV. IDENTIFICATION OF PRIOR ART

In addition to the disclosures provided below, Clover attaches separate charts detailing its invalidity contentions on a claim-by-claim basis. (*See* Exs. A-'640-01 through A-'640-21, B-'012-01 through B-'012-21, and C-'793-01 through C-'793-21). Moreover, Clover hereby discloses and reserves the right to rely upon all references incorporated by reference into the references disclosed below and in the accompanying charts.[4] For the avoidance of doubt (where not separately identified), identification of a granted patent by patent number also identifies the corresponding publication numbers and provisional application numbers.

[4] To the extent a cited reference is found not to predate the Asserted Patents for which it is cited, it remains relevant, for example, to show the technological background and/or objective indicia of obviousness and/or obviousness based on simultaneous/contemporaneous invention by others.

A. The ’640 Patent

1. Prior Art Patents / Patent Applications

The following prior art patents and patent applications, including those references listed in Exs. A-’640-01 through A-’640-10, anticipate and/or render obvious the asserted claims of the ’640 patent.

No.	Patent No. / Application No.	Country of Origin	Date Issued / Published	Bates No.	Chart
1.	U.S. Patent App. Pub. No. 2004/0128199 (“Cusack”)	United States	July 1, 2004	CLOVER 000203	A-’640-01
2.	U.S. Patent App. Pub. No. 2004/0181454 (“Manno”)	United States	Sept. 16, 2004	CLOVER 000221	A-’640-02
3.	U.S. Patent App. Pub. No. 2005/0021409 (“Michaud”)	United States	Jan. 27, 2005	CLOVER 000237	A-’640-03
4.	U.S. Patent App. Pub. No. 2005/0049921 (“Tengler”)	United States	Mar. 3, 2005	CLOVER 000248	A-’640-04
5.	U.S. Patent App. Pub. No. 2005/0182680 (“Jones”)	United States	Aug. 18, 2005	CLOVER 000285	A-’640-05
6.	U.S. Patent App. Pub. No. 2007/0038727 (“Bailey”)	United States	Feb. 15, 2007	CLOVER 000321	A-’640-06
7.	U.S. Patent App. Pub. No. 2007/0175992 (“Brown”)	United States	Aug. 2, 2007	CLOVER 000351	A-’640-05
8.	U.S. Patent App. Pub. No. 2007/0265935 (“Woycik”)	United States	Nov. 15, 2007	CLOVER 000410	A-’640-07
9.	U.S. Patent App. Pub. No. 2007/0276763 (“Kleinman”)	United States	Nov. 29, 2007	CLOVER 000448	A-’640-05
10.	U.S. Patent App. Pub. No. 2008/0208696 (“Olson”)	United States	Aug. 28, 2008	CLOVER 000460	A-’640-08
11.	U.S. Patent No. 5,897,622 (“Blinn I”)	United States	Apr. 27, 1999	CLOVER 000468	A-’640-05
12.	U.S. Patent No. 6,058,373 (“Blinn II”)	United States	May 2, 2000	CLOVER 000501	A-’640-05
13.	U.S. Patent No. 6,920,431 (“Showghi”)	United States	July 19, 2005	CLOVER 000542	A-’640-05
14.	U.S. Patent No. 6,982,733 (“McNally”)	United States	Jan. 3, 2006	CLOVER 000563	A-’640-09

No.	Patent No. / Application No.	Country of Origin	Date Issued / Published	Bates No.	Chart
15.	International Application Publication WO 01/065427 ("Costello")	WIPO (Australia)	Sept. 7, 2001	CLOVER 000583	A-'640-05
16.	International Application Publication WO 02/039226 ("Thompson")	WIPO (United States)	May 16, 2002	CLOVER 000617	A-'640-05
17.	International Application Publication WO 03/065178 ("Redmond")	WIPO (United States)	Aug. 2, 2003	CLOVER 000668	A-'640-05
18.	JP 2004-164194A ("Nakamura")	Japan	June 10, 2004	CLOVER 000697; CLOVER 000727	A-'640-10

2. Non-Patent Literature

The following prior art publication(s), including those publications listed in Ex. A-'640-11 anticipate and/or render obvious the asserted claims of the '640 patent.

No.	Title	Date of Publication	Author or Publisher	Bates No.	Chart
1.	Lukkari *et al.*,*SmartRestaurant-Mobile Payments in Context-Aware Environment*, Proceedings of the 6th International Conference on Electronic Commerce, ICEC 2004, Delft, The Netherlands, October 25-27, 2004, http://dx.doi.org/10.1145/1052220.1052293 (Mar. 25, 2004) ("Lukkari")	Mar. 25, 2004	Lukkari	CLOVER 000752	A-'640-11

3. Additional Prior Art References

The following references may be relied upon to demonstrate the state of the art and/or the knowledge of a POSITA as it relates to the '640 patent in the relevant timeframe:

All documents cited on the face of the '640 patent and related patents, patent publications, file histories, and applications, as well as:

No.	Title	Date of Publication	Author or Publisher	Bates No.
1.	U.S. Patent Publication No. 2006/0235755 ("Mueller")	Oct. 19, 2006	n/a	CLOVER 004570
2.	United Kingdom Patent Application GB2369201 ("Tomlinson")	May 22, 2002	n/a	CLOVER 004771
3.	U.S. Pat. No. 5,818,428 ("Eisenbrandt")	Oct. 6, 1998	n/a	CLOVER 004791
4.	U.S. Patent Publication No. 2002/0194074 ("Jacobs")	Dec. 19, 2002	n/a	CLOVER 004319
5.	U.S. Patent No. 7,051,091 ("Cohen")	May 23, 2006	n/a	CLOVER 004202
6.	U.S. Patent No. 6,384,850 ("McNally")	May 7, 2002	n/a	CLOVER 000563
7.	U.S. Patent No. 6,002,395 ("Wagner")	Dec. 14, 1999	n/a	CLOVER 004808
8.	U.S. Patent Publication No. 2007/0175992 ("Brown")	Aug. 2, 2007	n/a	CLOVER 000351
9.	U.S. Patent No. 6,247,032 ("Bernardo")	June 12, 2001	n/a	CLOVER 003726
10.	U.S. Patent No. 8,364,845 ("Richter")	Jan. 29, 2013	n/a	CLOVER 003950
11.	Chieh-Chou Chou, *Morder-Server Food Service*, Cal. State U.	Mar. 2004	Cal. State U.	CLOVER 003273
12.	Sam Chung, et al., *Service-Oriented Software Reengineering: SoSR*, Proceedings of the 40th Annual Hawaii International Conference on System Sciences	2007	IEEE	CLOVER 003396
13.	Crystal Point Restaurant Technology Unleashed, https://web.archive.org/web/20071223084216/http://crystalpointpos.com:80/	Dec. 23, 2007	Crystal Point	CLOVER 003406
14.	DiamondTouch, https://web.archive.org/web/20070113195521/http://www.diamondtouchpos.com/	Jan. 13, 2007	DiamondTouch	CLOVER 003408
15.	Hospitality Technology, Pilot Issue	Jan. 1997	Hospitality Technology	CLOVER 002082
16.	John Jesitus, *Wireless Technology Keeps Customers In Order*, Hospitality Technology	Jan. 1997	Hospitality Technology	CLOVER 003409
17.	JP 2004-234394A	Aug. 19, 2004	n/a	CLOVER 003417
18.	JP 2004-310679	Nov. 4, 2004	n/a	CLOVER 003467

No.	Title	Date of Publication	Author or Publisher	Bates No.
19.	JP 2009-075745	April 9, 2009	n/a	CLOVER 003475
20.	T. Ojala, *SmartRotuaari – Context-aware Mobile Multimedia Services*	2003	ACM	CLOVER 003492
21.	KR 2003-0088637A	Nov. 20, 2003	n/a	CLOVER 003502
22.	LivePOS, https://web.archive.org/web/20081011090143/http://www.livepos.com/	Oct. 11, 2008	LivePOS	CLOVER 003536
23.	LivePOS Cloud Point of Sale Solutions - About LivePOS, https://web.archive.org/web/20120222113856/http://www.livepos.com:80/2012/links/aboutus.aspx		LivePOS	CLOVER 003533
24.	LivePOS Cloud Point of Sale Solutions - Features, https://web.archive.org/web/20120222231941/http://www.livepos.com:80/2012/Links/features.aspx		LivePOS	CLOVER 003534
25.	POS Software Showcase ’07	2007	Hospitality Technology	CLOVER 002061
26.	pcAmerica, https://web.archive.org/web/20070202145449/http://www.pcamerica.com:80/restaurant_pos_systems.html	Feb. 2, 2007	pcAmerica	CLOVER 002073
27.	pcAmerica, https://web.archive.org/web/20070202145412/http://www.pcamerica.com:80/restaurant_POS_screen_shots.htm	Feb. 2, 2007	pcAmerica	CLOVER 002075
28.	pcAmerica, https://web.archive.org/web/20070212004913/http://www.pcamerica.com:80/pos_products_features_restaurant_features.html	Feb. 12, 2007	pcAmerica	CLOVER 002073
29.	Norshahril Bin Abu Talib, Executive Restaurant Order System, University College of Engineering and Technology	Nov. 2005	Universiti Malaysia Pahang	CLOVER 003632
30.	U.S. Patent No. 5,696,909	Dec. 9, 1997	n/a	CLOVER 003656
31.	U.S. Patent No. 5,912,743	June 15, 1999	n/a	CLOVER 003668
32.	U.S. Patent No. 5,937,041	Aug. 10, 1999	n/a	CLOVER 003709

No.	Title	Date of Publication	Author or Publisher	Bates No.
33.	U.S. Patent No. 6,401,074	June 4, 2002	n/a	CLOVER 003785
34.	U.S. Patent No. 7,174,312	Feb. 6, 2007	n/a	CLOVER 003815
35.	U.S. Patent No. 7,472,394	Dec. 30, 2008	n/a	CLOVER 003854
36.	U.S. Patent No. 7,711,100	May 4, 2010	n/a	CLOVER 003869
37.	U.S. Patent No. 7,721,969	May 25, 2010	n/a	CLOVER 003880
38.	U.S. Patent No. 7,810,723	Oct. 12, 2010	n/a	CLOVER 003900
39.	U.S. Patent No. 7,841,514	Nov. 30, 2010	n/a	CLOVER 003914
40.	U.S. Patent No. 7,845,554	Dec. 7, 2010	n/a	CLOVER 003950
41.	U.S. Patent No. 8,032,414	Oct. 4, 2011	n/a	CLOVER 003914
42.	U.S. Patent No. 8,146,077	Mar. 27, 2012	n/a	CLOVER 004018
43.	U.S. Patent No. 8,595,061	Nov. 26, 2013	n/a	CLOVER 004040
44.	U.S. Patent No. 9,009,060	Apr. 14, 2015	n/a	CLOVER 004060
45.	U.S. Patent No. 9,131,012	Sept. 8, 2015	n/a	CLOVER 004085
46.	U.S. Patent No. 9,269,082	Feb. 23, 2016	n/a	CLOVER 004107
47.	U.S. Patent No. 9,679,320	June 13, 2017	n/a	CLOVER 004121
48.	U.S. Patent No. 9,747,651	Aug. 29, 2017	n/a	CLOVER 004159
49.	U.S. Patent No. 10,664,826	May 26, 2020	n/a	CLOVER 004182
50.	U.S. Patent App. Pub. No. 2001/0037206A1	Nov. 1, 2001	n/a	CLOVER 004216
51.	U.S. Patent App. Pub. No. 2001/0037245A1	Nov. 1, 2001	n/a	CLOVER 004232
52.	U.S. Patent App. Pub. No. 2001/0047302A1	Nov. 29, 2001	n/a	CLOVER 004248
53.	U.S. Patent App. Pub. No. 2002/0059405A1	May 16, 2002	n/a	CLOVER 004290
54.	U.S. Patent App. Pub. No. 2002/0095342A1	July 18, 2002	n/a	CLOVER 004305

No.	Title	Date of Publication	Author or Publisher	Bates No.
55.	U.S. Patent App. Pub. No. 2003/0078843A1	Apr. 24, 2003	n/a	CLOVER 004350
56.	U.S. Patent App. Pub. No. 2003/0120546A1	June 26, 2003	n/a	CLOVER 004383
57.	U.S. Patent App. Pub. No. 2003/0220841A1	Nov. 27, 2003	n/a	CLOVER 004401
58.	U.S. Patent App. Pub. No. 2004/0205023A1	Oct. 14, 2004	n/a	CLOVER 004449
59.	U.S. Patent App. Pub. No. 2006/0085265A1	Apr. 20, 2006	n/a	CLOVER 004487
60.	U.S. Patent App. Pub. No. 2006/0111983A1	May 25, 2006	n/a	CLOVER 004517
61.	U.S. Patent App. Pub. No. 2006/0235755A1	Oct. 19, 2006	n/a	CLOVER 004570
62.	U.S. Patent App. Pub. No. 2007/0005685A1	Jan. 4, 2007	n/a	CLOVER 004604
63.	U.S. Patent App. Pub. No. 2013/0110656A1	May 2, 2013	n/a	CLOVER 004631
64.	P.L. Venetianer, et al., *Video Verification of Point of Sale Transactions*, ObjectVideo, Inc.	2007	IEEE	CLOVER 004654
65.	Norbert Weibenberg, et al., *Using Ontologies in Personalized Mobile Applications*, GIS'04 (Nov. 12-13, 2004)	2004	ACM	CLOVER 004660
66.	Joyce A. Young, *The Web as an E-Commerce Medium*, Indiana State University, http://dx.doi.org/10.1300/J049v14n01_02	Dec. 2006	Journal of Marketing Channels	CLOVER 004719
67.	International App. Pub. WO2005/115074A2	Dec. 8, 2005	n/a	CLOVER 004670

4. Product Prior Art Under 35 U.S.C. §§ 102(a) and (b)

Clover also contends that the '640 patent is invalid in view of public knowledge and uses and/or offers for sale or sales of products and services that are prior art under 35 U.S.C. § 102(a) or (b) that anticipate and/or render obvious the asserted claims at least when applying CloudofChange's apparent interpretation of the claims in its Initial Infringement Contentions. Clover's investigation of these sources of prior art is ongoing. The following lists each item of prior art under 35 U.S.C. § 102(a), and/or (b), by the name of the item and, to the extent now

known, when the item became publicly known or was used, offered for sale, or sold. Clover contends that the below are stated on information and belief, and are supported by the information and documents that will be produced by Clover and/or third parties. Clover continues to investigate these events and reserves the right to add additional prior art products and services and/or to rely on additional documents or portions of documents describing the products and services below.

- The 21st Century Restaurant system, which was used, offered for sale, or sold before Feb. 5, 2008, Ex. A-'640-12.
- The Flashpoint POS system, which was used, offered for sale, or sold before Feb. 5, 2008, Ex. A-'640-13.
- The Halo system, which was used, offered for sale, or sold by Apr. 19, 2007, Ex. A-'640-14.
- NextPOS and/or Aldelo For Restaurants system, which was used, offered for sale, or sold by Feb. 28, 2003, Ex. A-'640-15.
- The Phoenix Point of Sale system, which was used, offered for sale, or sold by Mar. 13, 2006, Ex. A-'640-16.
- The Point of Success system, which was used, offered for sale, or sold by Dec. 22, 2005, Ex. A-'640-17.
- The Restaurant Pro Express system, which was used, offered for sale, or sold by Feb. 2, 2007, Ex. A-'640-18.
- The InfoGenesis system, which was used, offered for sale, or sold by Dec. 10, 2005, Ex. A-'640-19.

- The Simphony and/or Micros system, which was used, offered for sale, or sold by Jan. 21, 2002, Ex. A-'640-20.
- The Squirrel system, which was used, offered for sale, or sold by Dec. 1, 2005, Ex. A-'640-21.

B. The '012 Patent

1. Prior Art Patents / Patent Applications

The following prior art patents and patent applications, including those references listed in Exs. B-'012-01 through B-'012-10, anticipate and/or render obvious the asserted claims of the '012 patent.

No.	Patent No. / Application No.	Country of Origin	Date Issued / Published	Bates No.	Chart
1.	U.S. Patent App. Pub. No. 2004/0128199 ("Cusack")	United States	July 1, 2004	CLOVER 000203	B-'012-01
2.	U.S. Patent App. Pub. No. 2004/0181454 ("Manno")	United States	Sept. 16, 2004	CLOVER 000221	B-'012-02
3.	U.S. Patent App. Pub. No. 2005/0021409 ("Michaud")	United States	Jan. 27, 2005	CLOVER 000237	B-'012-03
4.	U.S. Patent App. Pub. No. 2005/0049921 ("Tengler")	United States	Mar. 3, 2005	CLOVER 000248	B-'012-04
5.	U.S. Patent App. Pub. No. 2005/0182680 ("Jones")	United States	Aug. 18, 2005	CLOVER 000285	B-'012-05
6.	U.S. Patent App. Pub. No. 2007/0038727 ("Bailey")	United States	Feb. 15, 2007	CLOVER 000321	B-'012-06
7.	U.S. Patent App. Pub. No. 2007/0175992 ("Brown")	United States	Aug. 2, 2007	CLOVER 000351	B-'012-05
8.	U.S. Patent App. Pub. No. 2007/0265935 ("Woycik")	United States	Nov. 15, 2007	CLOVER 000410	B-'012-07
9.	U.S. Patent App. Pub. No. 2007/0276763 ("Kleinman")	United States	Nov. 29, 2007	CLOVER 000448	B-'012-05
10.	U.S. Patent App. Pub. No. 2008/0208696 ("Olson")	United States	Aug. 28, 2008	CLOVER 000460	B-'012-08
11.	U.S. Patent No. 5,897,622 ("Blinn I")	United States	Apr. 27, 1999	CLOVER 000468	B-'012-05
12.	U.S. Patent No. 6,058,373 ("Blinn II")	United States	May 2, 2000	CLOVER 000501	B-'012-05
13.	U.S. Patent No. 6,920,431 ("Showghi")	United States	July 19, 2005	CLOVER 000542	B-'012-05

No.	Patent No. / Application No.	Country of Origin	Date Issued / Published	Bates No.	Chart
14.	U.S. Patent No. 6,982,733 ("McNally")	United States	Jan. 3, 2006	CLOVER 000563	B-'012-09
15.	International Application Publication WO 01/065427 ("Costello")	WIPO (Australia)	Sept. 7, 2001	CLOVER 000583	B-'012-05
16.	International Application Publication WO 02/039226 ("Thompson")	WIPO (United States)	May 16, 2002	CLOVER 000617	B-'012-05
17.	International Application Publication WO 03/065178 ("Redmond")	WIPO (United States)	Aug. 2, 2003	CLOVER 000668	B-'012-05
18.	JP 2004-164194A ("Nakamura")	Japan	June 10, 2004	CLOVER 000697; CLOVER 000727	B-'012-10

2. Non-Patent Literature

The following prior art publication(s), including those publications listed in Ex. B-'012-11, anticipate and/or render obvious the asserted claims of the '012 patent.

No.	Title	Date Published	Author or Publisher	Bates No.	Chart
1.	Lukkari *et al.*,*SmartRestaurant-Mobile Payments in Context-Aware Environment*, Proceedings of the 6th International Conference on Electronic Commerce, ICEC 2004, Delft, The Netherlands, October 25-27, 2004, http://dx.doi.org/10.1145/1052220.1052293 (Mar. 25, 2004) ("Lukkari")	Mar. 25, 2004	Lukkari	CLOVER 000752	B-'012-11

3. Additional Prior Art References

The following references may be relied upon to demonstrate the state of the art and/or the knowledge of a POSITA as it relates to the '012 patent in the relevant timeframe:

All documents cited on the face of the '012 patent and related patents, patent publications, file histories, and applications, as well as:

No.	Title	Date of Publication	Author or Publisher	Bates No.
1.	U.S. Patent Publication No. 2006/0235755 ("Mueller")	Oct. 19, 2006	n/a	CLOVER 004570
2.	United Kingdom Patent Application GB2369201 ("Tomlinson")	May 22, 2002	n/a	CLOVER 004771
3.	U.S. Pat. No. 5,818,428 ("Eisenbrandt")	Oct. 6, 1998	n/a	CLOVER 004791
4.	U.S. Patent Publication No. 2002/0194074 ("Jacobs")	Dec. 19, 2002	n/a	CLOVER 004319
5.	U.S. Patent No. 7,051,091 ("Cohen")	May 23, 2006	n/a	CLOVER 004202
6.	U.S. Patent No. 6,384,850 ("McNally")	May 7, 2002	n/a	CLOVER 000563
7.	U.S. Patent No. 6,002,395 ("Wagner")	Dec. 14, 1999	n/a	CLOVER 004808
8.	U.S. Patent Publication No. 2007/0175992 ("Brown")	Aug. 2, 2007	n/a	CLOVER 000351
9.	U.S. Patent No. 6,247,032 ("Bernardo")	June 12, 2001	n/a	CLOVER 003726
10.	U.S. Patent No. 8,364,845 ("Richter")	Jan. 29, 2013	n/a	CLOVER 003950
11.	Chieh-Chou Chou, *Morder-Server Food Service*, Cal. State U.	Mar. 2004	Cal. State U.	CLOVER 003273
12.	Sam Chung, et al., *Service-Oriented Software Reengineering: SoSR*, Proceedings of the 40th Annual Hawaii International Conference on System Sciences	2007	IEEE	CLOVER 003396
13.	Crystal Point Restaurant Technology Unleashed, https://web.archive.org/web/20071223084216/http://crystalpointpos.com:80/	Dec. 23, 2007	Crystal Point	CLOVER 003406
14.	DiamondTouch, https://web.archive.org/web/20070113195521/http://www.diamondtouchpos.com/	Jan. 13, 2007	DiamondTouch	CLOVER 003408

No.	Title	Date of Publication	Author or Publisher	Bates No.
15.	Hospitality Technology, Pilot Issue	Jan. 1997	Hospitality Technology	CLOVER 002082
16.	John Jesitus, *Wireless Technology Keeps Customers In Order*, Hospitality Technology	Jan. 1997	Hospitality Technology	CLOVER 003409
17.	JP 2004-234394A	Aug. 19, 2004	n/a	CLOVER 003417
18.	JP 2004-310679	Nov. 4, 2004	n/a	CLOVER 003467
19.	JP 2009-075745	April 9, 2009	n/a	CLOVER 003475
20.	T. Ojala, *SmartRotuaari – Context-aware Mobile Multimedia Services*	2003	ACM	CLOVER 003492
21.	KR 2003-0088637A	Nov. 20, 2003	n/a	CLOVER 003502
22.	LivePOS, https://web.archive.org/web/20081011090143/http://www.livepos.com/	Oct. 11, 2008	LivePOS	CLOVER 003536
23.	LivePOS Cloud Point of Sale Solutions - About LivePOS, https://web.archive.org/web/20120222113856/http://www.livepos.com:80/2012/links/aboutus.aspx		LivePOS	CLOVER 003533
24.	LivePOS Cloud Point of Sale Solutions - Features, https://web.archive.org/web/20120222231941/http://www.livepos.com:80/2012/Links/features.aspx		LivePOS	CLOVER 003534
25.	POS Software Showcase ’07	2007	Hospitality Technology	CLOVER 002061
26.	pcAmerica, https://web.archive.org/web/20070202145449/http://www.pcamerica.com:80/restaurant_pos_systems.html	Feb. 2, 2007	pcAmerica	CLOVER 002073
27.	pcAmerica, https://web.archive.org/web/20070202145412/http://www.pcamerica.com:80/restaurant_POS_screen_shots.htm	Feb. 2, 2007	pcAmerica	CLOVER 002075
28.	pcAmerica, https://web.archive.org/web/20070212004913/http://www.pcamerica.com:80/pos_products_features_restaurant_features.html	Feb. 12, 2007	pcAmerica	CLOVER 002073

No.	Title	Date of Publication	Author or Publisher	Bates No.
29.	Norshahril Bin Abu Talib, Executive Restaurant Order System, University College of Engineering and Technology	Nov. 2005	Universiti Malaysia Pahang	CLOVER 003632
30.	U.S. Patent No. 5,696,909	Dec. 9, 1997	n/a	CLOVER 003656
31.	U.S. Patent No. 5,912,743	June 15, 1999	n/a	CLOVER 003668
32.	U.S. Patent No. 5,937,041	Aug. 10, 1999	n/a	CLOVER 003709
33.	U.S. Patent No. 6,401,074	June 4, 2002	n/a	CLOVER 003785
34.	U.S. Patent No. 7,174,312	Feb. 6, 2007	n/a	CLOVER 003815
35.	U.S. Patent No. 7,472,394	Dec. 30, 2008	n/a	CLOVER 003854
36.	U.S. Patent No. 7,711,100	May 4, 2010	n/a	CLOVER 003869
37.	U.S. Patent No. 7,721,969	May 25, 2010	n/a	CLOVER 003880
38.	U.S. Patent No. 7,810,723	Oct. 12, 2010	n/a	CLOVER 003900
39.	U.S. Patent No. 7,841,514	Nov. 30, 2010	n/a	CLOVER 003914
40.	U.S. Patent No. 7,845,554	Dec. 7, 2010	n/a	CLOVER 003950
41.	U.S. Patent No. 8,032,414	Oct. 4, 2011	n/a	CLOVER 003914
42.	U.S. Patent No. 8,146,077	Mar. 27, 2012	n/a	CLOVER 004018
43.	U.S. Patent No. 8,595,061	Nov. 26, 2013	n/a	CLOVER 004040
44.	U.S. Patent No. 9,009,060	Apr. 14, 2015	n/a	CLOVER 004060
45.	U.S. Patent No. 9,131,012	Sept. 8, 2015	n/a	CLOVER 004085
46.	U.S. Patent No. 9,269,082	Feb. 23, 2016	n/a	CLOVER 004107
47.	U.S. Patent No. 9,679,320	June 13, 2017	n/a	CLOVER 004121
48.	U.S. Patent No. 9,747,651	Aug. 29, 2017	n/a	CLOVER 004159
49.	U.S. Patent No. 10,664,826	May 26, 2020	n/a	CLOVER 004182

No.	Title	Date of Publication	Author or Publisher	Bates No.
50.	U.S. Patent App. Pub. No. 2001/0037206A1	Nov. 1, 2001	n/a	CLOVER 004216
51.	U.S. Patent App. Pub. No. 2001/0037245A1	Nov. 1, 2001	n/a	CLOVER 004232
52.	U.S. Patent App. Pub. No. 2001/0047302A1	Nov. 29, 2001	n/a	CLOVER 004248
53.	U.S. Patent App. Pub. No. 2002/0059405A1	May 16, 2002	n/a	CLOVER 004290
54.	U.S. Patent App. Pub. No. 2002/0095342A1	July 18, 2002	n/a	CLOVER 004305
55.	U.S. Patent App. Pub. No. 2003/0078843A1	Apr. 24, 2003	n/a	CLOVER 004350
56.	U.S. Patent App. Pub. No. 2003/0120546A1	June 26, 2003	n/a	CLOVER 004383
57.	U.S. Patent App. Pub. No. 2003/0220841A1	Nov. 27, 2003	n/a	CLOVER 004401
58.	U.S. Patent App. Pub. No. 2004/0205023A1	Oct. 14, 2004	n/a	CLOVER 004449
59.	U.S. Patent App. Pub. No. 2006/0085265A1	Apr. 20, 2006	n/a	CLOVER 004487
60.	U.S. Patent App. Pub. No. 2006/0111983A1	May 25, 2006	n/a	CLOVER 004517
61.	U.S. Patent App. Pub. No. 2006/0235755A1	Oct. 19, 2006	n/a	CLOVER 004570
62.	U.S. Patent App. Pub. No. 2007/0005685A1	Jan. 4, 2007	n/a	CLOVER 004604
63.	U.S. Patent App. Pub. No. 2013/0110656A1	May 2, 2013	n/a	CLOVER 004631
64.	P.L. Venetianer, et al., *Video Verification of Point of Sale Transactions*, ObjectVideo, Inc.	2007	IEEE	CLOVER 004654
65.	Norbert Weibenberg, et al., *Using Ontologies in Personalized Mobile Applications*, GIS'04 (Nov. 12-13, 2004)	2004	ACM	CLOVER 004660
66.	Joyce A. Young, *The Web as an E-Commerce Medium*, Indiana State University, http://dx.doi.org/10.1300/J049v14n01_02	Dec. 2006	Journal of Marketing Channels	CLOVER 004719
67.	International App. Pub. WO2005/115074A2	Dec. 8, 2005	n/a	CLOVER 004670

4. Product Prior Art Under 35 U.S.C. §§ 102(a) and (b)

Clover also contends that the ’012 patent is invalid in view of public knowledge and uses and/or offers for sale or sales of products and services that are prior art under 35 U.S.C. § 102(a) or (b) that anticipate and/or render obvious the asserted claims at least when applying CloudofChange’s apparent interpretation of the claims in its Initial Infringement Contentions. Clover’s investigation of these sources of prior art is ongoing. The following lists each item of prior art under 35 U.S.C. § 102(a), and/or (b), by the name of the item and, to the extent now known, when the item became publicly known or was used, offered for sale, or sold. Clover contends that the below are stated on information and belief, and are supported by the information and documents that will be produced by Clover and/or third parties. Clover continues to investigate these events and reserves the right to add additional prior art products and services and/or to rely on additional documents or portions of documents describing the products and services below.

- The 21st Century Restaurant system, which was used, offered for sale, or sold before Feb. 5, 2008, B-’012-12.
- The Flashpoint POS system, which was used, offered for sale, or sold before Feb. 5, 2008, B-’012-13.
- The Halo system, which was used, offered for sale, or sold by Apr. 19, 2007, B-’012-14.
- NextPOS and/or Aldelo For Restaurants system, which was used, offered for sale, or sold by Feb. 28, 2003, B-’012-15.
- The Phoenix Point of Sale system, which was used, offered for sale, or sold by Mar. 13, 2006, B-’012-16.

- The Point of Success system, which was used, offered for sale, or sold by Dec. 22, 2005, B-'012-17.
- The Restaurant Pro Express system, which was used, offered for sale, or sold by Feb. 2, 2007, B-'012-18.
- The InfoGenesis system, which was used, offered for sale, or sold by Dec. 10, 2005, B-'012-19.
- The Simphony and/or Micros system, which was used, offered for sale, or sold by Jan. 21, 2002, B-'012-20.
- The Squirrel system, which was used, offered for sale, or sold by Dec. 1, 2005, B-'012-21.

C. The '793 Patent

1. Prior Art Patents / Patent Applications

The following prior art patents and patent applications, including those references listed in Exs. C-'793-01 through C-'793-10, anticipate and/or render obvious the asserted claims of the '793 patent.

No.	Patent No. / Application No.	Country of Origin	Date Issued / Published	Bates No.	Chart
1.	U.S. Patent App. Pub. No. 2004/0128199 ("Cusack")	United States	July 1, 2004	CLOVER 000203	C-'793-01
2.	U.S. Patent App. Pub. No. 2004/0181454 ("Manno")	United States	Sept. 16, 2004	CLOVER 000221	C-'793-02
3.	U.S. Patent App. Pub. No. 2005/0021409 ("Michaud")	United States	Jan. 27, 2005	CLOVER 000237	C-'793-03
4.	U.S. Patent App. Pub. No. 2005/0049921 ("Tengler")	United States	Mar. 3, 2005	CLOVER 000248	C-'793-04

No.	Patent No. / Application No.	Country of Origin	Date Issued / Published	Bates No.	Chart
5.	U.S. Patent App. Pub. No. 2005/0182680 (“Jones”)	United States	Aug. 18, 2005	CLOVER 000285	C-’793-05
6.	U.S. Patent App. Pub. No. 2007/0038727 (“Bailey”)	United States	Feb. 15, 2007	CLOVER 000321	C-’793-06
7.	U.S. Patent App. Pub. No. 2007/0175992 (“Brown”)	United States	Aug. 2, 2007	CLOVER 000351	C-’793-05
8.	U.S. Patent App. Pub. No. 2007/0265935 (“Woycik”)	United States	Nov. 15, 2007	CLOVER 000410	C-’793-07
9.	U.S. Patent App. Pub. No. 2007/0276763 (“Kleinman”)	United States	Nov. 29, 2007	CLOVER 000448	C-’793-05
10.	U.S. Patent App. Pub. No. 2008/0208696 (“Olson”)	United States	Aug. 28, 2008	CLOVER 000460	C-’793-08
11.	U.S. Patent No. 5,897,622 (“Blinn”)	United States	Apr. 27, 1999	CLOVER 000468	C-’793-05
12.	U.S. Patent No. 6,058,373 (“Blinn II”)	United States	May 2, 2000	CLOVER 000501	C-’793-05
13.	U.S. Patent No. 6,920,431 (“Showghi”)	United States	July 19, 2005	CLOVER 000542	C-’793-05
14.	U.S. Patent No. 6,982,733 (“McNally”)	United States	Jan. 3, 2006	CLOVER 000563	C-’793-09
15.	International Application Publication WO 01/065427 (“Costello”)	WIPO (Australia)	Sept. 7, 2001	CLOVER 000583	C-’793-05
16.	International Application Publication WO 02/039226 (“Thompson”)	WIPO (United States)	May 16, 2002	CLOVER 000617	C-’793-05
17.	International Application Publication WO 03/065178 (“Redmond”)	WIPO (United States)	Aug. 2, 2003	CLOVER 000668	C-’793-05
18.	JP 2004-164194A (“Nakamura”)	Japan	June 10, 2004	CLOVER 000697; CLOVER 000727	C-’793-10

2. Non-Patent Literature

The following prior art publication(s), including those publications listed in Ex. C-'793-11, anticipate and/or render obvious the asserted claims of the '793 patent.

No.	Title	Date of Publication	Author or Publisher	Bates No.	Chart
1.	Lukkari *et al.*,*SmartRestaurant-Mobile Payments in Context-Aware Environment*, Proceedings of the 6th International Conference on Electronic Commerce, ICEC 2004, Delft, The Netherlands, October 25-27, 2004, http://dx.doi.org/10.1145/1052220.1052293 (Mar. 25, 2004) ("Lukkari")	Mar. 25, 2004	Lukkari	CLOVER 000752	C-'793-11

3. Additional Prior Art References

The following references may be relied upon to demonstrate the state of the art and/or the knowledge of a POSITA as it relates to the '793 patent in the relevant timeframe:

All documents cited on the face of the '793 patent and related patents, patent publications, file histories, and applications, as well as:

No.	Title	Date of Publication	Author or Publisher	Bates No.
1.	U.S. Patent Publication No. 2006/0235755 ("Mueller")	Oct. 19, 2006	n/a	CLOVER 004570
2.	United Kingdom Patent Application GB2369201 ("Tomlinson")	May 22, 2002	n/a	CLOVER 004771
3.	U.S. Pat. No. 5,818,428 ("Eisenbrandt")	Oct. 6, 1998	n/a	CLOVER 004791
4.	U.S. Patent Publication No. 2002/0194074 ("Jacobs")	Dec. 19, 2002	n/a	CLOVER 004319
5.	U.S. Patent No. 7,051,091 ("Cohen")	May 23, 2006	n/a	CLOVER 004202

No.	Title	Date of Publication	Author or Publisher	Bates No.
6.	U.S. Patent No. 6,384,850 ("McNally")	May 7, 2002	n/a	CLOVER 000563
7.	U.S. Patent No. 6,002,395 ("Wagner")	Dec. 14, 1999	n/a	CLOVER 004808
8.	U.S. Patent Publication No. 2007/0175992 ("Brown")	Aug. 2, 2007	n/a	CLOVER 000351
9.	U.S. Patent No. 6,247,032 ("Bernardo")	June 12, 2001	n/a	CLOVER 003726
10.	U.S. Patent No. 8,364,845 ("Richter")	Jan. 29, 2013	n/a	CLOVER 003950
11.	Chieh-Chou Chou, *Morder-Server Food Service*, Cal. State U.	Mar. 2004	Cal. State U.	CLOVER 003273
12.	Sam Chung, et al., *Service-Oriented Software Reengineering: SoSR*, Proceedings of the 40th Annual Hawaii International Conference on System Sciences	2007	IEEE	CLOVER 003396
13.	Crystal Point Restaurant Technology Unleashed, https://web.archive.org/web/20071223084216/http://crystalpointpos.com:80/	Dec. 23, 2007	Crystal Point	CLOVER 003406
14.	DiamondTouch, https://web.archive.org/web/20070113195521/http://www.diamondtouchpos.com/	Jan. 13, 2007	DiamondTouch	CLOVER 003408
15.	Hospitality Technology, Pilot Issue	Jan. 1997	Hospitality Technology	CLOVER 002082
16.	John Jesitus, *Wireless Technology Keeps Customers In Order*, Hospitality Technology	Jan. 1997	Hospitality Technology	CLOVER 003409
17.	JP 2004-234394A	Aug. 19, 2004	n/a	CLOVER 003417
18.	JP 2004-310679	Nov. 4, 2004	n/a	CLOVER 003467
19.	JP 2009-075745	April 9, 2009	n/a	CLOVER 003475
20.	T. Ojala, *SmartRotuaari – Context-aware Mobile Multimedia Services*	2003	ACM	CLOVER 003492
21.	KR 2003-0088637A	Nov. 20, 2003	n/a	CLOVER 003502
22.	LivePOS, https://web.archive.org/web/20081011090143/http://www.livepos.com/	Oct. 11, 2008	LivePOS	CLOVER 003536

No.	Title	Date of Publication	Author or Publisher	Bates No.
23.	LivePOS Cloud Point of Sale Solutions - About LivePOS, https://web.archive.org/web/20120222113856/http://www.livepos.com:80/2012/links/aboutus.aspx		LivePOS	CLOVER 003533
24.	LivePOS Cloud Point of Sale Solutions - Features, https://web.archive.org/web/20120222231941/http://www.livepos.com:80/2012/Links/features.aspx		LivePOS	CLOVER 003534
25.	POS Software Showcase '07	2007	Hospitality Technology	CLOVER 002061
26.	pcAmerica, https://web.archive.org/web/20070202145449/http://www.pcamerica.com:80/restaurant_pos_systems.html	Feb. 2, 2007	pcAmerica	CLOVER 002073
27.	pcAmerica, https://web.archive.org/web/20070202145412/http://www.pcamerica.com:80/restaurant_POS_screen_shots.htm	Feb. 2, 2007	pcAmerica	CLOVER 002075
28.	pcAmerica, https://web.archive.org/web/20070212004913/http://www.pcamerica.com:80/pos_products_features_restaurant_features.html	Feb. 12, 2007	pcAmerica	CLOVER 002073
29.	Norshahril Bin Abu Talib, Executive Restaurant Order System, University College of Engineering and Technology	Nov. 2005	Universiti Malaysia Pahang	CLOVER 003632
30.	U.S. Patent No. 5,696,909	Dec. 9, 1997	n/a	CLOVER 003656
31.	U.S. Patent No. 5,912,743	June 15, 1999	n/a	CLOVER 003668
32.	U.S. Patent No. 5,937,041	Aug. 10, 1999	n/a	CLOVER 003709
33.	U.S. Patent No. 6,401,074	June 4, 2002	n/a	CLOVER 003785
34.	U.S. Patent No. 7,174,312	Feb. 6, 2007	n/a	CLOVER 003815
35.	U.S. Patent No. 7,472,394	Dec. 30, 2008	n/a	CLOVER 003854
36.	U.S. Patent No. 7,711,100	May 4, 2010	n/a	CLOVER 003869
37.	U.S. Patent No. 7,721,969	May 25, 2010	n/a	CLOVER 003880

No.	Title	Date of Publication	Author or Publisher	Bates No.
38.	U.S. Patent No. 7,810,723	Oct. 12, 2010	n/a	CLOVER 003900
39.	U.S. Patent No. 7,841,514	Nov. 30, 2010	n/a	CLOVER 003914
40.	U.S. Patent No. 7,845,554	Dec. 7, 2010	n/a	CLOVER 003950
41.	U.S. Patent No. 8,032,414	Oct. 4, 2011	n/a	CLOVER 003914
42.	U.S. Patent No. 8,146,077	Mar. 27, 2012	n/a	CLOVER 004018
43.	U.S. Patent No. 8,595,061	Nov. 26, 2013	n/a	CLOVER 004040
44.	U.S. Patent No. 9,009,060	Apr. 14, 2015	n/a	CLOVER 004060
45.	U.S. Patent No. 9,131,012	Sept. 8, 2015	n/a	CLOVER 004085
46.	U.S. Patent No. 9,269,082	Feb. 23, 2016	n/a	CLOVER 004107
47.	U.S. Patent No. 9,679,320	June 13, 2017	n/a	CLOVER 004121
48.	U.S. Patent No. 9,747,651	Aug. 29, 2017	n/a	CLOVER 004159
49.	U.S. Patent No. 10,664,826	May 26, 2020	n/a	CLOVER 004182
50.	U.S. Patent App. Pub. No. 2001/0037206A1	Nov. 1, 2001	n/a	CLOVER 004216
51.	U.S. Patent App. Pub. No. 2001/0037245A1	Nov. 1, 2001	n/a	CLOVER 004232
52.	U.S. Patent App. Pub. No. 2001/0047302A1	Nov. 29, 2001	n/a	CLOVER 004248
53.	U.S. Patent App. Pub. No. 2002/0059405A1	May 16, 2002	n/a	CLOVER 004290
54.	U.S. Patent App. Pub. No. 2002/0095342A1	July 18, 2002	n/a	CLOVER 004305
55.	U.S. Patent App. Pub. No. 2003/0078843A1	Apr. 24, 2003	n/a	CLOVER 004350
56.	U.S. Patent App. Pub. No. 2003/0120546A1	June 26, 2003	n/a	CLOVER 004383
57.	U.S. Patent App. Pub. No. 2003/0220841A1	Nov. 27, 2003	n/a	CLOVER 004401
58.	U.S. Patent App. Pub. No. 2004/0205023A1	Oct. 14, 2004	n/a	CLOVER 004449
59.	U.S. Patent App. Pub. No. 2006/0085265A1	Apr. 20, 2006	n/a	CLOVER 004487

No.	Title	Date of Publication	Author or Publisher	Bates No.
60.	U.S. Patent App. Pub. No. 2006/0111983A1	May 25, 2006	n/a	CLOVER 004517
61.	U.S. Patent App. Pub. No. 2006/0235755A1	Oct. 19, 2006	n/a	CLOVER 004570
62.	U.S. Patent App. Pub. No. 2007/0005685A1	Jan. 4, 2007	n/a	CLOVER 004604
63.	U.S. Patent App. Pub. No. 2013/0110656A1	May 2, 2013	n/a	CLOVER 004631
64.	P.L. Venetianer, et al., *Video Verification of Point of Sale Transactions*, ObjectVideo, Inc.	2007	IEEE	CLOVER 004654
65.	Norbert Weibenberg, et al., *Using Ontologies in Personalized Mobile Applications*, GIS'04 (Nov. 12-13, 2004)	2004	ACM	CLOVER 004660
66.	Joyce A. Young, *The Web as an E-Commerce Medium*, Indiana State University, http://dx.doi.org/10.1300/J049v14n01_02	Dec. 2006	Journal of Marketing Channels	CLOVER 004719
67.	International App. Pub. WO2005/115074A2	Dec. 8, 2005	n/a	CLOVER 004670

4. Product Prior Art Under 35 U.S.C. §§ 102(a) and (b)

Clover also contends that the '793 patent is invalid in view of public knowledge and uses and/or offers for sale or sales of products and services that are prior art under 35 U.S.C. § 102(a) or (b) that anticipate and/or render obvious the asserted claims at least when applying CloudofChange's apparent interpretation of the claims in its Initial Infringement Contentions. Clover's investigation of these sources of prior art is ongoing. The following lists each item of prior art under 35 U.S.C. § 102(a), and/or (b), by the name of the item and, to the extent now known, when the item became publicly known or was used, offered for sale, or sold. Clover contends that the below are stated on information and belief, and are supported by the information and documents that will be produced by Clover and/or third parties. Clover continues to investigate these events and reserves the right to add additional prior art products

and services and/or to rely on additional documents or portions of documents describing the products and services below.

- The 21st Century Restaurant system, which was used, offered for sale, or sold before Feb. 5, 2008, C-'793-12.
- The Flashpoint POS system, which was used, offered for sale, or sold before Feb. 5, 2008, C-'793-13.
- The Halo system, which was used, offered for sale, or sold by Apr. 19, 2007, C-'793-14.
- NextPOS and/or Aldelo For Restaurants system, which was used, offered for sale, or sold by Feb. 28, 2003, C-'793-15.
- The Phoenix Point of Sale system, which was used, offered for sale, or sold by Mar. 13, 2006, C-'793-16.
- The Point of Success system, which was used, offered for sale, or sold by Dec. 22, 2005, C-'793-17.
- The Restaurant Pro Express system, which was used, offered for sale, or sold by Feb. 2, 2007, C-'793-18.
- The InfoGenesis system, which was used, offered for sale, or sold by Dec. 10, 2005, C-'793-19.
- The Simphony and/or Micros system, which was used, offered for sale, or sold by Jan. 21, 2002, C-'793-20.
- The Squirrel system, which was used, offered for sale, or sold by Dec. 1, 2005, C-'793-21.

V. CLAIM CHARTS

Individual claim charts that identify where each element of each asserted claim can be found in each item of prior art are attached hereto. A listing of these claim charts is provided below:

Exhibit A-'640-1 through A-'640-21: Claim charts for the '640 patent

Exhibit B-'012-1 through B-'012-21: Claim charts for the '012 patent

Exhibit C-'793-1 through C-'793-21: Claim charts for the '793 patent

VI. DISCLOSURE OF INVALIDITY DUE TO ANTICIPATION AND/OR OBVIOUSNESS

Subject to the reservation of rights above and based on Clover's present understanding of the asserted claims of the Asserted Patents and the apparent constructions CloudofChange is asserting based on CloudofChange's Preliminary Infringement Contentions, the prior art references identified above in Section IV, and charted in Exs. A-'640-01 through A-'640-21, B-'012-01 through B-'012-21, and C-'793-01 through C-'793-21 attached hereto and incorporated by reference herein, each anticipate and/or render obvious the asserted claims. The charts identify where each element of each asserted claim can be found in each item of prior art, at least when applying CloudofChange's apparent constructions.

The anticipation and/or obviousness grounds listed herein are exemplary only and Clover reserves the right to amend or present additional anticipation and/or obviousness grounds based at least in part on the Court's construction of the asserted claims, any evidence or findings as to the priority or invention date of the asserted claims, any evidence that results from discovery, and/or positions that CloudofChange or its expert witness(es) may take concerning claim construction, infringement, and/or invalidity issues.

To the extent a finder of fact finds that a limitation of a given claim was not disclosed by one of the references identified above, those claims are nevertheless unpatentable as obvious because the asserted claims contain nothing that goes beyond ordinary innovation. To the extent not anticipated, no asserted claim goes beyond combining known elements to achieve predictable results or does more than choose between clear alternatives known to persons of ordinary skill in the art, particularly in view of the state of the art as reflected in the relevant references listed above in Section IV.

Moreover, to the extent the prior art is found not to anticipate the asserted claims, the prior art renders the asserted claims obvious either alone or in combination with one or more of the other prior art references herein. As explained herein and/or in the accompanying charts, it would have been obvious to a POSITA at the time of the alleged invention of the asserted claims of the Asserted Patents to combine the various references cited herein so as to practice the asserted claims of the Asserted Patents. In addition to the specific illustrative combinations of prior art and the specific illustrative combinations of groups of prior art disclosed, Clover reserves the right to rely on any other combination of any prior art references disclosed herein or in any amendment to Clover's invalidity contentions. Clover further reserves the right to rely upon combinations disclosed within the prosecution history of the references cited herein. These obviousness combinations reflect Clover's present understanding of the potential scope of the claims that CloudofChange appears to be advocating and should not be construed as Clover's acquiescence to CloudofChange's interpretation of the patent claims.

A. The Claimed Concepts Were Known

The Asserted Patents are purportedly directed to cover a combination of concepts, all of which were well known before the patents' alleged priority date. Below, Clover provides a description of general concepts that are relevant to one or more claims of the Asserted Patents.

Patent-specific concepts are discussed in more detail, *infra*, in Section VI.B. Clover reserves the right to rely on the references cited below and concepts described herein to show the state of the art and/or the knowledge of a skilled artisan.

1. Software as a Subscription / Software as a Service (SaaS) ("SaaS References")

The Asserted Patents purport to provide a solution to the long-recognized problem of providing software solutions, and particularly software for point-of-sale operations, to "non-expert business operators." '640 patent, 1:15-18. The Asserted Patents state that "Software as a Service (SMS) is a software distribution model in which applications are hosted by a vendor or service provider and made available to customers over a network, typically the Internet." '640 patent, 5:60-63.

The benefits of software as a subscription or as a service (SaaS) were known long before the Asserted Patents. Among other benefits, SaaS reduces hardware costs for business owners, such as restaurant owners. Moreover, SaaS is pre-installed and configured in the cloud by the SaaS vendor, so an instance can be provisioned quickly and available for use by a subscriber, with reduced time spent on installation and configuration. SaaS also provides cost savings, because it may reside in a shared or multi-tenant environment. Further, SaaS allows for rapidly scaling a customer base, because it allows small or medium businesses or individuals to use software that might not otherwise be accessible due to high licensing costs. In addition, SaaS reduces maintenance costs, because the SaaS provider maintains or provides for maintenance of hardware, and those costs are shared among all customers of the SaaS provider. SaaS is also often hosted in a cloud environment, which allows for rapid scaling, as well as straightforward integration with other SaaS offerings. This provides flexibility to customers to scale SaaS use up or down depending on specific needs, as well as enable or disable different features or software, depending on specific needs. SaaS providers, as part of maintaining the software solution, upgrade the software, which further reduces costs and complexity for customers, who are not required to purchase and install upgrade software packages, or upgrade hardware that may be

required to accompany software upgrades. SaaS may allow customers to access multiple different versions of software, which saves costs due to reduced requirements for re-training employees, and allows customers to continue to access files or projects created on earlier versions of software, while also having access to the latest software for new projects. SaaS is often straightforward to use, as SaaS offerings frequently provide samples, training, proof-of-concepts, test versions, test features, or the like.

SaaS is ubiquitous in the prior art, for example as described in the following ("SaaS References"):

- Manno at [0065] ("In an alternative arrangement, a WPOS system with a single server 12 can be used for a number of different merchants, each with his or her own point-of-sale computer. In that case, the server can be partitioned so that each subscribing merchant will have access only to his or her own accounting and inventory information. One possible example where this can be used is a small-scale or start-up retail business, where only a single POS or cash register would be needed. Then space on the server 12 can be rented temporarily or on a long term basis. Other partitioned space on the same server could be rented out to other similar businesses. Another example is in a craft fair or "flea market" environment, where many independent merchants would be present at a single retailing location. Also, WPOS can perform as a subscription service for companies to connect remotely to a staged WPOS server that is maintained by a WPOS management service. This would permit the small-scale business to subscribe to WPOS for a monthly fee, instead of having to purchase a WPOS server. A small business, a business under construction, or any business that feels

an in-store server is not necessary, might be interested in subscribing to a remote WPOS server, if the business's connection (i.e., Internet speed) supports it.").

- Michaud at Abstract, ("An application service provider point of sale system and method provide point of sale transaction processing and management for subscribers through remote access and use of a point of sale system hosted on ASP server. A thin client device known as a lane commander provides input and output for point of sale transaction processing. The lane commander also provides an off-line mode that enables transaction processing in the event on-line access to the ASP server becomes disabled. Subscriber demand for third party products and services is aggregated to secure volume discounts for the subscribers."), [0009] ("To solve one or more of the problems set forth above, in a first aspect of an exemplary implementation of the invention, an application service provider point of sale system is provided. The system includes an ASP server having point of sale software, at least one remote lane commander for at least one subscriber, and communications connectivity between said at least one lane commander and said ASP server."), [0012] ("In a fourth aspect of an exemplary implementation of the invention, an application service provider point of sale business method is provided. The method includes a step of providing an ASP server having point of sale software for access and use by the subscriber. An account is established for the subscriber on said point of sale software, thereby allowing subscriber access and use. A fee is charged to the subscriber for use of the point of sale software, said fee being based on point of sale transactions processed using the ASP server having point of sale software."), [0013] ("In a fifth aspect of an exemplary

implementation of the invention, the application service provider point of sale business method includes a step of securing discounts for third party products or services purchased by subscribers."), [0020] ("Referring to FIG. 1, a high-level schematic diagram of an application service provider (ASP) point of sale system in accordance with an exemplary implementation of the invention is shown. ASP server 105 hosts software for managing point of sale transactions and providing services. A plurality of subscribers 110, 115 and 120 access the ASP server 105 to perform point of sale transactions and obtain services. By way of example and not limitation, the services may include access to third party service providers 130. Access between the plurality of subscribers 110, 115 and 120 and the ASP server 105 is preferably via data communications networks, which may include the Internet. The ASP server 105 preferably maintains compilations of data (e.g., databases) 114, 119 and 123 for each of the plurality of subscribers 110, 115 and 120."), [0022] ("The ASP POS system provides many benefits. Advantageously, the ASP model provides a low cost of entry and short setup time for subscribers. In an exemplary implementation, subscribers pay a transaction fee for using the point of sale system, thus tying payment for use to a subscriber's sales."), [0023] ("The ASP model also eliminates the need for a specialized information technology infrastructure with costly and complex point of sale, network operating system and database management software. Furthermore, it reduces or eliminates the need for costly information technology staff."), [0024] ("Other advantages of the ASP model include access to a feature-rich preconfigured system and continued software and hardware updates at the ASP server."), [0025]

("In another aspect of the invention subscribers will have their own individual website hosted around a standard group-marketing program made possible because of the ASP format. Sites will be maintained for subscribers as part of the transaction fee. Subscribers will be able to use web sites for individual store promotions, coupons, general store information and community information as desired. Subscribers will also be able to participate in national and regional promotional programs from selected vendors. Additionally, subscribers will be able to participate in national and regional co-op advertising and have the funds credited back against their transaction fees."), [0026] ("In yet another aspect of the invention, an opt-in e-mail program, subscribers will be able to solicit their current customers for their e-mail address so that they can receive information on specials and events. Subscribers may maintain the program on a local basis and/or participate in a regional program and have the ASP host assist in producing and sending out periodic mails."), [0071] ("ACR5000™ by ACR Systems, Inc. is an exemplary point of sale software system that is configurable for operation in an ASP mode in accordance with the subject invention. Various features may be offered to subscribers as options."), [0072] ("Advantageously, the ASP model provides a vehicle for combining purchasing power for the overall benefit of each individual subscriber. Demand can be aggregated across multiple subscribers. The aggregated demand may help the subscribers meet minimum volumes required by certain manufacturers and/or qualify for volume discounts. Subscribers may purchase goods and services at substantially reduced prices that result from higher purchasing volumes."), [0073] ("Concomitantly, the ASP model facilitates

identifying goods and services required by subscribers. Such information can be determined from transaction data and the nature of the subscribers' businesses. Nearby delivery points can also be readily identified to lower shipping costs."), [0074] ("Typically, there are certain products and services that most, if not all, subscribers will require, irrespective of their type of business. Such services may (for example) include office supplies, telecommunications, payroll processing, computers, software, credit and debit card processing. An exemplary implementation of the invention aggregates subscriber demand to secure volume discounts from vendors for the benefit of the aggregated subscribers. Discounts will be negotiated with third party vendors based on the number of subscribers. Thus, the present invention advantageously provides a means for aggregating demand and strengthening bargaining power for the benefit of subscribers."), [0076] ("Once target products and services are identified, negotiations for volume discounts can commence with manufacturers, suppliers and providers. Volume discount opportunities can be communicated to subscribers, such as via the ASP server."), [0077] ("In addition to identifying volume discount opportunities, the system can improve logistical efficiency. It can readily identify subscribers having nearby delivery points (e.g., by zip code). This can facilitate delivery and reduce shipping costs. For example, a truckload of products may be delivered to several subscribers located within the same or adjacent towns."), [0078] ("The ASP server and lane commanders may also be configured to enable subscribers to access and use a wide array of software applications and services. By way of example, the ASP server may host financial management software (e.g.,

QuickBooks® by Intuit®), customer relationship management software, and any other applications and utilities that may be configured to run in the networked environment. Subscribers may access and use the software through lane commanders. Advantageously, this provides subscribers convenient access to installed, ready-to-use, up-to-date software without the expense of acquisition and the hassle of installation and updating.").

- Olson at [0027] ("The key advantages of this point of sale system with a web-based back-office system are as follows. The web and PC architecture both offer full-scalability which results in improved performance and lower cost. There is a lower cost associated with projects developed with the technology of this invention due to the flexibility of easy design changes and well-understood software. There is less training required for programmers and system testers. Projects can draw on the huge talent pool in the open source development community. The invention allows configurable modules for different types of businesses and sales promotions. The invention allows remote monitoring of both the POS and back-office systems from anywhere via the web. There is minimal time required for POS installation, since POS setup is as basic as a home PC setup. Another advantage is that the back-office system can be provided as a service or deployed within a corporation. For example, Software as a Service (SAAS) is a software distribution model in which applications are hosted by a vendor or service provider and made available to customers over a network, typically the Internet. Another advantage of this invention is that the back-office system is maintained in customer centric databases, making it impossible for

customers to see other's data. Each BO client (individual POS, store or multi-store owner) gets their own instance of a database. When they log into the BO they are attached to their own relational database associated and validated via their user login and password.").

- Nakamura (JP2004-164194A) at [0052] ("Moreover, the store server 50 requests the system server 30 to send email information to the customer (general user) 41, and accesses the system server 30 to update the POS application program. For this reason, the store 21 pays a predetermined fee, as a usage fee for the system server 30 and an information fee for receiving provision of general information, financial accounting data and sales analysis data, in addition to the above POS application program development cost, on a monthly or yearly basis to the management company 31 that manages the system server 30. Also, data communication between the system server 30 and the store server 50 is encrypted and authenticated using a protocol such as SSL (Netscape, USA) to prevent falsification or theft of data by a third party."), [0053] ("By acquiring and downloading a customer application program for receiving service provision from the system server 30, the mobile phone customer terminal 40 can receive various information providing services from the system server 30 and the mobile phone POS terminal 20. In order to receive the provision of the customer application program, it is necessary to register as a member of the system server 30 and pay a membership fee or the like on a monthly or yearly basis. A customer (member) 41 who has registered as a member can receive services from a plurality of member stores 21 contracted with the system server 30. For members, it is possible to save

the trouble of individually registering as members at multiple stores 21; the store 21 has the advantage of being able to reduce expenses such as customer management and member service implementation costs and advertising expenses.").

- Blinn (US 5,897,622) at 6:1-8 ("Customers pay monthly access fees to the online service providers for help services and access to the Internet through local telephone connections. Of course, the online service providers are optional, and in some cases, the clients 100 may have direct access to the Internet 104. For example, the client 100 may be connected to a local area network 108 which in turn is directly connected to the Internet 104.").
- Costello (WO 01/65427) at 8:5-7 (The server 12 runs application service provider software 40 capable of providing applications in the form of the POS program and browser interface to the plurality of POS devices 14, 15A to 15B."); 10:6-9 ("The ASP model provides flexibility in the functionality provided to the POS device client. The ASP infrastructure also allows access to the accounting backend by business advisers such as accountants.")
- Thompson at Claim 25 ("The system of claim 21 wherein establishing the functionality of the network appliance includes configuring the network appliance to function as means for delivering application service provider capabilities embodied in the primary server."); see also claims 26, 51 and 52.
- Redmond at [014] ("The POS Portal also includes a number of application services 14. These application services 14 are utilized by the POS Portal to integrate, coordinate, configure and present the third-party services 12 in a

seamless manner. For example, one application service includes a metering application that keeps track of the usage of each subscribed service by a subscriber thereby allowing the subscriber to be billed appropriately. Another application service includes a loyalty application that monitors the usage of each subscribed service in order to award earned credits or rewards to specific customers or merchants. In addition, these application services 14 may include specific customized services requested by a subscriber. For example, a subscriber who is a merchant may desire to offer the subscribed services for use by its customers or employees in a particular manner to provide branding or a specific 'look and feel' that characterizes the merchant's brand and offerings."); *see also* at Claim 6 ("The system of claim 1 wherein at least one or more of the computing devices is located at a subscriber location; and wherein the appliance server is either located at the subscriber location or hosted by a service provider"); claim 10, claim 12, claim 15, claim 16, claim 27 and FIG. 1.

- Halo
    - Halo, *Price Your Restaurant POS System*, https://web.archive.org/web/20080804002851/http://www.myhalo.com:80/restaurant/Price_your_POS/ (2008) ("Halo takes a different approach: we are committed to transparent pricing and quality, cheap point of sale software! Our low subscription fee includes upgrades, support, and maintenance. There are no hidden costs.").
    - The Green Sheet Online Edition, *Look to the light: Retail POS system can replace ECR*, www.greensheet.com/emagazine.php?article_id=136 (June

7, 2007) (“And since the data is housed off-site on the Halo secure server, merchants don't have to worry about losing information in a hard drive crash, theft or natural disaster, such as fire or flood. Halo relieves some other burdens as well. With an ECR, merchants must back up data and update systems with new versions. With Halo, upgrades are free and take place during the night.”).

- InfoGenesis
    - William Atkinson, *Point of Service: Now and Later of Restaurant POS* (Apr. 9, 2007) (“Infogenesis hosts Revelation as an application service provider (ASP) solution from a secure data center in California and maintains Internet connectivity with each of the three restaurants, processing all outlet and site transactions in a real-time environment. This approach allows the restaurant to receive the benefits of POS without having to hire a support staff or install server hardware. In addition, the system integrates with those the restaurants have from other vendors.”).
- NextPOS-Aldelo
    - NextPOS Corporation and Transactional Web, Inc. Partner to Replace Expensive and Labor Intensive Reporting and Accounting for Restaurants with Affordable Subscription-Priced Web Automation, https://www.prweb.com/releases/2004/11/prweb183924.htm#! (Nov. 2004) (“The integrated iEnterprise solution is provided as a subscription service Â a model frequently referred to as Âsoftware as a serviceÂ Â that gives businesses access to leading applications without large upfront

expenses or the burden of ongoing hardware and software maintenance. Software as service providers are able to offer solutions at a fraction of the cost of traditional systems primarily because they work from a single centralized version of the application delivered over the Internet. All enhancements, maintenance and upgrades happen for everyone at the same time. And, subscription pricing typically makes these applications cost-effective to a broader base of businesses than traditional software licensing.").

- Simphony
    - Micros Simphony TM, https://web.archive.org/web/20071112050945/http://www.micros.com:80/Products/Simphony/ (2007) ("Although Simphony may be implemented at a single property, the true value can be found in its ability to scale to a business with thousands of workstations spread over many properties. This solution can be hosted using one of MICROS's four worldwide datacenters, or it can be self-hosted by a customer. The Enterprise Management Console allows for the management of the entire system from within a single application. Users are able to create and define the parameters of the enterprise, properties, and revenue centers from any PC that has access to the central server, thus allowing local users to make changes to their configuration if necessary.")
    - Micros Simphony, https://web.archive.org/web/20060324063129/http://www.micros.com/Pro

ducts/Simphony/ (2006) (“Simphony is MICROS’s latest enterprise class hospitality POS product designed specifically for the leisure and entertainment market. Simphony is capable of being centrally-hosted using MICROS’s world-class hosting services or it can be self-hosted by a customer. Designed and built for scalability and flexible deployment, Simphony employs a modern services-oriented architecture. Simphony can handle a multi-property POS configuration consisting of thousands of workstations generating terabytes of sales data, and its web-based reporting solution minimizes the technology footprint on property.”).

- Hard Rock International Deploys Centralized Point of Sale with MICROS Simphony(TM), Oracle Hospitality, https://www.hospitalitynet.org/news/4032053.html (June 26, 2007) (“Hard Rock International has embraced the MICROS 8700 since 1996. New data security requirements and the age of the current solution required Hard Rock International to seek its next generation POS solution. Hard Rock strategically chose MICROS Simphony(TM) because of its centralized, service-oriented architecture (SOA), which reduces administrative and maintenance costs and allows for more effective management of the POS. Centralized database management ensures consistency and integrity for the Hard Rock brand while enabling IT to be more responsive to the needs of Operations. Hard Rock will also eliminate 60+ high-end computer servers, minimizing its overall support costs. Simphony(TM) service-

oriented architecture allows Hard Rock to deploy a tailored POS configuration to each location.

2. Point of Sale Builder Tool POS Builder ("POS Builder References")

The Asserted Patents purport to provide a solution to the long-recognized problem of "online, web-based point of sale (POS) building and configuration, which can assist non-expert business operators in building, editing and testing a point of sale system to manage their businesses." *See* '640 patent, Abstract. The Asserted Patents continue, "[c]urrent practice in the field of assembling point of sale systems includes manually coding front-of-screen information. Typically, this front-of-screen information contains menu selections, page selections, and general answers to business questions. This front-of-screen menu is typically manually coded by a business expert with the help of a programmer or data expert. Also, currently the entry of this front-of-screen information requires intimate knowledge of a complex interface to a front-of-screen programming language. In summary, current practice includes the manual building of a point of sale (POS) screen. This manual process requires defining the position and operation of touch screen keys and their database correspondence. Currently, only specially trained people can build or change POS screens. This manual POS building and editing is prone to mistakes and is time-consuming. Since POS screen changes are difficult and prone to error, store owners tend to retain older, inaccurate, out-of-date POS screens in order to avoid the POS screen editing process. Also, current POS screen editing occurs off-line with the testing of the screens occurring at a later date, at a remote store location." *See* '640 patent, 1:20-40. The Asserted Patents state, "[w]ith this invention, the store operator will be able to build his POS screens online over the Internet. With input from the store operator, the POS builder can specify and display the number, shape and arrangement of selection keys or buttons on said POS screens.

The store operator, who does not have to be technically trained, will be able to edit and test his screens until he is satisfied with the end results." *See* '640 patent, 2:61-3:1.

The benefits of (POS) builder tools were known long before the Asserted Patents. Among other benefits, POS builder tools allow fast and easy updates to POS menu screens, such as those in restaurants or retail establishments, with no or minimal special training. Providing a POS builder tool would allow a wide range of restaurant or retail employees (e.g., owners, managers, assistant managers, assistants to the managers, day workers, night workers, workers with technical backgrounds, workers without technical backgrounds) with tool(s) for setting up, configuring, modifying, or maintaining a POS interface. POS builder tools allow for companies to save money on training and IT services, and also to improve flexibility in updating menus or items for sale.

POS Builder Tools are ubiquitous in the prior art, for example as described in the following ("POS Builder References"):

- Woycik at [0013] ("The administration tool application includes a menu editor. The menu editor enables the administrator to create and edit the interactive menu screens provided by the self-order application. The menu editor accesses a library of menu templates containing predefined groupings of buttons, and uses the menu templates, as directed by the administrator, during creation and editing of the interactive menu screens."), [0095] ("The local server kiosk (16, FIG. 1; 82, FIG. 3) includes an administrative tool (98, FIG. 3) to run reports, check the system status, add or remove kiosks from the system, or edit the screen displays that the customers see. All of these settings are stored on the local server 16, 82. The administrative tool 98 allows an administrator to modify these settings via a local kiosk touch screen. The administrative

tool 98 is built using the same enhanced vector graphics approach (e.g., FLASH and .NET) described above for the self-order application interface so that administrators can setup menus and items, and change configuration items primarily via a kiosk touch screen. Therefore, the administrative tool 98 provides a graphical representation of menu options, button selections, images, text fields, etc. on the kiosk. Those items are readily editable through the graphical administrative tool 98 as will be discussed in more detail below. An advantage of this approach is that it enables administrators and store owners to make changes to the store offerings and customer interface without requiring any knowledge of the underlying code. In addition, administration is available from any client kiosk (30, FIG. 1; 80, FIG. 3) configured to communicate with the server 16, 82. This can be implemented by, for example, storing the administrative tool 98 on each client kiosk 30, 80 or by providing the system with the ability to execute the administrative tool 98 on the server 16, 82 using an application interface located on the client kiosks 30, 80. Where the administrative tool 98 is separately stored on each client kiosk 30, 80 and run locally on the client, the tool can be configured with the server IP address of the server kiosk 16, 82 so that, prior to giving the administrator access to change the kiosk, menu, and item setups, it verifies that there are no other instances of the administrative tool 98 running. Also, in this implementation, the configuration data for the kiosks, menus, and items can be stored on the server kiosk 16, 82 and downloaded to the client kiosk 30, 80 when the administrative tool 98 is run at the client. Changes to the configuration are then uploaded back to the server kiosk 16, 82 when the administrator is done. Thus, there is no need to go to the server 16, 82 in order to use the administrative tools 98 or to modify any of the

settings in the client interface. Moreover, remote administration using the administrative tool 98 can be carried out in any of the various ways discussed further above."), FIG. 10.

- McNally at 1:13-23 ("This invention relates to an information management and synchronous communications system and method for generation of computerized menus for restaurants and other applications with specialized display and synchronous communications requirements related to, for example, the use of equipment or sore with non-PC-standard graphical formats, display sizes and/or applications for use in remote data entry, information management and synchronous communication between host computer, digital input device or remote pager via standard hardwired connection, the internet, a wireless link, smart phone or the like."), 2:57-63 ("A principal object of the present invention is to provide an improved information management and synchronous communications system and method which facilitates user-friendly and efficient generation of computerized menus for restaurants and other applications that utilize equipment with non-PC-standard graphical formats, display sizes and/or applications."), 3:9-13 ("A further object of the present invention is to provide a synchronous information management and communications system which enables automatic updating of both wireless and internet menu systems when a new menu item is added, modified or deleted from any element of the system."), 3:17-38 ("The foregoing and other objects of the present invention are provided by a synchronous information management and communications system and method optimized for simplicity of operation which incorporates menu generation for creation of menus to be used with wireless remote handheld computer and PDA devices, the internet or any application where simple and efficient generation of menus is appropriate. The menu generation approach of the

present invention includes a desktop software application that enables the rapid creation and building of a menu and provides a means to instantly download the menu configuration onto, e.g., a handheld device or Web page and to seamlessly interface with standard point of sale ('POS') systems to enable automatic database updates and communication exchanges when a change or input occurs in any of the other system elements. To solve the above and other related problems, an information management and communications system is provided which results in a dramatic reduction in the amount of time, and hence cost, to generate and maintain computerized menus for, e.g., restaurants and other related applications that utilize non-PC-standard graphical formats, display sizes or applications."), 3:39-47 ("The menu generation approach of the present invention has many advantages over previous approaches in solving the problem of converting paper-based menus or Windows® PC-based menu screens to small PDA-sized displays and Web pages. In one embodiment, the present invention is a software tool for building a menu, optimizing the process of how the menu can be downloaded to either a handheld device or Web page, and making manual or automatic modifications to the menu after initial creation."), 4:59-65 ("The menu generation approach of the present invention is the first coherent solution available to accomplish these objectives easily and allows one development effort to produce both the handheld and Web page formats, link them with the existing POS systems, and thus provides a way to turn a complicated, time-consuming task into a simple process."), 5:40-47 ("FIG. 1 is a schematic representation of a window displayed on a computer display screen which shows a hierarchical tree menu, modifier window and sub-modifier window in conformity with a preferred embodiment of the present invention. FIG. 2 is a schematic representation of a modifier

dialog box in conformity with a preferred embodiment of the present invention."), 7:24-37 ("An aspect of the preferred embodiment of the information management and communications system of the invention is shown in FIG. 1. FIG. 1 shows an example of the GUI provided by the operating system of the preferred embodiment of the present invention. With reference to FIG. 1, the preferred embodiment includes an intuitive GUI I from which to build a menu on a desktop or other computer. A hierarchical tree structure 2 is used to show the different relationships between the menu categories 3 (e.g., soups, salads, appetizers, entrees, deserts, etc.), menu items 4 (e.g., green salad, chicken caesar salad, etc.), menu modifiers 5 (e.g., dressing, meat temperature, condiments, etc.) and menu sub-modifiers 6 (e.g., Italian, French, ranch, bleu cheese, etc.)."), 7:42-65 ("The menu configuration application is launched by clicking on the appropriate icon on the desktop display screen. FIG. 1 will then be displayed. There are three windows on the screen shown in FIG. 1. The left window is the menu tree 7, also called the tree view. The top right window is the Modifiers window 8 and the bottom right window is the Sub-Modifiers window 9. The Sub-Modifiers window lists the sub-modifiers that correspond to the modifier that is selected. The views on the right are referred to as list views. There are several ways of invoking a command, including using the menu options; using the context menu (right mouse click); using the keyboard or using the toolbar icons. For example, if it is desired to add a category to the menu, the following four options are available: (1) clicking on Edit, Add Category; (2) right mouse clicking on Menu, then clicking on Add Category; (3) highlighting Menu, then typing Ctrl+T or (4) clicking on the Add Category icon on the toolbar. To add an item to a category, the following options are available: (1) highlighting the category to which it is desired to add an item and then

clicking on Edit>Add Item; (2) right mouse clicking on the desired category and then clicking on Add Item; (3) highlighting the desired category, then typing Ctrl+N or (4) clicking on the Add icon on the toolbar."), 9:62-10:9 ("When the menu has been completely configured, it can be previewed on a POS emulator on the desktop to verify that the menu is correctly configured before downloading it to the handheld device. To preview, File>Preview Database is clicked on or the Preview Database icon from the toolbar is clicked on. The handheld POS emulator on the desktop can then be run. If the configuration is deemed acceptable, the handheld device is connected to the desktop PC to ensure that a connection has been established; the POS application on the handheld device is exited and File>Download Database is clicked on or the Download Database icon from the toolbar is clicked on. If there is an existing menu database on the handheld device, the system will ask if the existing database should be replaced. Yes is clicked if existing database replacement is desired."), 10:58-11:12 ("As stated, the preferred embodiment of the present invention includes the use of and compatibility with GUI technology. A drag-and-drop approach is used for organizing the tree structure 2 in the generated menu. Drag-and-drop is also used for assigning modifiers (modifiers can be dragged from the modifiers window 5 and dropped onto the menu item 4 for assignment). In-cell editing results in fast editing of items in building the menus. Customizable fonts enable users to change font types, style and size. Customizable layouts enable users to resize windows, change icons and display preferences. The inventive approach provides for fully persistent storage between sessions, even if a session is improperly or abruptly terminated. Font and the tree state (i.e., which nodes are expanded/collapsed) are stored between sessions. Layout for modifiers and sub-modifiers list views (filter, columns,

formatting, font, etc.) are stored between sessions. The last database used is likewise stored between sessions. Splitter views allow the user to see different views at the same time. Each view is displayed on its own section of the screen. Views can be resized via the keyboard or a mouse by simply dragging the splitter in the middle."), 11:13-30 ("An automated function is provided to import existing POS databases into the inventive menu generation system and, as discussed above with respect to the detailed example of how to use the preferred embodiment, an automated download procedure is provided to transfer the desktop database onto a handheld device and/or Web page. Also as discussed, the preferred embodiment facilitates preview of the handheld device or Web page version of the POS menu on the desktop before downloading and configuration. Customizable desktop menu generation is contemplated, as discussed above, in the form of customizable fonts, columns, layouts, etc. The inventive approach also includes templates for common modifiers that can be assigned to similar menu items. The preferred embodiment also supports multiple databases, thus providing for the creation and storing of different menu databases on handheld devices such as breakfast, lunch or dinner menus. The user can then select the appropriate database to reflect the time of day."), 11:51-12:4 ("Advanced database functions are provided in the preferred embodiment of the invention, including an automated download process onto handheld devices and/or Web sites. In the preferred embodiment, the menu generation system of the present invention uses an API called ActiveX Data Objects ('ADO') for database access. ADO is useful in a variety of settings. It is built on top of OLE DB and can be used to talk to databases and, in the future, any data source with any OLE DB driver. Advanced querying is supported. The database can be queried on virtually all fields. Queries can be

built using SQL syntax for experienced users or can be created using a query builder which guides users through the creating process. Advanced error handling is supported. Errors occurring at run time can be trapped. A descriptive message is displayed to alert the user and provide error information. However, the application does not terminate when the errors happen. The source code is easy to maintain and modify, thus allowing for on time delivery of customized versions of the software. The advanced database functions produce well-designed databases that accommodate growth and scalability."), 15:1-9 ("While the preferred embodiment of the invention includes the selection of items from a master menu wherein the master menu is displayed using a graphical user interface, it is to be appreciated that any means for displaying the master menu to the user and generating another menu in response to and comprised of the selections made is encompassed by the contemplated invention. The invention encompasses the selection of nontextual symbols, characters, icons and the like, in addition to text, from a hierarchical tree menu or the like for generation of another menu comprised of such items."), FIG. 1, FIG. 2.

- Brown at [0026] ("A merchant may add a terminal by clicking a button that labeled, 'add terminal'. When the add terminal button is clicked, the terminal information is displayed and the merchant inputs the terminal information into the system. The merchant may designate the terminal as an internet terminal by clicking on the button labeled, 'internet.' The merchant now has a terminal for accepting payments over the internet. For example, if a merchant wishes to accept Visa and Master Card, the merchant would acquire a merchant account from a merchant account provider, add a terminal by clicking the 'add terminal' button and input the numbers provided by the merchant banking provider into

the terminal fields. The merchant may define a method of payment from a list that contains all the methods of payment that UTOPA.com accepts, for example, cash, Visa, Master Card, Discover, American Express, PayPal, etc. The payment options that a merchant may accept is listed as options that appear at the checkout page on UTOPA.com, the external website and the point of sale in the merchant's retail store. All three sales avenues will therefore share the same methods of payment."), [0037] ("Item and item quantity management. The point of sale engine manages how an item appears on the online store and on the retail store's point of sale system, and controls item price. When a new item is added to inventory, a plurality of parameters about the item are available for input. The merchant has the option to be detailed and input many parameters or input few parameters. The minimum requirement for an item is a description and SKU number."), [0042] ("Once an item is created, an inventory number is assigned. A description field is displayed, whereby any text description may be entered. Further optional information may be input, which may include the location of the item, such as showroom or retail store, or the age and material of the item. Further information may include date or origin; quantity in stock; the merchant's vendor; maker of the item; dimensions and weight; shipping options; pricing, such as percent to mark up or exact price; where to sell, such as in store, to dealers, on UTOPA.com, or on the merchant's website; costing method such as first-in-first-out (FIFO), or last-in-last-out (LIFO); and vendor internal code. The optional information may include the condition of the item such as mint, excellent, good, fair, or poor. The merchant can manage each list of options for each item."), [0043] ("The inventory feature of the system provides several functions to the merchant at the item input interface. The inventory system communicates with the

purchase order system to provide an “allow auto order” function. For each item entered into the system, a purchase order is generated. The purchase order (PO) has a line item associated therewith representing the item. An item cannot be listed without a way to track it. When an item is initially entered, the quantity is zero by default, but can be specified by the merchant. The amount paid for the item by the merchant may also be specified.”), [0048] (“Many of the display boxes in the system are collapsible, so that a merchant can ignore a set of data and functions if it does not plan to use them. Also, for ease of use, the system can be set to remember the particular sets of boxes that are collapsed by clicking ‘save screen layout.’”), [0059] (“Each item may have as many as five images associated. The merchant may upload an image on to the system whereby the system will automatically resize the image, and adjust the image quality to achieve a maximum of about a three second load time for a customer to load an image of the item in the customer's web browser. A page that requires more than three seconds to load typically causes a potential customer to lose interest in the item. The system will automatically adjust the image to have about 500 pixels, while maintaining the aspect ratio, and maximizing the bit depth for the best resolution possible.”), [0061] (“The merchant may also display the (DSA) to a customer when viewing an item. The merchant may specify up to three lines of text for the DSA to communicate verbally to the customer when viewing the item. The DSA will appear with the item on UTOPA.com When the merchant clicks, ‘preview,’ the system displays the DSA and accompanying monolog customers will see and hear, when they view the item. Studies have shown that when a human face speaks about an item to a customer, there is a greater chance of selling the item.”), [0070] (“All screens in the POS are designed for use by either a

keyboard and mouse or by a touch-screen. The main point of sale screen provides information such as the software version, on-line status, location of register and/or register identification, transaction, date, time, and the user who is currently logged in."), [0073] ("The system further includes a website builder and manager. Referring to FIGS. 51-56, the website manager provides the merchant with options to display inventory on the merchant's website. For example, the system provides background color; font color; link color; font size; container height; font; border color; number of items to display; and container width; When the merchant specifies his desired options, he may click 'generate code,' whereby the website builder will generate html code and instructions on where to paste the code. The system generates code pertaining to inventory, login page, shopping bag, and 'my account.' The system also provides suggestions and templates for the merchant's website."), [0074] ("The inventory and point of sale system is completely based through the internet and is accessible through an ordinary web browser. The system is capable of handling a plurality of independent merchants, whereby each merchant may access the system through the internet with a web browser.").

- Nakamura (JP2004-164194A) at [0017] ("In these cases, it is preferable for the POS application program to have a manager master consisting of manager information regarding the persons in charge of handling the goods to be deposited; to further provide a manager information inputting step of inputting the manager information into the mobile phone in association with the deposit information or the product information; and for the output information to comprise the manager information."), [0049] ("The system server 30 also provides the mobile phone POS terminal 20 with a POS application program for executing accounting processing by wireless communication. Based on a

development request to the system server 30 from the store 21 side, this POS application program is created based on the settings after the system server 30 transmits a creation utility for the POS application program via the network 60 and the store manager inputs and sets necessary items. On the other hand, the store 21 pays a predetermined fee to the management company 31 as the development cost of the POS application program. The update of the POS application program (correction or rewriting of the master or table) can be appropriately executed by the store manager accessing the system server 30 and reading out the POS application program for the store."), [0064] ("Next, the control configuration of the system server 30 will be described with reference to the functional block diagram of FIG. 3. System server 30 is composed of an information acquiring unit 310 that acquires information from the mobile phone POS terminal 20 and the store server 50; a database storage unit 320 for storing the acquired information in the database 35; a data creating unit 330 for creating data relating to accounting processing based on the output information and inventory information output from the mobile phone POS terminal 20; a POS application creating unit 340 for creating a POS application program; a customer service processing unit 350 that processes various services for the mobile phone customer terminal 40; and an outputting unit 360 that outputs various information to the mobile phone POS terminal 20, store server 50 and mobile phone customer terminal 40."), [0066] ("Also, the database storage unit 320 stores information acquired by each part of the information acquiring unit 310 (output information storage unit 321, inventory information storage unit 322, POS application program storage unit 323 and email information storage unit 324); of these, the output information and inventory information are used for creating accounting processing data by the data creating unit 330

and are transmitted to the store server 50 via the outputting unit 360 at any time. Also, the POS application update information is used for program data update by the POS application updating unit 342, and the email information is sent to the email transmission unit 352. Further, the POS application program storage unit 323 stores POS application programs corresponding to the number of customer stores, which are updated at any time based on the POS application update information. Therefore, the POS application updating unit 342 reads out the POS application program for each store from the POS application program storage unit 323 and updates it."), [0070] ("The POS application creating unit 340 is provided with a POS application development reception unit 341 that receives a request for development of a POS application program from a customer store 21 that is a customer to the management company 31, a POS application updating unit 342 that updates the POS application based on update information transmitted from the store server 50, and a creating utility providing unit 343 for providing a creating utility for creating a POS application program in response to a POS application program development request."), [0071] ("In the creation utility providing unit 343, as a creation utility, an execution program for setting screens for inputting information for creating product masters, amount tables, manager masters, customer masters, etc. is sent to the store server 50 (see FIGS. 12 and 13), and the data are created by the manager of the customer store 21 entering necessary items in the setting screen. The created data are returned to the system server 30, and based on this, the creating utility providing unit 343 is able to create a semi-order format POS application program. Also, the creating utility providing unit 343 transmits the created POS application program to the specified mobile phone POS terminal 20 via the outputting unit 360 and stores it in the POS application

program storage unit 323. Additionally, instead of sending the creation utility to the store server 50, the system server 30 may have a configuration in which the store manager accesses the system server and inputs necessary items in the setting screen. By providing the creation utility in this way, the manager of the customer store 21 is able to easily input these information items, and the system server 30 is also able to easily and quickly create a POS application program."), [0108] ("As described above, according to the accounting processing method, POS application program, recording medium, and mobile phone POS terminal 20 of the present invention, by simply downloading the POS application program to a commonly used mobile phone, payment and settlement processing can be performed on a normal stationary POS terminal (accounting processing is performed based on deposit information; furthermore, since the processing of storing and outputting this can be performed, there is no need to introduce an expensive POS terminal, and the POS system 10 can be constructed at low cost. In addition, by inputting the deposit information using the operation keys of the mobile phone, the operation is easy to perform and the system can be used easily."), [0115] ("Also, in the example above, the system server 30 receives a request for development of the POS application program from the customer store 21 via the network 60, and provides the POS application program creation utility in response to the development request. The system provider of the management company 31 may provide a program developed and tuned for a specific store in response to a program development request from the store 21. According to this configuration, an easy-to-use program can be developed according to the business form of each store 21, and the customer store 21 does not have to spend time and effort to create a POS application program."), [0117] ("In the above example, the

POS application program is mainly updated by the store server 50; however, when one mobile phone POS terminal 20 is centrally controlled for one store server 50, in a configuration without store server 50 (that is, in store 21, accounting is performed only by mobile phone customer terminal 40), a configuration in which the POS application program can be updated on the mobile phone POS terminal 20 side may be employed. Also, in this case, it is preferable for the updated information to be transmitted to the system server 30 without fail. According to this configuration, even if the price of the product changes frequently, such as a small store or a store that handles fresh food, since the amount table and the like can be changed from the mobile phone POS terminal 20, it is possible to respond flexibly.").

- Jones (US2005/0182680) at Abstract ("A master copy of a menu is stored in the server, and changes to the master copy are automatically made to local copies stored in the hand-held and touchscreen units. Thus, all copies of the menu, in all of the units, are quickly and automatically synchronized. The system allows a manager to indicate the quantity remaining of a particular item, and that quantity is updated, in all of the hand-held and touchscreen units, as orders for that item are placed."), [0054] ("1. The present invention enables all of the wireless hand-held units 1, and all of the stationary touchscreen units 3, to be linked to a central database, such as database 8 associated with server 5, the database containing a menu of items available at the restaurant; 2. The system provides means for quickly and continuously updating the status of the menu, such that all units connected to the system will display current and updated information to waiters and waitresses; and 3. The system provides means for alerting users of the system, including all waiters and waitresses, concerning the quantity remaining of selected menu items."),

[0058] (“The invention also enables a manager to make permanent changes to the restaurant menu.”), [0059] (“FIG. 2 illustrates part of the programming of the system, in one preferred embodiment. FIG. 2 represents the software that permanently modifies the content of the restaurant menu. In block 20, the restaurant menu is modified in the server database. This modification, which could be the permanent addition or deletion of a menu item, is accomplished by modifying database 8, using the server 5. This modification would normally be made by a manager, at the server location. In the preferred embodiment, this type of modification cannot be made through the hand-held units 1 or the stationary touchscreen units 3.”), [0068] (“The menu modifications contemplated by FIG. 2 include permanently adding or deleting menu items. FIG. 3 deals with temporary changes, and shows how a manager can “enable” or “disable” a menu item. As used in this specification, the term “disabling” means temporarily removing an item from the menu. The most common reason for disabling an item would be that the supply of that item is exhausted. The term “deleting” means permanently removing the item from the menu. As shown in more detail below, one difference between “disabling” or “deleting” an item is that a disabled item still appears on the screen displays of the various hand-held units, and the stationary touchscreen units, but with an indication that the item is not available. A deleted item does not appear in any form.”), [0069] (“The flow chart of FIG. 3 is generally similar to that of FIG. 2. However, in the preferred embodiment, the enabling and disabling of an item, represented by FIG. 3, is performed through one of the stationary touchscreen units 3, and not through the server 5.”), [0083] (“FIG. 5 represents a typical screen display, generated by the “Manager Helper” software that is loaded on the server 5. The manager uses the Manager Helper software module to make permanent

changes to the restaurant menu. This software may be accessed at the location of the server 5, or, more preferably, at any of the various stationary touchscreen units 3."), [0084] ("The display shown in FIG. 5 shows that the manager is viewing the "Dinner Menu", and the category "Main Course", the latter fields having been highlighted on the left-hand side. The menu items pertaining to main courses on the dinner menu are shown on the right-hand side. Each item is shown adjacent to a column of "forced modifiers". A forced modifier is a property of the menu item that the waiter is obligated to enter. For example, "London Broil" is associated with the forced modifier "Temp" (for "temperature"), indicating that the waiter must specify whether the food is to be "well done", "rare", etc. A "regular modifier" is a property of the menu item which is optional, and which the waiter may enter, as part of an order, at the request of a customer. The manager can click on the indicated tabs to view and edit the regular and forced modifiers."), [0085] ("To add an item to the menu permanently, the manager presses or clicks on the "Add" button, which is encircled in FIG. 5. The system responds with a second screen, shown in FIG. 6, onto which the manager can enter the new menu item, using a keyboard associated with the server. In the example shown, the item to be added is "Salmon". After entering the details associated with this item, the manager presses the "Add" button, shown encircled in FIG. 6. The result is shown in FIG. 7, wherein "Salmon" now appears as one of the menu items on the dinner menu."), [0092] ("FIG. 13 depicts a typical display that appears on one of the stationary touchscreen units 3. This display, titled the "Manager Screen", allows the manager to access any item on the menu, and to enable or disable that item. In the example given, the manager has chosen "Main Course" from among the buttons at the upper right-hand side, and the corresponding

menu items appear as buttons at the lower right-hand side. To disable an item, the manager selects the item, and presses the button labeled “86 Item”. In the restaurant trade, the term “86” is typically used to indicate disabling of a menu item, and the term “68” is used to indicate restoration of an item. When the manager presses a button to enable or disable an item, a message appears, confirming that the status of the item in the system has changed.”), [0093] (“In the example represented in FIGS. 13-15, the manager wishes to disable “Calves' Liver” from the menu. As shown in FIG. 13, the manager highlights this menu item, and the left-hand side indicates the status of “Calves' Liver”. Since calves' liver was originally available, FIG. 11 shows the “86 Item” button as available, and shows that the “68 Item” button is grayed out (i.e. unavailable for use). When the manager presses the “86 Item” button, a message appears, confirming that the status of the menu item has changed, as shown in FIG. 14.”), [0094] (“In FIG. 15, the status of calves' liver is now indicated as unavailable, and the “86 Item” button is grayed out, while the “68 Item button” is available for use. Thus, FIG. 15 shows that the manager now has the option of restoring calves' liver to the menu.”), [0098] (“The manager places an item on count in the following manner. First, the manager uses a main menu screen (not shown), generated by the software, to reach the screen labeled “Put Item on Count”, as shown in FIG. 20. The menu appears on the right-hand side, possibly divided by category. Thus, by clicking on “Main Course” in the upper right-hand screen, the manager sees a display of main courses in the lower right-hand screen.”), [0099] (“The manager then browses the menu, as displayed, and selects the item to be placed on count. The manager enters the value of the count, using the numeric keypad that appears

at the left-hand side of the screen, and presses the “Set Count” button. The manager then receives a confirmation (not shown) that the item has been placed on count.”).

- Blinn (US5897622) at 3:1-19 (The present invention enables merchants to enter the online shopping market by providing a system and architecture to obtain and perform a large set of processing operations and computations on a rich set of dynamically generated information from a wide variety of data sources. In contrast to the rigid display formats and fixed database schemas of existing merchant systems, the present invention provides a dynamic page generator that permits the display of dynamically generated data in any format or presentation desired by the merchant. The present invention provides this flexibility through the use of display templates and a database schema independent query mechanism. In this manner, a merchant changes the display formats by modifying a template instead of revising the system modules producing the display formats. Similarly, a merchant handles database modifications by modifying the queries stored in the database instead of modifying the system modules performing the database queries”); 4:44-46 (FIG. 6 shows a portion of a template, a product table and an access object which illustrate an example of the data flow of FIG. 5 used to produce a HTML table for display.); 9:15-17 (For example, actions can add an item to an order form, clear an order from, initiate a purchase or insert and delete data from the database.); 13:46-14:14 (For example, a template portion 200 includes the HTML and directives shown in FIG. 6. The directive fetchrows product "get-product"! 202 instructs the dynamic page generator 125 to retrieve and execute the query named "get-product" on the database 121. In a preferred embodiment, the "get-product" query is the SQL query "SELECT * FROM Product--table" and the database 121 is a relational database having product information in a

Product-- table 220. The dynamic page generator 125 passes the name "get-product" to the database module 127 which retrieves the SQL query "SELECT * FROM Product-- table" and passes it to the database 121 for execution. The database 121 returns the "get-product" query results to the database module 127 to incorporate into an access object 230 with a temporary name "product" for other directives in the template to reference. The product access object 230 includes a descriptor 232, a first row 234 and a second row 236. The HTML commands 204 instruct the browser 122, 123 to create and display a table having columns titled "Item" and "Price". The directive eachrow product! 206 instructs the dynamic page generator 125 to initiate a loop to iterate through the rows of the product access object 230, one row at a time. The directive /eachrow! 208 denotes the end of an iteration of the loop. Thus, the directive value product.sku! 210 evaluates the product.sku value reference and places the value into the current page. Similarly, the directive money product.list-- price! 212 evaluates the product.list-- price value reference, formats it as currency and places the formatted value into the current page for display. For example, for the first row 234 of the product access object 230, the directive value product.sku! 210 evaluates to "GLOVE" and the directive money product.list-- price! 212 evaluates to "$12.25". At the completion of the eachrow product! 206 directive, the dynamic page generator 125 passes a HTML file portion to the browser 122, 123 to format and display a table 240.); FIG. 6

- Blinn II (US6058373) at 1:65-2:9 (“The present invention provides a method and system for processing electronic sales transactions. In a preferred embodiment, the electronic merchandising system allows merchants to create electronic orders which are easily adaptable for different sales situations. The preferred electronic order comprises flexible

blackboards which allow merchants to add sales information with what are called key-value pairs. The sales transaction information stored in the key-value pairs may include, by a way of example, special shipping information, unique billing information, gift wrap information, monogram information, etc"); 7:64-8:11 ("The consumer client 102 contains a consumer browser 110. The consumer browser 110 communicates with the store server process 106 and displays the web documents created by the store server process 106. Each store server process 106 provides a server architecture that supports the presentation and administration of a virtual store. Preferably, the store server process 106 comprises a number of components including a dynamic page generator 120, an action manager 122, one or more orders 124 and an order processing module 126. Furthermore, in communication with the store server process 106 is a storage device such as a hard disk which contains HTML structures 128 which define the layout of different HTML pages. In addition, the store server process 106 is in communication with one or more databases 130 with a database module 132."); see also FIG. 1; 2:10-18 ("Unlike other order forms which rely on specific database structures or other structured formats, the order of the present invention contains as many key-value pairs as are necessary to define a sales transaction. This allows merchants to easily customize the electronic merchandising system for diverse sales transactions. Advantageously, merchants can add new key-value pairs or delete existing key-value pairs to modify the electronic merchandizing system"); 8:45-53 ("For instance, the key-value pairs define information about the order 124 such as who the shopper is, where to send the watch, what is the means of payment, etc. In addition, the key-value pairs identify information about the watch such as the watch's identification code, the number of desired watches, the style

and color or watch, etc. Because of the implementation of the unique order 124 in the present invention, different key value pairs can be easily added for different sales situations."); FIG. 1.

- Thompson (WO 02/39226) at 12-13 ("The network server program 48 of the primary server 40 is preferably stored in the internal memory device 44 and executed upon initialization of the primary server 40. It preferably includes appropriate display routines for generating a set of display screens that together comprise the user interface for the site, as will be displayed at the POS 12 via the network appliance 38. To that regard, and as noted above, the preferred embodiment of the invention is preferably implemented within at least one primary server 40. Thus, the invention does not require any modifications to conventional POS 12 hardware or software. Furthermore, by housing the software applications on the primary server 40, the single-POS businesses 12 need not afford the cost of the equipment or software or staff to implement the system functionality that is being described and had previously been reserved for the multi-POS businesses 14.");
- Showghi (US6920431) at 3:27-44 ("It is a further object of this invention to provide a system which automatically checks the location information of the patron against a database of locations within the venue and determines the validity of the location and the level of available service and notify patrons if service is not available in that area, or offer only a menu of items specifically to be made available in that service area."); 5:14-34 (FIG. 1 depicts a system that uses a custom, application specific, local area, wireless data communication network. As patrons arrive at a venue/facility for a concert, game or other event, they will be presented with the opportunity to obtain a battery-operated wireless

remote ordering device 10, illustrated in FIG. 3 *d*, which will be available at a kiosk having a device rental station 12. Devices 10 are operable, as later described, to communicate an order for menu items to a transceiver node 14, which communicates orders from multiple remote ordering devices 10 to a fulfillment center having server 16 and order processing computers 18. In this embodiment, the remote ordering devices 10 and the base transceiver nodes/stations would be designed specifically for this application using available transceiver chip sets, and other off-the-shelf components such as keypads, microcontrollers, displays and the like which are commonly known to those skilled in the art. In FIG. 1, the base transceiver nodes 14 and other client work stations are networked to the local server computer 16 which maintains system records and facilitates the overall operation of the system.), FIG. 1

- Brown (US2007/0175992) at [0037] ("Item and item quantity management. The point of sale engine manages how an item appears on the online store and on the retail store's point of sale system, and controls item price. When a new item is added to inventory, a plurality of parameters about the item are available for input. The merchant has the option to be detailed and input many parameters or input few parameters. The minimum requirement for an item is a description and SKU number."), FIG. 2
- Costello (WO 01/65427 A1) at 9:7-11 ("Where a POS sales transaction or an adjustment is made by direct interface, the inventory and user changes output process 44 broadcasts to the POS devices that receive this inventory in the reference data. A broadcast may be, for example, by a JMS compliant messaging system. This is received by an inventory and user changes input 38 of each POS device"); 11:24-12:1 ("Referring to Figure 5, the back office management sub-system provides control and configuration functionality

necessary to configure applications. The functions are grouped into a Global Management Layer 120 and a Client Management Layer 122. The Global Management Layer deals with information for all users of the system such as master product files, industry specific inventory lists, master configuration perimeters, application components, etc. The Client Management Layer deals with information specific to a client, such as stock item lists, customer lists and specific configuration of the POS device (terminal) to suit the needs of the client. Access to these functions can be limited to a systems administrator of the particular client."); 14:12-15 ("a) The point of sale system provides a great degree of flexibility, as updates to the POS program may be downloaded at the initialisation of the POS device thereby incorporating any amendments or improvements to the point of sale program functionality;"); FIG. 5

- Redmond (WO 03/065178 A2) at [23] ("The POS device 20 is the primary service delivery component of the POS Portal. The POS device 20 can be configured to deliver specific services subscribed to by a particular merchant. The POS device 20 is an intelligent network device, i.e., it can host applications and process transactions from a wide range of systems and services such as the cash register 22, the in-store appliance server 28, the merchant's back office applications 24 and third party services from affiliated providers via the Internet 31. An exemplary embodiment of the POS device 20 has the following operating characteristics: an embedded real time operating system, a built-in Internet browser , a touch sensitive screen, audio supports a number of peripherals (magnetic stripe, smart card reader and check readers, keyboards, printers, PIN pad, etc.) supports high resolution color graphics and signature image capture supports Internet protocols and networking (TCP/IP, Ethernet and Wireless 802.1 lb,

Bluetooth, etc.) interfaces to the electronic cash register environment (ECR) supports field upgradeable and network loadable system and application modules, supports EMN specifications, supports PKI based security specifications"); [29] (In an exemplary embodiment, the in-store appliance server 28 is implemented for a large merchant utilizing a number of servers which support packet filtering firewalls, routers, load balancing units, application servers for security, performance monitoring and screen presentation for POS devices. Exemplary characteristics of the hardware used to implement the in-store appliance server 28 are, for example, high reliability and availability, and support for server clustering and plug and play components. Physical redundancy for processors, memory, disk and power supplies are also included. For a smaller merchant this configuration can be scaled down and still provide basic in-store functionality. [30] The services syndication hub 30 acts as the merchant's gateway to third-party services , and coordinates the transactions between a merchant's in-store appliance server 28 and the affiliated service providers 31. The services syndication hub 30 is primarily enabled by network connectivity and middleware software interfacing to the service providers 31. The communications between the services syndication hub 30 and the third party service providers 31 may be conducted via a private computer network, such as, a virtual private network 32. In general, portions of the service processing are carried out at the data centers of the service providers 31 while the services syndication hub 30 performs a packaging and distribution role"); claim 3 ("The system of claim 1 wherein at least one of the one or more computing devices is configurable to allow one or more of the plurality of services to be presented in one or more configurations");

- Halo
    - *Juice It Up! Moves to Web-based POS*, Jun 6, 2008, at 1:
        - IRVINE Calif. Juice It Up!, the 150-plus-unit chain with headquarters here, is switching to Halo, Vivonet Inc.'s centrally hosted, Web-based subscription point-of-sale system.
        - They said Halo's secure architecture will support Juice It Up! management and franchisees in making strategic decisions based on real-time information about different regions, menu items and store performance, while letting them view reports and change menu items from anywhere at any time using a Web browser.
    - Medina, Ahmed H., *An Analysis of Market Development Strategy of a Point-of-Sale Solutions Provider's Market Research Database*, at 3:
        - HALO, which Vivonet introduced in 2004, is an ASP (Application Service Provider) based product which only requires a POS terminal connected via Internet to a central computer server maintained by Vivonet. The ASP technology allows the application logic and data to exist on a remotely located computer server, which the application users access over the internet. In Halo's case, ASP allows multiple outlets of a restaurant chain to share information and lowers the complexity of maintaining the POS system. As of 2006, Halo's customer base has grown to 200 customers.
        - Halo's main features include:
            - A touch-screen terminal (shown in Figure 1) installed at the customer site. Halo does not require any computer server at the

restaurant location to host the POS software; instead, the touch screen terminal connects via the public Internet to a central application server located at Vivonet's office. . .

- Halo Enterprise Manager, which allows the restaurant to configure its POS system by setting menu process and promotions. Restaurants can operate the feature through any web browser.

- Medina, Ahmed H., *An Analysis of Market Development Strategy of a Point-of-Sale Solutions Provider's Market Research Database*, at FIG. 2
- Halo For Restaurants, Nov. 20, 2006, at 1:
  - Halo for restaurants is a web-based, touch screen management solution that helps boost your bottom line by consolidating equipment and giving you easy access to important information.
  - With Halo for Restaurants, you have the ability to monitor and manage a single store or your entire chain from anywhere you have access to a web browser. Change prices, add menu items, manage your touch screen terminals and drill down to the transaction level of detail in real time.
  - Replacing your cash register and payment processing terminal with a point of sale (POS) system is a great way to take your business to the next level. Few purchases can have as dramatic an effect on your restaurant as a POS system.

- Restaurant POS Systems, Halo, https://web.archive.org/web/20080612142133/http://www.myhalo.com/restaurant/restaurant, at 1:
    - Halo restaurant POS systems are designed for small and medium sized operators who are interested in saving time and money, improving business decisions and increasing customer service. Halo for restaurants is a web based, touch screen management solution that helps boost your bottom line by consolidating equipment and giving you easy access to important information.
    - With the Halo point of sale restaurant system, you have the ability to monitor and manage a single store or your entire chain from anywhere you have access to a web browser. Change prices, add menu items, manage your touch screen terminals and drill down to the transaction level of detail in real time.
    - Replacing your cash register and payment processing terminal with a restaurant POS system is a great way to take your business to the next level. Few purchases can have as dramatic an effect on your restaurant as a point of sale system.
    - Run your restaurant as efficiently as possible with a secure web based POS system for restaurants that meets the needs of your specific restaurant dining type.
- T-Net, Today's News, *Vivonet Completes Major Agreement with Sodexo,* July 19, 2010, at 1-2.

- Vancouver-based Vivonet was the winner in an evaluation process that included all major POS providers. Halo, Vivonet's Software as a Service ("SaaS") product, is a flexible and affordable solution that is delivered to restaurant and foodservice clients simply and securely over the Internet. Halo enables real-time transaction processing, acquires centralized point of sale, product, pricing and people data and organizes it all to be securely analyzed, reported and shared with partners. In the fast-paced foodservices industry Halo is in stark contrast to traditional POS solutions by its ability to eliminate the need for expensive and complicated site specific equipment and on-site IT support.
- This agreement sets a new benchmark in the retail and restaurant technology sector," said Ryan Volberg, president and CEO of Vivonet. "Our partnership with Sodexo represents the largest ever Software as a Service point of sale rollout and one of the largest hospitality point of sale rollouts in North America. We believe that a transaction of this scope is a validation of Software as a Service as a legitimate technology solution for clients in the restaurant and foodservice industry who need better data to make better decisions."
- In North America alone, Sodexo serves ten million customers a day in corporate offices, health care facilities, long term care facilities, retirement homes, schools and college campuses as well as government, military, and remote sites. Sodexo employs 120,000 people and operates an estimated 6,000 sites in North America.

- Halo will provide Sodexo with point of sale and payment processing capabilities as well as robust web-based enterprise management and reporting delivered via web-browser to Sodexo locations all over the continent. Halo will allow Sodexo to make better and faster product decisions and improve supply chain optimization. Put simply, it will make doing business easier and more efficient. It will be installed in Sodexo's industry-leading dining facilities and in their retail brand locations such as Jazzman's Cafe, Panini's and Sub Connection. Halo will also be installed in the leading national brands operated by Sodexo Sodexo will use this new technology to improve the overall experience of our customers and thereby help our clients provide the best possible quality of daily life environment," said Sodexo President and CEO George Chavel. "Halo will help us meet the challenges of our ever-changing competitive landscape and rapidly evolving customer tastes."
- The implementation of Halo will allow Sodexo to track the movement of a single product across all their brands. Behind the scenes, Vivonet's Item Harmony technology standardizes Sodexo's tens of thousands of product offerings into a single, standardized item file allowing Sodexo to better understand customer demand, manage supply chain and optimize product pricing.

- T-Net, Today's News, *Vivonet Introduces Halo Retail POS*, April 19, 2007, at 1
  - Vivonet Introduces Halo Retail POS (Point of Sale), an 'All-in-One' Cashiering and Payment Processing Solution for Small to Midsize Retailers.
  - http://www.vivonet.com
  - Vancouver, BC, April 19, 2007--(T-Net)--Vivonet announced today that it unveiled Halo Retail at the Electronic Transaction Association annual meeting and expo in Las Vegas, Nevada.
  - Halo Retail is a complete solution designed to replace the electronic cash register currently being used in 74% of all small retail operations. Designed for small and medium retailers that have been using cash registers, Halo is an all-in-one web-based touch-screen point of sale and payments system that protects retailers from security breaches, fulfills payment processing needs, and gives them easy access to their critical business information from anywhere they have access to the web.
  - "Small and medium sized retailers finally have an affordable end-to-end solution that allows them to re-focus their energy on their core business needs to further their success rather than managing complex PC technology" said Ryan Volberg, CEO and Founder of Vivonet Inc. "The modular PC on Cash Drawer solutions currently in the market are still out of reach for the retail operators that need these tools the most. Halo Retail was designed specifically with their needs in mind including the need for affordability."

- The Halo touch screen terminal is connected to the internet ensuring all department, category, and product sales performance information is automatically stored in a secure data center and easily accessible by store owners and managers from a web browser anywhere, anytime. Now even the smallest retailers can have the kind of rich historical information and business insight that was once only available to the biggest retailers – and without all the time-consuming challenges they faced managing their business with z tapes and spreadsheets.
- "Halo has taken away a ton of paperwork virtually overnight, and really helped me improve my business" said Matt Wagstaffe, owner/operator of North Shore Linens and Halo Retail early adopter. "I would recommend Halo Retail to anyone using a cash register,"
- Halo Retail is set for general availability throughout North American in the summer 2007and will be sold through a channel network of Independent Sales Organizations (ISO) and Agents. More information on this program can be found at www.myhalo.com.

- The Green Sheet Online Edition, *Look to the light: Retail POS system can replace ECR,* June 7, 2007, Issue 07:06:01, at 1:
  - Halo is a Web-based all-in-one cashiering and payment processing solution designed to replace the electronic cash register (ECR) that many small and medium-sized merchants use.

- Halo includes a touch-screen terminal connected to the Internet. The system tracks details of every transaction (cash, credit and debit) and sends that information via the Internet to a secure data center off-site.
- Real-time department, category and product sales performance data are accessible by storeowners and managers from a Web browser anywhere, anytime. Even when the Internet connection goes down, the Halo offline manager keeps store operations running.
- And since the data is housed off-site on the Halo secure server, merchants don't have to worry about losing information in a hard drive crash, theft or natural disaster, such as fire or flood.
- Halo relieves some other burdens as well. With an ECR, merchants must back up data and update systems with new versions. With Halo, upgrades are free and take place during the night.
- Since Halo is used for credit card and PIN-based debit card transactions, it eliminates the need for stand-alone terminals and the double entry and end-of-day reconciliations they can require.
- Halo also integrates multiple locations and enables merchants to compare them in real-time and change items and prices instantly.
- Additionally, Halo is secure. It reportedly is compliant with the Payment Card Industry (PCI) Data Security Standard and has achieved the Visa Cardholder Information Security Program (CISP)certification for level 1 service providers.

        - Halo Retail POS will be available to retailers this summer. The product will be sold through ISOs and merchant level salespeople.

- Flashpoint
    - Flashpoint Operations Guide, Menu Setup Manual, 3-90.
    - Flashpoint News (Jan. 18, 2007) ("Flashpoint version 14 is released. Features include a complete Enterprise solution along with above store reporting, centralized menu setups, system setup, and centralized employee management.").
    - Flashpoint Operations Guide, System Setup Manual, 4-140
- NextPOS-Aldelo
    - Aldelo for Restaurants Training Manual, Chapter 7 – Information Data Entry, 45-60 ("Menu Setup")
    - Aldelo for Restaurants User Manual, 54-72 ("Menu Setup")
    - Aldelo NextPOS Software Manual, 101-111 ("Restaurant Menu Setup")
    - Aldelo NextPOS Product Overview, https://web.archive.org/web/20030228034043/http://bcdata.com/ (Feb. 28, 2003) ("With 100% touch screen operated point of sales, super easy to follow screen layouts, and the simplicity of Windows based interface, your wait staff will be able to use the software in a matter of minutes, rather than days!")
    - NextPOS Corporation and Transactional Web, Inc. Partner to Replace Expensive and Labor Intensive Reporting and Accounting for Restaurants with Affordable Subscription-Priced Web Automation, https://www.prweb.com/releases/2004/11/prweb183924.htm#! (Nov. 2004) ("NextPOS for Restaurants software was designed with the independent

restaurant owner in mind. The design concept was to create a software that would be easy to setup and use by even a computer novice. Although the software is easy to use, it has all the features of the more expensive Point of Sale software programs on the market with a price that even the smallest restaurant owner can afford.")

    - Aldelo for Restaurants, https://web.archive.org/web/20060521002319/http://www.aldelo.com/ProductsandSolutions/AldeloForRestaurants/tabid/67/Default.aspx (May 20, 2006) ("Aldelo For Restaurants is full of benefits. It is a user friendly Restaurant POS software that does not require you to know about computers to use it."; "Menu Recipe Control & Complete Inventory Management")

- PointofSuccess
    - Restaurant Software for POS Systems, https://web.archive.org/web/20060104044810/http:/www.pointofsuccess.com/restaurant-pos-software.htm (Jan. 4, 2006) ("Customize ordering menus to fit your way of doing business . . . Accounting and tax reporting information is retrieved from the manager's workstation with a few simple commands.").
    - Order Entry and Preparation, https://web.archive.org/web/20051222105812/http:/www.pointofsuccess.com/orderentry.htm (Dec. 22, 2005) ("Define Products and Order Entry Menus. Some other point of sale systems' menus are so difficult to design that the job requires a technician. With Point of Success, menu design is as easy as placing buttons on the screen and setting the product code they activate. Menu design is even easier

with the available training CD. When menus are scheduled to change, the changes can be made and stored in an inactive area of the menu designer, then added to the active menu with a few mouse clicks. Temporary or seasonal menu changes are almost automatic! Your menu is easy to set up in Point of Success. Select the products to appear on the menu and organize the menu to fit your way of doing business. The only limit to the number of buttons on a screen is the size of the computer display. Buttons can be sized, colored and located anywhere on the screen.").

  - Point of Success, https://web.archive.org/web/20060101024912/http:/pointofsuccess.com/ (Jan. 1, 2006) ("Point of Success software is made to be set up and used by a restaurant owner or manager")
  - Feature Comparison for Point of Success Standard and Point of Success Premium, https://web.archive.org/web/20060104041129/http:/www.pointofsuccess.com/comparison.htm (Jan. 4, 2006) ("Change order entry background screens to customize image for a restaurant . . . Customize the order entry menu to fit your restaurant").

- Squirrel
  - SquirrelOne Solutions, https://web.archive.org/web/20051201101248/http://www.squirrelsystems.com/solutions/index.html (Dec. 1, 2005) ("SquirrelOne's open architecture, built on a Microsoft SQL relational database, provides on-demand access to real time information. Make menu and price changes instantly, at your convenience.

Review and analyze live data from any location. Know what's happening in your operations now, not tomorrow.").

- Corporate POS, https://web.archive.org/web/20060327174026/http://www.squirrelsystems.com/solutions/corppos.html (Mar. 27, 2006) ("Corporate POS Manager facilitates the centralization of menu management, freeing restaurant managers from the burden of making menu changes. New information can be sent to locations at any time and can be automatically incorporated into the local database, whenever it's convenient. Built on the latest Microsoft SQL Server relational database which powers some of the largest enterprise systems today. Squirrel's POS Manager, with a familiar browser style desktop is able to be accessed directly from the central PC or remotely via the internet. It has an open architecture Ethernet network communication which is a world wide standard. Upload and download communications becomes routine with POS Manager. You can easily make changes to the menu database at the corporate center, and download them to the restaurants which can in turn perform the up-dates at any time. Or, changes can be pre-scheduled to update automatically (unattended if you prefer). Menu Management, Consolidated Corporate Reporting, and Event Manager for scheduling, are the components of Corporate POS Manager that help you control the many databases that are part of running a multi-unit operation. Menu Manager This Menu Management tool is, for chain restaurants…where the potato hits the plate. Menu Management is how large and smaller chains drive their menus. System-wide menu updates are made easily, at the drop of a napkin. Multiple

menus can be developed and managed from this single program. This solution enables corporate centers to define the scope of menu control for individual locations – some prefer to define a certain amount of latitude in food and beverage items, for example, while others choose to "lock down" all menu entries and make all decisions centrally. The choice in pre-configuring the program is entirely yours. Menu management is also where content is created for the "workspace" area of POS terminals. Head office/store parameters for the rest of the display area are also determined here.").

- Point of Sale, https://web.archive.org/web/20060101034917/http://www.squirrelsystems.com/solutions/pos.html (Jan. 1, 2006) ("The results? Our POS modules are so user-centric and so intuitive, they give 'ergonomic' a whole new meaning. For servers, kitchen staff and managers, we believe Squirrel POS will bring new ease, accuracy and speed to every task in your foodservice operations. How easy? If you can use an ATM, you're ready for SquirrelOne POS. Most of our customers achieve lower training costs immediately.").

3. Point of Sale Graphical User Interface ("POS GUI References")

The Asserted Patents purport to provide a solution to the long-recognized problem of "online, web-based point of sale (POS) building and configuration, which can assist non-expert business operators in building, editing and testing a point of sale system to manage their businesses." *See* '012 patent, Abstract. The Asserted Patents state, "[c]urrent practice in the field of assembling point of sale systems includes manually coding front-of-screen information. Typically, this front-of-screen information contains menu selections, page selections, and general answers to business questions. This front-of-screen menu is typically manually coded by a

business expert with the help of a programmer or data expert. Also, currently the entry of this front-of-screen information requires intimate knowledge of a complex interface to a front-of-screen programming language. In summary, current practice includes the manual building of a point of sale (POS) screen. This manual process requires defining the position and operation of touch screen keys and their database correspondence. Currently, only specially trained people can build or change POS screens. This manual POS building and editing is prone to mistakes and is time-consuming. Since POS screen changes are difficult and prone to error, store owners tend to retain older, inaccurate, out-of-date POS screens in order to avoid the POS screen editing process. Also, current POS screen editing occurs off-line with the testing of the screens occurring at a later date, at a remote store location." '012 patent, 1:31-53. The Asserted Patents describe that "[w]ith this invention, the store operator will be able to build his POS screens online over the Internet. With input from the store operator, the POS builder can specify and display the number, shape and arrangement of selection keys or buttons on said POS screens. The store operator, who does not have to be technically trained, will be able to edit and test his screens until he is satisfied with the end results." *See* '012 patent, 3:10-16.

The benefits of point-of-sale (POS) graphical user interfaces (GUIs) were known long before the Asserted Patents. Among other benefits, POS graphical user interfaces allow fast and easy updates to POS menu screens, such as those in restaurants or retail establishments, with no or minimal special training. Graphical user interfaces allow nearly any user to quickly understand how a system works, provide input to the system, and understand output from the system. A POSITA would have recognized that the general public is very aware of, familiar with, and competently able to use computing devices (e.g., smartphones, tablets, laptops, desktops, televisions, set-top boxes) with graphical user interfaces, which allow a user to provide

input to the computing device using a graphical interface (e.g., as opposed to a text-based interface, such as UNIX or DOS). Devices with graphical user interfaces are extremely well known, so a POSITA would have been motivated to implement a graphical user interface in a POS system—whether a POS terminal or a POS builder—to achieve the benefits of a graphical user interface in the POS system.

POS graphical user interfaces are ubiquitous in the prior art, for example as described in the following ("POS GUI References"):

- Woycik at [0013] ("The administration tool application includes a menu editor. The menu editor enables the administrator to create and edit the interactive menu screens provided by the self-order application. The menu editor accesses a library of menu templates containing predefined groupings of buttons, and uses the menu templates, as directed by the administrator, during creation and editing of the interactive menu screens."), [0073] ("As will be discussed in greater detail below, the client software is a computer program that is accessed by the user via a user interface application written using a multimedia authoring program. For customer ordering of food items, the user interface application is a self-order application that displays a graphic-intensive user interface that allows the customer to select and customize the desired food items, specify quantity, options, etc. Additionally, the local server kiosk 16 further includes an administrative tool comprising a second user interface application that can be accessed by the administrator to perform various administrative functions such as configuring kiosks, creating and editing menus and available food items, and specifying tax and payment features of the system. This administrative tool is also written using a multimedia authoring program that provides an administrative user interface to the client program.

During normal use, the server process runs in the background while the food ordering client software application provides the same user interface (screen display, touch screen input) as the other client kiosks 30. When access to server administration is needed, the store owner or other administrator can switch to the administrative tool on the server kiosk 16. As will also be discussed further below, the system also allows the administrative tool to be run from any of the local clients 30, and this can be done, for example, by loading the administrative tool on each client computer 30."), [0095] ("The local server kiosk (16, FIG. 1; 82, FIG. 3) includes an administrative tool (98, FIG. 3) to run reports, check the system status, add or remove kiosks from the system, or edit the screen displays that the customers see. All of these settings are stored on the local server 16, 82. The administrative tool 98 allows an administrator to modify these settings via a local kiosk touch screen. The administrative tool 98 is built using the same enhanced vector graphics approach (e.g., FLASH and .NET) described above for the self-order application interface so that administrators can setup menus and items, and change configuration items primarily via a kiosk touch screen. Therefore, the administrative tool 98 provides a graphical representation of menu options, button selections, images, text fields, etc. on the kiosk. Those items are readily editable through the graphical administrative tool 98 as will be discussed in more detail below. An advantage of this approach is that it enables administrators and store owners to make changes to the store offerings and customer interface without requiring any knowledge of the underlying code. In addition, administration is available from any client kiosk (30, FIG. 1; 80, FIG. 3) configured to communicate with the server 16, 82. This can be implemented by, for example, storing the administrative tool 98 on each client kiosk 30, 80 or by providing

the system with the ability to execute the administrative tool 98 on the server 16, 82 using an application interface located on the client kiosks 30, 80. Where the administrative tool 98 is separately stored on each client kiosk 30, 80 and run locally on the client, the tool can be configured with the server IP address of the server kiosk 16, 82 so that, prior to giving the administrator access to change the kiosk, menu, and item setups, it verifies that there are no other instances of the administrative tool 98 running. Also, in this implementation, the configuration data for the kiosks, menus, and items can be stored on the server kiosk 16, 82 and downloaded to the client kiosk 30, 80 when the administrative tool 98 is run at the client. Changes to the configuration are then uploaded back to the server kiosk 16, 82 when the administrator is done. Thus, there is no need to go to the server 16, 82 in order to use the administrative tools 98 or to modify any of the settings in the client interface. Moreover, remote administration using the administrative tool 98 can be carried out in any of the various ways discussed further above."), [0098] ("Selecting 'Menu Editor' 220 launches the menu editor shown in the screen 250 depicted in FIG. 11. The menu editor enables the administrator to create and edit the screens or menus that customers see. The main menu in the menu editor lists all menus and submenus 254 created for the client interface. The administrator may change the menu order on the list by touching up or down arrow buttons 256 and 258, respectively, next to any menu 254 (except the top/main menu) in the list. Note that the order on the menu editor does not correspond to the order that customers see the menus when placing an order, but merely allows the administrator to organize and group the menus in the editor. From the menu editor, the administrator can configure the overall look and content of each menu (via the menu edit button 260) and, for that menu, can edit the individual buttons to associate

them with different items and options (via the edit items button 262). A menu is removed via the remove menu button 264."), [0111] ("In FIG. 12, the process flow is depicted by block 304 (selecting "edit items" button 262 in the menu editor main screen 250 of FIG. 11), block 306 (displaying edit items screen 420 of FIG. 20), block 308 (selecting the button type from pull-down list 426), block 310 (edit button attributes by selecting edit button 432), block 312 (using enable/disable button 424), block 314 (selecting OK button 434 when button editing is completed to return to the menu editor main screen 250 of FIG. 11), block 316 and block 318 (selecting Main Menu button 252 on menu editor main screen 250 to return to main menu 212 of FIG. 10).").

- McNally at 1:13-23 ("This invention relates to an information management and synchronous communications system and method for generation of computerized menus for restaurants and other applications with specialized display and synchronous communications requirements related to, for example, the use of equipment or sore with non-PC-standard graphical formats, display sizes and/or applications for use in remote data entry, information management and synchronous communication between host computer, digital input device or remote pager via standard hardwired connection, the internet, a wireless link, smart phone or the like."), 2:57-63 ("A principal object of the present invention is to provide an improved information management and synchronous communications system and method which facilitates user-friendly and efficient generation of computerized menus for restaurants and other applications that utilize equipment with non-PC-standard graphical formats, display sizes and/or applications."), 3:9-13 ("A further object of the present invention is to provide a synchronous information management and communications system which enables automatic updating of both

wireless and internet menu systems when a new menu item is added, modified or deleted from any element of the system."), 3:17-38 ("The foregoing and other objects of the present invention are provided by a synchronous information management and communications system and method optimized for simplicity of operation which incorporates menu generation for creation of menus to be used with wireless remote handheld computer and PDA devices, the internet or any application where simple and efficient generation of menus is appropriate. The menu generation approach of the present invention includes a desktop software application that enables the rapid creation and building of a menu and provides a means to instantly download the menu configuration onto, e.g., a handheld device or Web page and to seamlessly interface with standard point of sale ('POS') systems to enable automatic database updates and communication exchanges when a change or input occurs in any of the other system elements. To solve the above and other related problems, an information management and communications system is provided which results in a dramatic reduction in the amount of time, and hence cost, to generate and maintain computerized menus for, e.g., restaurants and other related applications that utilize non-PC-standard graphical formats, display sizes or applications."), 3:39-47 ("The menu generation approach of the present invention has many advantages over previous approaches in solving the problem of converting paper-based menus or Windows® PC-based menu screens to small PDA-sized displays and Web pages. In one embodiment, the present invention is a software tool for building a menu, optimizing the process of how the menu can be downloaded to either a handheld device or Web page, and making manual or automatic modifications to the menu after initial creation."), 4:59-65 ("The menu generation approach of the present

invention is the first coherent solution available to accomplish these objectives easily and allows one development effort to produce both the handheld and Web page formats, link them with the existing POS systems, and thus provides a way to turn a complicated, time-consuming task into a simple process."), 5:40-47 ("FIG. 1 is a schematic representation of a window displayed on a computer display screen which shows a hierarchical tree menu, modifier window and sub-modifier window in conformity with a preferred embodiment of the present invention. FIG. 2 is a schematic representation of a modifier dialog box in conformity with a preferred embodiment of the present invention."), 7:24-37 ("An aspect of the preferred embodiment of the information management and communications system of the invention is shown in FIG. 1. FIG. 1 shows an example of the GUI provided by the operating system of the preferred embodiment of the present invention. With reference to FIG. 1, the preferred embodiment includes an intuitive GUI I from which to build a menu on a desktop or other computer. A hierarchical tree structure 2 is used to show the different relationships between the menu categories 3 (e.g., soups, salads, appetizers, entrees, deserts, etc.), menu items 4 (e.g., green salad, chicken caesar salad, etc.), menu modifiers 5 (e.g., dressing, meat temperature, condiments, etc.) and menu sub-modifiers 6 (e.g., Italian, French, ranch, bleu cheese, etc.)."), 7:42-65 ("The menu configuration application is launched by clicking on the appropriate icon on the desktop display screen. FIG. 1 will then be displayed. There are three windows on the screen shown in FIG. 1. The left window is the menu tree 7, also called the tree view. The top right window is the Modifiers window 8 and the bottom right window is the Sub-Modifiers window 9. The Sub-Modifiers window lists the sub-modifiers that correspond to the modifier that is selected. The views on the right are referred to as list views. There

are several ways of invoking a command, including using the menu options; using the context menu (right mouse click); using the keyboard or using the toolbar icons. For example, if it is desired to add a category to the menu, the following four options are available: (1) clicking on Edit, Add Category; (2) right mouse clicking on Menu, then clicking on Add Category; (3) highlighting Menu, then typing Ctrl+T or (4) clicking on the Add Category icon on the toolbar. To add an item to a category, the following options are available: (1) highlighting the category to which it is desired to add an item and then clicking on Edit>Add Item; (2) right mouse clicking on the desired category and then clicking on Add Item; (3) highlighting the desired category, then typing Ctrl+N or (4) clicking on the Add icon on the toolbar."), 9:62-10:9 ("When the menu has been completely configured, it can be previewed on a POS emulator on the desktop to verify that the menu is correctly configured before downloading it to the handheld device. To preview, File>Preview Database is clicked on or the Preview Database icon from the toolbar is clicked on. The handheld POS emulator on the desktop can then be run. If the configuration is deemed acceptable, the handheld device is connected to the desktop PC to ensure that a connection has been established; the POS application on the handheld device is exited and File>Download Database is clicked on or the Download Database icon from the toolbar is clicked on. If there is an existing menu database on the handheld device, the system will ask if the existing database should be replaced. Yes is clicked if existing database replacement is desired."), 10:58-11:12 ("As stated, the preferred embodiment of the present invention includes the use of and compatibility with GUI technology. A drag-and-drop approach is used for organizing the tree structure 2 in the generated menu. Drag-and-drop is also used for assigning modifiers (modifiers can be

dragged from the modifiers window 5 and dropped onto the menu item 4 for assignment). In-cell editing results in fast editing of items in building the menus. Customizable fonts enable users to change font types, style and size. Customizable layouts enable users to resize windows, change icons and display preferences. The inventive approach provides for fully persistent storage between sessions, even if a session is improperly or abruptly terminated. Font and the tree state (i.e., which nodes are expanded/collapsed) are stored between sessions. Layout for modifiers and sub-modifiers list views (filter, columns, formatting, font, etc.) are stored between sessions. The last database used is likewise stored between sessions. Splitter views allow the user to see different views at the same time. Each view is displayed on its own section of the screen. Views can be resized via the keyboard or a mouse by simply dragging the splitter in the middle."), 11:13-30 ("An automated function is provided to import existing POS databases into the inventive menu generation system and, as discussed above with respect to the detailed example of how to use the preferred embodiment, an automated download procedure is provided to transfer the desktop database onto a handheld device and/or Web page. Also as discussed, the preferred embodiment facilitates preview of the handheld device or Web page version of the POS menu on the desktop before downloading and configuration. Customizable desktop menu generation is contemplated, as discussed above, in the form of customizable fonts, columns, layouts, etc. The inventive approach also includes templates for common modifiers that can be assigned to similar menu items. The preferred embodiment also supports multiple databases, thus providing for the creation and storing of different menu databases on handheld devices such as breakfast, lunch or dinner menus. The user can then select the appropriate database to reflect the time of day."),

11:51-12:4 (“Advanced database functions are provided in the preferred embodiment of the invention, including an automated download process onto handheld devices and/or Web sites. In the preferred embodiment, the menu generation system of the present invention uses an API called ActiveX Data Objects (‘ADO’) for database access. ADO is useful in a variety of settings. It is built on top of OLE DB and can be used to talk to databases and, in the future, any data source with any OLE DB driver. Advanced querying is supported. The database can be queried on virtually all fields. Queries can be built using SQL syntax for experienced users or can be created using a query builder which guides users through the creating process. Advanced error handling is supported. Errors occurring at run time can be trapped. A descriptive message is displayed to alert the user and provide error information. However, the application does not terminate when the errors happen. The source code is easy to maintain and modify, thus allowing for on time delivery of customized versions of the software. The advanced database functions produce well-designed databases that accommodate growth and scalability.”), 15:1-9 (“While the preferred embodiment of the invention includes the selection of items from a master menu wherein the master menu is displayed using a graphical user interface, it is to be appreciated that any means for displaying the master menu to the user and generating another menu in response to and comprised of the selections made is encompassed by the contemplated invention. The invention encompasses the selection of nontextual symbols, characters, icons and the like, in addition to text, from a hierarchical tree menu or the like for generation of another menu comprised of such items.”), FIG. 1, FIG. 2.

- Brown at [0026] (“A merchant may add a terminal by clicking a button that labeled, ‘add terminal’. When the add terminal button is clicked, the terminal information is displayed and the merchant inputs the terminal information into the system. The merchant may designate the terminal as an internet terminal by clicking on the button labeled, ‘internet.’ The merchant now has a terminal for accepting payments over the internet. For example, if a merchant wishes to accept Visa and Master Card, the merchant would acquire a merchant account from a merchant account provider, add a terminal by clicking the ‘add terminal’ button and input the numbers provided by the merchant banking provider into the terminal fields. The merchant may define a method of payment from a list that contains all the methods of payment that UTOPA.com accepts, for example, cash, Visa, Master Card, Discover, American Express, PayPal, etc. The payment options that a merchant may accept is listed as options that appear at the checkout page on UTOPA.com, the external website and the point of sale in the merchant's retail store. All three sales avenues will therefore share the same methods of payment.”), [0037] (“Item and item quantity management. The point of sale engine manages how an item appears on the online store and on the retail store's point of sale system, and controls item price. When a new item is added to inventory, a plurality of parameters about the item are available for input. The merchant has the option to be detailed and input many parameters or input few parameters. The minimum requirement for an item is a description and SKU number.”), [0042] (“Once an item is created, an inventory number is assigned. A description field is displayed, whereby any text description may be entered. Further optional information may be input, which may include the location of the item, such as showroom or retail store, or the age and material of the item. Further information may

include date or origin; quantity in stock; the merchant's vendor; maker of the item; dimensions and weight; shipping options; pricing, such as percent to mark up or exact price; where to sell, such as in store, to dealers, on UTOPA.com, or on the merchant's website; costing method such as first-in-first-out (FIFO), or last-in-last-out (LIFO); and vendor internal code. The optional information may include the condition of the item such as mint, excellent, good, fair, or poor. The merchant can manage each list of options for each item."), [0043] ("The inventory feature of the system provides several functions to the merchant at the item input interface. The inventory system communicates with the purchase order system to provide an "allow auto order" function. For each item entered into the system, a purchase order is generated. The purchase order (PO) has a line item associated therewith representing the item. An item cannot be listed without a way to track it. When an item is initially entered, the quantity is zero by default, but can be specified by the merchant. The amount paid for the item by the merchant may also be specified."), [0048] ("Many of the display boxes in the system are collapsible, so that a merchant can ignore a set of data and functions if it does not plan to use them. Also, for ease of use, the system can be set to remember the particular sets of boxes that are collapsed by clicking 'save screen layout.'"), [0059] ("Each item may have as many as five images associated. The merchant may upload an image on to the system whereby the system will automatically resize the image, and adjust the image quality to achieve a maximum of about a three second load time for a customer to load an image of the item in the customer's web browser. A page that requires more than three seconds to load typically causes a potential customer to lose interest in the item. The system will automatically adjust the image to have about 500 pixels, while maintaining the aspect

ratio, and maximizing the bit depth for the best resolution possible."), [0061] ("The merchant may also display the (DSA) to a customer when viewing an item. The merchant may specify up to three lines of text for the DSA to communicate verbally to the customer when viewing the item. The DSA will appear with the item on UTOPA.com When the merchant clicks, 'preview,' the system displays the DSA and accompanying monolog customers will see and hear, when they view the item. Studies have shown that when a human face speaks about an item to a customer, there is a greater chance of selling the item."), [0070] ("All screens in the POS are designed for use by either a keyboard and mouse or by a touch-screen. The main point of sale screen provides information such as the software version, on-line status, location of register and/or register identification, transaction, date, time, and the user who is currently logged in."), [0073] ("The system further includes a website builder and manager. Referring to FIGS. 51-56, the website manager provides the merchant with options to display inventory on the merchant's website. For example, the system provides background color; font color; link color; font size; container height; font; border color; number of items to display; and container width; When the merchant specifies his desired options, he may click 'generate code,' whereby the website builder will generate html code and instructions on where to paste the code. The system generates code pertaining to inventory, login page, shopping bag, and 'my account.' The system also provides suggestions and templates for the merchant's website."), [0074] ("The inventory and point of sale system is completely based through the internet and is accessible through an ordinary web browser. The system is capable of handling a plurality of independent merchants, whereby each merchant may access the system through the internet with a web browser.").

- Jones at Abstract ("A master copy of a menu is stored in the server, and changes to the master copy are automatically made to local copies stored in the hand-held and touchscreen units. Thus, all copies of the menu, in all of the units, are quickly and automatically synchronized. The system allows a manager to indicate the quantity remaining of a particular item, and that quantity is updated, in all of the hand-held and touchscreen units, as orders for that item are placed."), [0054] ("1. The present invention enables all of the wireless hand-held units 1, and all of the stationary touchscreen units 3, to be linked to a central database, such as database 8 associated with server 5, the database containing a menu of items available at the restaurant; 2. The system provides means for quickly and continuously updating the status of the menu, such that all units connected to the system will display current and updated information to waiters and waitresses; and 3. The system provides means for alerting users of the system, including all waiters and waitresses, concerning the quantity remaining of selected menu items."), [0058] ("The invention also enables a manager to make permanent changes to the restaurant menu."), [0059] ("FIG. 2 illustrates part of the programming of the system, in one preferred embodiment. FIG. 2 represents the software that permanently modifies the content of the restaurant menu. In block 20, the restaurant menu is modified in the server database. This modification, which could be the permanent addition or deletion of a menu item, is accomplished by modifying database 8, using the server 5. This modification would normally be made by a manager, at the server location. In the preferred embodiment, this type of modification cannot be made through the hand-held units 1 or the stationary touchscreen units 3."), [0068] ("The menu modifications contemplated by FIG. 2 include permanently adding or deleting menu items. FIG. 3 deals with temporary

changes, and shows how a manager can “enable” or “disable” a menu item. As used in this specification, the term “disabling” means temporarily removing an item from the menu. The most common reason for disabling an item would be that the supply of that item is exhausted. The term “deleting” means permanently removing the item from the menu. As shown in more detail below, one difference between “disabling” or “deleting” an item is that a disabled item still appears on the screen displays of the various hand-held units, and the stationary touchscreen units, but with an indication that the item is not available. A deleted item does not appear in any form.”), [0069] (“The flow chart of FIG. 3 is generally similar to that of FIG. 2. However, in the preferred embodiment, the enabling and disabling of an item, represented by FIG. 3, is performed through one of the stationary touchscreen units 3, and not through the server 5.”), [0083] (“FIG. 5 represents a typical screen display, generated by the “Manager Helper” software that is loaded on the server 5. The manager uses the Manager Helper software module to make permanent changes to the restaurant menu. This software may be accessed at the location of the server 5, or, more preferably, at any of the various stationary touchscreen units 3.”), [0084] (“The display shown in FIG. 5 shows that the manager is viewing the “Dinner Menu”, and the category “Main Course”, the latter fields having been highlighted on the left-hand side. The menu items pertaining to main courses on the dinner menu are shown on the right-hand side. Each item is shown adjacent to a column of “forced modifiers”. A forced modifier is a property of the menu item that the waiter is obligated to enter. For example, “London Broil” is associated with the forced modifier “Temp” (for “temperature”), indicating that the waiter must specify whether the food is to be “well done”, “rare”, etc. A “regular modifier” is a property of the menu item which is optional,

and which the waiter may enter, as part of an order, at the request of a customer. The manager can click on the indicated tabs to view and edit the regular and forced modifiers."), [0085] ("To add an item to the menu permanently, the manager presses or clicks on the "Add" button, which is encircled in FIG. 5. The system responds with a second screen, shown in FIG. 6, onto which the manager can enter the new menu item, using a keyboard associated with the server. In the example shown, the item to be added is "Salmon". After entering the details associated with this item, the manager presses the "Add" button, shown encircled in FIG. 6. The result is shown in FIG. 7, wherein "Salmon" now appears as one of the menu items on the dinner menu."), [0092] ("FIG. 13 depicts a typical display that appears on one of the stationary touchscreen units 3. This display, titled the "Manager Screen", allows the manager to access any item on the menu, and to enable or disable that item. In the example given, the manager has chosen "Main Course" from among the buttons at the upper right-hand side, and the corresponding menu items appear as buttons at the lower right-hand side. To disable an item, the manager selects the item, and presses the button labeled "86 Item". In the restaurant trade, the term "86" is typically used to indicate disabling of a menu item, and the term "68" is used to indicate restoration of an item. When the manager presses a button to enable or disable an item, a message appears, confirming that the status of the item in the system has changed."), [0093] ("In the example represented in FIGS. 13-15, the manager wishes to disable "Calves' Liver" from the menu. As shown in FIG. 13, the manager highlights this menu item, and the left-hand side indicates the status of "Calves' Liver". Since calves' liver was originally available, FIG. 11 shows the "86 Item" button as available, and shows that the "68 Item" button is grayed out (i.e. unavailable for use).

When the manager presses the “86 Item” button, a message appears, confirming that the status of the menu item has changed, as shown in FIG. 14.”), [0094] (“In FIG. 15, the status of calves' liver is now indicated as unavailable, and the “86 Item” button is grayed out, while the “68 Item button” is available for use. Thus, FIG. 15 shows that the manager now has the option of restoring calves' liver to the menu.”), [0098] (“The manager places an item on count in the following manner. First, the manager uses a main menu screen (not shown), generated by the software, to reach the screen labeled “Put Item on Count”, as shown in FIG. 20. The menu appears on the right-hand side, possibly divided by category. Thus, by clicking on “Main Course” in the upper right-hand screen, the manager sees a display of main courses in the lower right-hand screen.”), [0099] (“The manager then browses the menu, as displayed, and selects the item to be placed on count. The manager enters the value of the count, using the numeric keypad that appears at the left-hand side of the screen, and presses the “Set Count” button. The manager then receives a confirmation (not shown) that the item has been placed on count.”).

- Flashpoint
    - Flashpoint Operations Guide, Menu Setup Manual, 3-90.
    - Flashpoint News (Jan. 18, 2007) (“Flashpoint version 14 is released. Features include a complete Enterprise solution along with above store reporting, centralized menu setups, system setup, and centralized employee management.”).
    - Flashpoint Operations Guide, System Setup Manual, 4-140
- NextPOS-Aldelo
    - Aldelo for Restaurants Training Manual, Chapter 7 – Information Data Entry, 45-60 (“Menu Setup”)

- Aldelo for Restaurants User Manual, 54-72 (“Menu Setup”)
- Aldelo NextPOS Software Manual, 101-111 (“Restaurant Menu Setup”)
- Aldelo NextPOS Product Overview, https://web.archive.org/web/20030228034043/http://bcdata.com/ (Feb. 28, 2003) (“With 100% touch screen operated point of sales, super easy to follow screen layouts, and the simplicity of Windows based interface, your wait staff will be able to use the software in a matter of minutes, rather than days!”)
- NextPOS Corporation and Transactional Web, Inc. Partner to Replace Expensive and Labor Intensive Reporting and Accounting for Restaurants with Affordable Subscription-Priced Web Automation, https://www.prweb.com/releases/2004/11/prweb183924.htm#! (Nov. 2004) (“NextPOS for Restaurants software was designed with the independent restaurant owner in mind. The design concept was to create a software that would be easy to setup and use by even a computer novice. Although the software is easy to use, it has all the features of the more expensive Point of Sale software programs on the market with a price that even the smallest restaurant owner can afford.”)
- Aldelo for Restaurants, https://web.archive.org/web/20060521002319/http://www.aldelo.com/ProductsandSolutions/AldeloForRestaurants/tabid/67/Default.aspx (May 20, 2006) (“Aldelo For Restaurants is full of benefits. It is a user friendly Restaurant POS software that does not require you to know about computers to use it.”; “Menu Recipe Control & Complete Inventory Management”)

- PointofSuccess
    - Restaurant Software for POS Systems, https://web.archive.org/web/20060104044810/http:/www.pointofsuccess.com/restaurant-pos-software.htm (Jan. 4, 2006) (“Customize ordering menus to fit your way of doing business . . . Accounting and tax reporting information is retrieved from the manager's workstation with a few simple commands.”).
    - Order Entry and Preparation, https://web.archive.org/web/20051222105812/http:/www.pointofsuccess.com/orderentry.htm (Dec. 22, 2005) (“Define Products and Order Entry Menus. Some other point of sale systems' menus are so difficult to design that the job requires a technician. With Point of Success, menu design is as easy as placing buttons on the screen and setting the product code they activate. Menu design is even easier with the available training CD. When menus are scheduled to change, the changes can be made and stored in an inactive area of the menu designer, then added to the active menu with a few mouse clicks. Temporary or seasonal menu changes are almost automatic! Your menu is easy to set up in Point of Success. Select the products to appear on the menu and organize the menu to fit your way of doing business. The only limit to the number of buttons on a screen is the size of the computer display. Buttons can be sized, colored and located anywhere on the screen.”).
    - Point of Success, https://web.archive.org/web/20060101024912/http:/pointofsuccess.com/ (Jan. 1,

2006) (“Point of Success software is made to be set up and used by a restaurant owner or manager”)

- Feature Comparison for Point of Success Standard and Point of Success Premium, https://web.archive.org/web/20060104041129/http:/www.pointofsuccess.com/comparison.htm (Jan. 4, 2006) (“Change order entry background screens to customize image for a restaurant . . . Customize the order entry menu to fit your restaurant”).

4. Remote Web Server (“Remote Web Server References”)

The Asserted Patents purport to provide a solution to the long-recognized problem of managing business operations from a location remote from the store or business, and particularly, from anywhere in the world. *E.g.*, ’640 patent, 2:6-8. The Asserted Patents state that the alleged invention “allows remote monitoring of screen designs from anywhere via the web.” ’640 patent, 5:54-55.

The benefits of a remote web server were known long before the Asserted Patents. Implementing a POS system using a remote web server allows for the management of multiple stores or businesses from one central location, such as a corporation’s headquarters. Additionally, using a remote web server to manage multiple stores or businesses allows for uniformity across the multiple locations. Implementing a system using a remote web server would also reduce hardware costs and simplify management of multiple stores or businesses, because it can eliminate the need to place a server at each location and avoid having to separately manage and configure each POS system at each location. Making the remote server a “web” server in a system with thin clients was also desired because HTML is a standard protocol and avoids the requirements of special hardware, software, or training. Furthermore, a web server

would allow the POS terminals to be implemented with a wide variety of low-cost standardized devices that can support a web browser.

Remote web servers were ubiquitous in the prior art, for example as described in the following ("Remote Web Server References"):

- Woycik at [0071] ("FIG. 1 shows an exemplary self-service ordering system 10 constructed in accordance with the invention as it might be used by a chain of quick-service restaurants. The system 10 includes two store locations 12, 14 each having a plurality of kiosks arranged in a client-server configuration. In each store, the local (on-site) server 16 is connected to the Internet 18 which allows remote access by the restaurant chain operator 20 and enables the local server 16 to access a central server 22 for software and media updates. This Internet connection also makes possible use of the local server 16 for receiving online customer orders from customers 24. In general, a single store location will include at least one local server 16 and at least one client kiosk 30. In a more typical configuration such as is shown for Store 1 (store location 12), the store location will include a single server 16, multiple client kiosks 30, a cash register 32 (for cash, credit card, debit card, bar code card, or other payment means), and possibly a kitchen build screen or printer 34 for use by the food preparers in making and/or aggregating the food items for a particular order. The store location might also include one or more remote clients 36 (e.g., for ordering outside the store, but in a nearby location such as a drive-through kiosk, or at a kiosk located elsewhere in the same mall in which the store is located)."), [0075] ("Remote access also allows the chain operator 20 to remotely run the administrative tool. This can be done several ways. One is for the chain operator 20 to have the administrative

tool loaded on a remote computer. For this, the remote computer can access the configuration information (menu screens, items, and other settings) at the local server 16, then allow the operator to make changes, and then update the local server 16 with the new configuration information. A second approach is to have the administrative tool loaded on the central server 22 and then provide the chain operator 20 with web access to the central server 22. In this approach, the central server 22 then accesses and stores updated configuration information on the local server 16. A third approach is to provide a web interface to the administrative tool on the local server 16 so that the chain operator 20 can access this interface from any Internet connected general purpose computer using the IP address of the local server 16. This can be done using any of a number of different remote access and control approaches known to those skilled in the art. Other means of providing remote administration of the kiosks can be utilized."), [0080] ("[T]he web services platform provided by .NET can be used to provide remote administration by the chain operator from any Internet-connected computer (such as a home office computer) so that various store locations can be configured from a single computer. The programming needed to implement this software architecture strategy is known to those skilled in the art."); [0093] ("The system also includes remote web service to allow customers to place orders using a standard web browser via the Internet. The remote ordering can be handled either using the central server or the local server kiosk which includes a web server."), [0121] Apart from using the Internet for online ordering and centralized storage and distribution of menu configurations, media, and software, it can also be used by the administrators to access the administrative tools remotely using a standard web

browser, rather than having to use a kiosk. The web server on the local kiosk server includes either a web-enabled version of the administrative tool or includes a separate web services process that provides a web interface into the server process that allows configuration of the self-order application in the same way as the administrative tool. Like the web client, the web enabled administrative tool reduces overhead by using vector graphics. The web enabled administrative tool also enables store owners to update multiple servers from a single location, for example, by logging in and updating the local servers either one at a time or as a group. This feature can be used to update any of the items noted above-menu configurations, media updates, templates, software enhancements and patches."), [0122] The remote server can be used for store owners, franchisees, etc. to update multiple local servers, such as at multiple store locations, from one central location. By tracking the IP addresses or names of each local server registered with the remote server, all local servers can have updates happen from the central location, eliminating the need for administrators to make the changes at each local server individually."), [0128] Additional features of the remote server are to provide a central data repository for multiple local servers and provide a reporting module. The centralized data repository stores local transaction data from kiosks as well as customer data for use with customer recognition and recommendations. The reporting module combines data from client orders and remote client orders for consolidated reporting and graphing. As seen in FIG. 28, customer data and/or order data is stored on the local server when a customer places an order through a client or remote client (blocks 590, 592). The local server connects to the remote server periodically and stores the customer and

order data in the centralized data repository (blocks 594, 596). Administrators request data or a report from the remote server that then supplies the data or report requested (blocks 598, 600). The consolidated reporting allows a store owner or corporation with multiple locations to run reports across multiple locations to show popular products, store performance comparisons, trends over time, etc. The reporting module also contains an export function to export raw data for analysis in third party tools.")

- Manno at [0015] ("In accordance with an aspect of this invention, a web-based point-of-sale system, or WPOS system, is provided in which a computer at the retail store is programmed and connected with a computer network, i.e., a LAN, a WAN, and/or the Internet, to create a fully functional WPOS that allows general purpose small computers, i.e., personal computers, to connect to the store interface and to serve as point-of-sale stations, and also permits the retail business manager to oversee sales, inventory, and other retail operations over the Internet."), ("In any of several embodiments, a network based point-of-sale system employs at least one point-of-sale terminal based on a small computer. The computer including the usual means for entering and processing data, means for connecting to said network, one or more ports to which a POS device (i.e., point-of-sale peripheral device) may be connected, and basic software including an operating system and a web browser."), [0017] ("In a multiple-store retail environment, a point-of-sale system can involve a main central computer, i.e., a "Mainframe", a network permitting communication between devices on the network, and also at each store location (i.e., a LAN, a WAN, or the Internet), and at least one point-of-sale terminal at each store location, where the POS terminal employs a computer as described earlier, with a web browser; and one or more

associated POS devices to be operated by the computer in connection with a sales transaction of merchandise or service."), [0046] ("An off-site management facility 54 permits store management personnel to monitor store activity and obtain management reports from the store server(s) 12 and/or from the mainframe 40. Here, a management laptop computer 56 is coupled through the Internet 18 to the system 10, or alternatively, the management functions can be carried out from the remote site via a web-enabled cellular phone or PDA 58, or from another computer 60, each being coupled over the Internet to the system 10. The facility 54 can be at any location, and the computer devices 56, 58, 60 can be wired or wireless. This allows management to check inventory levels, sales activity, and other store activity, as well as historical sales data, customer demographics, or other valuable trade information, from any remote location. Access is provided through the web browser of the computer 56 or other device, without requiring any additional or special software to be installed at the remote location. All necessary software is provided from the server through the web browser. A security functionality requires special access codes and PIN numbers for management level access."), [0051] ("A management functionality for a multiple-store configuration is illustrated in FIG. 4. Here, as discussed in respect to FIG. 1, there is a Store 1 server 12, a Store 2 server 12, and a headquarters office mainframe computer 40, which connect with one another over the Internet 18, together with local network or networks 14 (not shown here). At initial installation of the store servers 12, each store registers with the headquarters main frame computer 40. Then, once the store servers are registered, the information stored on each of the servers 12 is available on the system for management purposes. At a management location 54,

which can be anywhere where there is access to the Internet 18 (or to a LAN at either store location or at the headquarters location), an authorized management employee or officer can access the WPOS system using a standard Internet or web-enabled device such as laptop computer 56, PDA or cell phone 58, or desktop computer 60. As with other implementations previously discussed, the computer device accesses the WPOS system through the computer's web browser. The management employee or officer is provided with a higher level of access, via an access code, PIN, or the like. At the management location, it is possible to view (and print or save) daily or other periodic management reports, as well as to obtain specific management related information. The management employee or officer can write and send email messages to staff or customer, or receive same; to create or alter web pages on the store web sites; to obtain sales and demographic information, i.e., 'data mining'; or to view any other commercial data for either store location in real time.")

- Tengler at [0073] ("The manager station 70 can use a broadband data line 72 to link the manager to the Internet. Referring to FIGS. 7A and 7B, managers can view information about the multiple restaurants 2 using interfaces 192 and 196 over the Internet. These managers can also modify menus and change prices using interfaces 192 and 196."), [0109] ("Managers access information about the restaurant 2 using the web server 620 and interfaces 192 and 196. When a manager attempts to connect to the web server 620 from off-site (such as the manager's home or a central corporation office), the connection requires an Internet protocol (IP) address. The IP address associated with the restaurant 2 can dynamically change for a number of reasons. Some restaurants 2 can use DSL (Digital Subscriber Line) for an internet

connection. For restaurants 2 using a DSL connection, Internet Service Providers (ISP) have different rules for when IP addresses are assigned. A restaurant 2 may not have an "always on" Internet connection because of the cost or availability in its locale, and the store must dial in to the ISP in order to have connectivity and to be assigned an IP address. Thus, an IP address is required to connect to the restaurant 2, but the IP address may not be known and the restaurant 2 may not even be connected to the Internet.")

- Cusack at [0014] ("As such, the present invention achieves the above stated and other objects to deliver the aforementioned benefits by providing new and improved systems and methods for facilitating real-time, web based point of sale operations such as sales, reporting, etc. The system and method include and involve a central database management facility storing and managing data related to at least one user interface form operable within a browser application and items which may be sold or otherwise processed at a point of sale. Also included and involved is a plurality of client data processing facilities which are coupled to the central database management facility via an electronic data network. Each client data processing system is located at a remote point of sale and is coupled to the central database management facility via a network connection over the electronic data network. Each client data processing facility is configured to execute the browser application, to access the central database management facility via the network connection to retrieve and process at least one user interface form within the browser application, to access the central database management facility in real time to retrieve data related to at least one of the items to facilitate a remote point of sale related operation, and to cause the

central database management facility to manage the databased on the remote point of sale related operation."), [0028] ("Referring now to FIG. 1B, depicted therein is a POS system which facilitates real-time, web based POS operations according to a preferred embodiment of the present invention. POS system 150 is a web based POS system that performs all the functions of a present day system with the addition of an enterprise wide solution to the aforementioned problems associated with the prior art as discussed above with reference to FIG. 1A. POS system 150 may include a plurality of remotely located retail establishments (possibly among disparate retail vendors) including multiple facility locations shown as store 1 and store 2 (phantom lines). Each remotely located retail establishment includes at least one POS terminal 162 (denoted in FIG. 1B as POSX where X is the subscript that indicates any number of POS terminals) and a print server 170. Each POS terminal is capable of running an entire POS system in accordance with the present invention."), [0029] ("POS system 150 further includes an applicationserver 152 and a central database 154 which can be integrated with back office systems 156. Application server 152 can be any well know Web server that is coupled to a network 175 (e.g., a globally accessible network such as the Internet and World Wide Web (WWW), a network incorporating open-standards based technologies and protocols, etc.) and is configured to serve and process content such as web pages in accordance with the present invention."), [0030] ("Application server 152 is coupled to central data base 154 and is configured so that systems and processes running with and on application server 152 can make real-time inserts, updates, queries and deletions, and per form other database operations in relation to data Stored in central database 154. For example, such real-time database

operations may include real-time price changes related to particular items about which POS transactions may relate (e.g., sale transactions, etc.). In particular, the present invention and, in particular, system 150 now permits real-time changes to pricing data say for a large drink at perceived peak times during a business period to maximize revenue or sales, etc. In particular, the present invention now permits a user (e.g., a regional store manager associated with a multiple facility enterprise) to access a web connection and change pricing data, quantity and availability data, etc. to more effectively meet changing market conditions to enhance sales, increase revenues, etc. And, a price or other datum may be reduced to aid slow moving inventory based on real-time transaction data analysis. The present invention now provides real-time control over what may be a global enterprise via a common and well know user medium-the Internet and WWW."), [0032] ("To achieve such functionality, application server 152 hosts at least one web page that forms a user interface to the POS system (described below with reference to FIGS. 3A-3D). Such web pages provide in software all the various functions expected in a POS system (e.g., item selection, pricing, sub and total computations, etc.). Such an application server arrangement is well known, with the front end of a system being stored and served on an application server coupled electronically (via common protocols, for example, SQLNET) to the database, or backend, of the system. One having ordinary skill in the art will immediately understand the configuration described herein."), [0033] ("Each POS terminal 162 can provide standard POS functionality, such as touch screens, cash drawers, etc., and is configured to execute a common web browser, access the global network 175 (e.g., via modem or another network interface, etc.) and

download the web pages that make the system's user interface, hosted by application server 152. Accordingly, a store can arrange its POS terminals 162 on a Local Area Network (LAN), as stand alone PC's, dedicated type POS terminals. Such as those manufactured and marketed by JAVELIN (VIPER machines including touch screen entry devices), etc. attached to network 175 via a modem or other common network connection, or in any other well known arrangement such that each POS terminal is provided Internet network access. For example, POS6 is shown directly accessing an ISP and POS5 is shown directly accessing network 175. Each POS terminal 162 may be stationed throughout, such as throughout a retail restaurant establishment, and may be placed anywhere that is accessible to network 175.")

- Olson at Fig 1, [0012] ("The objects of this invention are achieved by point of sale system with a web-based back-office system comprised of one or more point of sale (POS) terminals and one or more web servers. The POS terminals and web servers communicate via commonly used, standardized internet protocols. The POS terminals are implemented with personal computer hardware and software using one or more relational databases. The web server is implemented with web server hardware and software, also using one or more relational databases."), [0017] ("FIG. 1 shows a high level diagram of this invention. Each point of sale (POS) terminal such as POS 21 represents a single cash register in a store, retail location, or business location. POS 21 is in STORE 1 and POS 2 (22) is in store 2. Each POS includes personal computer hardware and software. Additional POS terminals beyond the two shown, as well as additional stores beyond the two shown, are within the scope of the invention. Each POS normally operates with a hardware/software connection 23 to the Internet or

Web. However, if the web goes down, the POS terminal continues to operate. There is a loose coupling of the POS to the back office (BO). The POS to BO connection is not required for the basic business functions of the POS. All transaction data is stored in a relational database on the hard drive in the POS."), [0019] ("As transactions are created at a POS a log entry for the newest transaction is also created, this log entry is used to flag if the transaction has been uploaded to the web server. Part of the POS application, the BO interface is continuously running in the background. This component reads the log of transactions. If a transaction needs to be sent, it tries to send it. If the send fails (for example, if the connection to, or the Internet itself, is down), it goes to sleep and tries again later. Additionally, the BO interface requests update from the BO such as new items, price changes, employees, etc. The POS terminals communicate via HTTP protocol (hypertext transfer protocol) 23 with Back-office BO software, which is implemented on web servers 24, which can be located anywhere in the world. In addition, the BO software and data can be viewed from any store employee at any PC who has Internet access 26 and a password."), [0026] ("Remote and local web browsers 33 illustrate a method used by store managers, corporate administrators, or system administers to access and view a company & its store's instantaneous performance from minute to minute. The POS, BO, and Web Browsers all communicate via the Web. 34. The protocols between the POS and Web browsers and BO servers is HTTPs, XMURPC. 35. XML-RPC is a remote procedure call protocol which uses XML to encode its calls and HTTP as a transport mechanism. It is a very simple protocol which defines only a handful of data types and commands."), [0027] ("The invention allows remote monitoring of both the

POS and back-office systems from anywhere via the web."), Claim 5 (The point of sale system with a web-based back-office system of claim 1, wherein a real-time status of said POS terminals and said back-office web servers is obtained at remote locations via personal computers with browser based web access."), Claim 10 ("The web-based back-office system of claim 6, wherein a real-time status of said POS terminals and/or said back-office system is obtained at remote locations via personal computers with browser based web access."), Claim 15 (The point of sale POS system of claim 11, wherein a real-time status of said POS terminals and/or said back-office web servers is obtained at remote locations via personal computers with browser based web access."), Claim 20 ("The method of providing a point of sale system with a web-based back-office system of claim 16, wherein a realt ime status of said POS terminals and said back-office web servers is obtained at remote locations via personal computers with browser based web access."), Claim 25 ("The method of providing a web-based back-office system of claim 21, wherein a real-time status of said POS terminals and/or said back-office system is obtained at remote locations via personal computers with browser based web access."), Claim 30 ("The method of providing a point of sale POS system of claim 26, wherein a real-time status of said POS terminals and/or said back-office web servers is obtained at remote locations via personal computers with browser based web access.")

- Lukkari at 2-3 ("Restaurant employees operate the SmartRestaurant service with a laptop equipped with WLAN connectivity and a standard web browser. Laptops with wireless connectivity can be placed freely at the restaurant premises and they also allow the employees be mobile."), 5 ("The restaurant employees access the

SmartRestaurant service via a web browser. After logging in with an employee account the employee is provided with the main menu, which provides access to following functions: configuration of the service, status page of orders (used by chef), listing of all orders (used by cashier) and reports (used by cashier).")

- Blinn (US5897622) at 6:9-25 ("Focusing now on the client 100, the client is a general purpose computer. In a preferred embodiment, the client 100 is a conventional personal computer equipped with an operating system supporting Internet communication protocols, such as Microsoft Windows 95 and Microsoft Windows NT, a browser, such as Microsoft Explorer or Netscape Navigator, to access the Merchant System and a conventional modem for access to the Internet 104. In other embodiments, the client 100 could, for example, be a computer workstation, a local area network of computers, an interactive television, an interactive kiosk, a personal digital assistant, an interactive wireless communications device or the like which can interact with the network. While the operating systems may differ in such systems, they will continue to provide the appropriate communications protocols needed to establish communication links with the network 104.")
- Showghi (US6920431) at 4:24-28 ("FIG. 2 is a diagram of a remote ordering system according to this invention which is based on the public wireless telecommunications and Internet infrastructures using cell phones, two-way pagers, PDAs and the like wireless, hand-held communication devices"); 6:5-22 ("FIG. 2 illustrates a system that is based on the public wireless telecommunications and Internet infrastructures. In this system conventional hand-held wireless communication devices, such as two-way pagers 22 (FIG. 3 a), cell phones 24 (FIG. 3 b) and PDAs 26 (FIG. 3 c) would

also be useful as remote ordering devices. Here, a base transceiver station 28, such as a cell tower, transmits menu information to the wireless device and, receives orders with payment information from the wireless device. These signals would be routed by the telecommunications service provider 30 via the Internet 32 to menu/order server 31 which manages the order taking process and financial transaction software applications. When the order is complete, it is transmitted via the Internet 32 to the venue fulfillment center server 16 and order processing computers 18. Depending on the size of the venue and the number of fulfillment centers within it, there may be one or more order processing computers, and there may or may not be a need for a separate server 16."); claim 2 ("The system of claim 1, where said order-taking computer is an Internet connected server hosting an order-taking application"); FIG. 2;

- Blinn II (US6,058,373) at 7:50-63 ("II. Overview Of The Preferred Electronic Merchandising System This section provides an overview of the preferred electronic merchandising system 100 as illustrated in FIG. 1. The preferred electronic merchandising system 100 is a client/server system which allows merchants to provide a virtual store which processes sales transactions. The electronic merchandising system 100 includes a consumer client 102 (the client) and a merchant system 104 which contains at least one store server process 106 (the server). The consumer client 102 and store server process 106 are in communication with each other by use of a communications medium 108."); 7:64-8:11 ("The consumer client 102 contains a consumer browser 110. The consumer browser 110 communicates with the store server process 106 and displays the web documents created by the store

server process 106. Each store server process 106 provides a server architecture that supports the presentation and administration of a virtual store. Preferably, the store server process 106 comprises a number of components including a dynamic page generator 120, an action manager 122, one or more orders 124 and an order processing module 126. Furthermore, in communication with the store server process 106 is a storage device such as a hard disk which contains HTML structures 128 which define the layout of different HTML pages. In addition, the store server process 106 is in communication with one or more databases 130 with a database module 132."); FIG. 1

- Costello (WO 01/65427 A1) at Abstract ("A method of conducting a point of sale (POS) transaction includes providing a POS device at the location of the transaction, providing a server at a location remote from the POS device, providing a means of communication between the POS device and the server and downloading a POS program from the server to POS device. The POS program controls the operation of the POS device to conduct a sale. The POS device then receives details of the transaction transmitting details of the POS transaction from POS device to the server. The details of the POS transaction are then stored in a database within the server. A point of sale (POS) system, includes a plurality of POS devices for conducting POS transactions and a server."); 2:8-28 ("According to a first aspect of the present invention there is provided a method of conducting a point of sale (POS) transaction, including; providing a POS device at the location of the transaction; providing a server at a location remote from the POS device; providing a means of communication between the POS device and the server; providing a POS program for

the POS device, the POS program controlling the operation of the POS device to conduct a sale; the POS device receiving details of the transaction; transmitting details of the POS transaction from POS device to the server; and storing the details of the POS transaction in a database within the server. According to a second aspect of the present invention there is a point of sale (POS) system, including; a plurality of POS devices for conducting POS transactions, the POS devices configured to be loaded with a POS program for controlling the operation of the POS device to conduct the POS transaction and provide the server with details of the POS transactions; and a server arranged to communicate with the POS devices, the server configured to store the details of each POS transaction in a database storage means. Preferably the server downloads the POS device with updates to the POS program"); 5:16-24 ("Referring to Figure 1, there is shown a point of sale (POS) system 10 which includes a server 12, a plurality of POS devices 14, 15A to 15B and a virtual private network 16 facilitating communication between each of the POS devices 14, 15A to 15B and the server 12. The server 12 is provided with a firewall 18 to provide secure access to the server and protect data held within. The POS device 14 is a "thin client" of the server 12, that runs a virtual environment, such as a Java virtual machine that allows uploaded Java language programs to be executed within the virtual machine environment. The server 12 provides an Accounting Backend 15 accessible via a browser running on the POS device 14 or an external PC or laptop."); 5:26-6:9 ("Referring to Figure 2, the POS device 14 is a terminal 20 located at the point the sale is made. It may be loaded with two main components, a point of sale client component and a back office access component. Each of these components is a Java

program downloaded from the server 12 through the virtual private network 16 to the POS device 14 and executed within the Java virtual machine. The POS device 14 can be provided with special facilities to cater for a specific industry the POS device 14 is used within. These include input and output means 24. This may be for example, a screen, a keyboard, a weight measuring scale, a bar code reader, a touch screen, a receipt printer, a bar code scanner or a mouse. The POS device I/O communicates with the substantive POS program 28 using a Java Application Programming Interface (API) called "JavaComm". A smart card device 26 or an EFTPOS terminal may also be included. The smart card device 26 communicates with the POS program 28 using API's called "Opencard" and "SwingGUI". The EFTPOS facility is provided by an appropriate card reader and proprietary Java software for interfacing the card reader with the POS program 28"); 7:1-7 ("The POS program 28 uses a "Store and Forward Messaging" API 36, to communicate with the server 12 using Java Message Service (JMS) protocol and Extensible Markup Language (XML) over the virtual private network 16. This API receives and transmits messages and XML from the server 12. An "Inventory and User Changes Input" API 38 also communicates with the server 12 over the virtual private network by JMS/XML. This API receives reference data including inventory prices from the server 12."); 7:19-30 ("Referring to Figure 1, the backend system is provided within the server 12. This includes a "POS Backend" 13 and an "Accounting Backend" 15. The POS Backend 13 provides business logic, data access, authentication, configuration and gateway services to the "client" POS device 14. It also provides the web browser client user interface via standard web page technology. The "Accounting Backend" 15 provides business

logic, web client interface, data access and gateway services to implement an accounting system. The Accounting Backend also provides capabilities for managing multi-site customers, managing the data and reporting requirements for tax compliance, and a data warehouse that provides storage of information obtained from POS sales in order to provide market data. The data may be provided in a generic form so that privacy issues can be overcome. The backend access is via either the POS device 14 using a web browser client 22 or via a computer running a standard web browser."); 13:23-29 ("The transfer of data between a point of sale device 14 and the server 12 is shown undertaking a POS program update, an update of reference data and recording of a POS transaction on the server. The POS device 14 contacts the server 12. In reply, the server 12 transfers updates to the POS program to the POS device 14. The POS program then downloads the updated reference data. The POS device 14 acknowledges receipt of updates to the POS program and reference data."); Claim 6 ("A method according to claim 1, wherein the server is provided with a POS backend server that downloads the POS program to the POS device."); FIGs. 1, 3, 4 and 11.

- Thompson (WO 02/39226 A2) at Abstract (A system and method for consummating a financial transaction at a POS (12) based on a secure electronic communication over an active, full-time public network such as the Internet (36). The invention comprises at least one POS (12), at least one network appliance (38), at least one primary server (40), and at least one financial service provider (14), the network appliance (38), primary server (40), and financial service provider (14) being in electronic communication over an active full-time public network (36), whereby a financial

transaction is consummated at the POS (12) based on the electronic communications over the network (36).”); 12 (“In the Internet paradigm, the present invention is preferably implemented using a computer program that is operative at a web server such as the primary server 40, which operates a so called "web-site" by known techniques. A flow diagram for such program is attached as Figure 3. An advantage of this embodiment is that all hardware for implementing the invention can be obtained commercially off the shelf.”); 12-13 (“The network server program 48 of the primary server 40 is preferably stored in the internal memory device 44 and executed upon initialization of the primary server 40. It preferably includes appropriate display routines for generating a set of display screens that together comprise the user interface for the site, as will be displayed at the POS 12 via the network appliance 38. To that regard, and as noted above, the preferred embodiment of the invention is preferably implemented within at least one primary server 40. Thus, the invention does not require any modifications to conventional POS 12 hardware or software. Furthermore, by housing the software applications on the primary server 40, the single-POS businesses 12 need not afford the cost of the equipment or software or staff to implement the system functionality that is being described and had previously been reserved for the multi-POS businesses 14.”); *see also* claim 1, FIG. 3.

- Redmond (WO 03/065178 A2) at [20] (“Fig. 2 illustrates an exemplary embodiment of the physical components of the POS Portal. The POS Portal can be conceptually categorized into three system domains, namely, the POS devices or terminals 20, in-store appliance servers 28 and a services syndication hub 30, each of which will be further described below. It should be understood that the components shown herein

merely illustrate an exemplary system configuration and system interactions for each specific domain. Alternatively, for example, some application services of the POS Portal could be deployed to multiple in-store appliance servers 28 each supporting one or more POS devices 20 for performance purposes or other reasons. Based on the disclosure and teachings provided herein, a person of ordinary skill in the art will know of other ways to implement the POS Portal in accordance with the present invention."); [21] ("The POS devices 20 are located at a merchant or point-of-sale location. Each of the POS devices is configured to deliver one or more of the subscribed services to a merchant's customers and/or employees. In addition, the POS devices 20 are capable of performing a number of functions including, for example, managing PIN entry, signature capturing, content rendering, providing a cash register interface, and functioning as a card reader, etc."); *see also* Claim 1, Claim 25 and FIG. 2.

- Flashpoint
    - FLASHPOINT™ P.O.S. Software, Flashpoint Operations Guide, Menu Setup Manual at 3 ("If you are setting up your menu for the first time follow these few simple steps (If you are using the corporate software you will need to select the location being setup before completing any of the following steps)."), 47 ("Setting Concepts Available and Regions (Corporate Software Only): When using the corporate software with multiple concepts you will need to setup the concepts as well as the regions the concepts are available in, before being able to set buttons to concepts. To setup your concepts, set the concepts available at this location, select the "Set Concepts Available and

Region" button to display the following screen; To add a new concept, select the "Add" button. To delete a concept, highlight the concept to delete and select the "Delete" Button. To edit a concept, highlight the concept and select the "Edit" button. If you wish to change the region that this location belongs to (also setup under system setup -> station configurations) enter the region in the white area beside "Set to Region". Once Add or Edit concept has been selected the following screen will display Enter the name that you wish to call the concept and mark whether or not the concept is available at the location being setup. Setting up prices by the Region (Corporate Software Only): If you are using the corporate software and have different prices by the region, you can use this function to setup the prices. Once you have selected the window to set the pricing on (only items included on the highlighted window will display) and "Region Pricing" is selected, the following screen will display;")

- FLASHPOINT™ P.O.S. Software, Flashpoint Operations Guide, System Setup Manual at 4 ("Once the System Set Up button has been selected the following screen will appear, allowing you to complete many of the setups for Flashpoint. If you are using the corporate software, you must select the location to setup before going into system setup."), 45 ("Set Station Configurations: Once selected, "Set Station Configurations" will display the following screen; allowing you to select the station that you wish to setup. This allows you to complete all station specific setups from one machine."), 47 ("Store # / Region – The store # must be set the same as the store number

of the location, the region is the region # that the store belongs to (if using the corporate software the store must be set to a region, be sure to select the appropriate region as setup in the Corporate Software). IF THE LOCATION ID IS SET INCORRECTLY, REPORTS WILL NOT WORK."), 47 ("Set Corporate Location Access On All Excluding Administrator– If you are using the Corporate software and want to force the employees at this location to only have access to "Menu In Stock Only", and the basic inventory screen, select this. Once turned, on the system will only allow employees that have access to menu setup and inventory, to use the limited screen no matter how they are set. The administrator will still have full rights."), 99 ("Enter the IP address where the Main Port runner is located (see your system administrator to obtain this information). This is the dedicated IP address given to you by your Internet provider. The Main Port number is the port to which the location will be accessing to check for orders; this port number should be the same as the one setup under the "setup Ports" section of the Setup for Corporate Controller. Although you can have all stores go to one Port, we suggest that you do not control more than 10 stores by one Main Port Runner (you can have multiple port runners running on the same corporate machine).").

- Halo
    - NRN, *Juice It Up! moves to Web-based POS,* https://www.nrn.com/print/20670 ("They said Halo's secure architecture will support Juice It Up! management and franchisees in making strategic decisions based on real-time information about different regions, menu items

and store performance, while letting them view reports and change menu items from anywhere at any time using a Web browser.")

- halo, *Halo for Restaurants*, https://web.archive.org/web/20061120125846/http://www.myhalo.com/section.asp?catid=267 (November 20, 2006) ("With Halo for Restaurants, you have the ability to monitor and manage a single store or your entire chain from anywhere you have access to a web browser. Change prices, add menu items, manage your touch screen terminals and drill down to the transaction level of detail in real time.")
- Medina thesis at 3 ("HALO, which Vivonet introduced in 2004, is an ASP (Application Service Provider) based product which only requires a POS terminal connected via Internet to a central computer server maintained by Vivonet. The ASP technology allows the application logic and data to exist on a remotely located computer server, which the application users access over the Internet. In Halo's case, ASP allows multiple outlets of a restaurant chain to share information and lowers the complexity of maintaining the POS system."); 56 ("The ASP technology allows an application code to exist on a remotely located computer server and execute with only a web browser on the local computer."), Fig. 2
- halo, *Updates Return to HomePage Restaurant POS Systems – Halo*, https://web.archive.org/web/20080612142133/http://www.myhalo.com:80/restaurant/restaurant_POS/ (June 12, 2008) ("With the Halo point of sale restaurant system, you have the ability to monitor and manage a single store or

your entire chain from anywhere you have access to a web browser. Change prices, add menu items, manage your touch screen terminals and drill down to the transaction level of detail in real time.")

- Green Sheet, *Look to the light: Retail POS system can replace ECR*, The Green Sheet Online Edition Issue 7:06:01 (June 7, 2007) ("Halo includes a touch-screen terminal connected to the Internet. The system tracks details of every transaction (cash, credit and debit) and sends that information via the Internet to a secure data center off-site. Real-time department, category and product sales performance data are accessible by store owners and managers from a Web browser anywhere, anytime. Even when the Internet connection goes down, the Halo offline manager keeps store operations running. And since the data is housed off-site on the Halo secure server, merchants don't have to worry about losing information in a hard drive crash, theft or natural disaster, such as fire or flood.")
- T-Net News, *Vivonet Introduces Halo Retail POS*, https://www.bctechnology.com/news/2007/4/19/Vivonet-Introduces-Halo-Retail-POS.cfm (Apr. 19, 2007) ("Designed for small and medium retailers that have been using cash registers, Halo is an all-in-one web-based touch-screen point of sale and payments system that protects retailers from security breaches, fulfills payment processing needs, and gives them easy access to their critical business information from anywhere they have access to the web.")

- InfoGenesis
    - William Atkinson, *Point of Service: Now and Later of Restaurant POS* (Apr. 9, 2007) ("Infogenesis hosts Revelation as an application service provider (ASP) solution from a secure data center in California and maintains Internet connectivity with each of the three restaurants, processing all outlet and site transactions in a real-time environment. This approach allows the restaurant to receive the benefits of POS without having to hire a support staff or install server hardware. In addition, the system integrates with those the restaurants have from other vendors.").
- 21Century
    - Hospitality Net, *Ameranth Wireless(TM) Announces Award of Patent for Its Revolutionary 21st Century Restaurant(TM) Products*, https://www.hospitalitynet.org/news/4011916.html (May 16, 2002) ("Keith McNally, Ameranth's principal founder and the lead patent inventor said, 'A principal element of this patent involves the methodology by which hand-held operator displays are synchronized from and with the large displays of fixed, legacy computing systems. The patent covers the use of HTML, XML and similar messaging methodologies to instantly create the displays and GUI on the hand-held's smaller screens – with no custom programming required for each restaurant/hotel or similar application. Another element of the invention, while closely related to the first, addresses the ability for web, wireless and fixed computing systems to be synchronized with each other so that an action on any of the system nodes automatically updates all the other nodes. We

believe that this invention will lead to the widespread adoption of wireless and web based solutions in our markets and that this patent – and others soon to follow – will position us very strongly to retain our leadership position.'")

- Ameranth, Our Products, https://web.archive.org/web/20071210185311/http://www.ameranth.com/product.htm (Dec. 10, 2007) ("21st Century Restaurant™: Eliminate the numerous inefficiencies common to restaurants while increasing profitability and guest satisfaction through an integrated wireless and Internet solution for table service, POS extension and payment processing.")

- Phoenix
  - FireFly Technologies, *Management and Back-Office*, Brochure ("Enjoy Anywhere, Anytime System Access: At work, at home, or far away, stay in touch with your business. Phoenix has a remote access feature that gives you full access to system functionality and reports without tying up a store terminal, without alerting the staff, and without waiting for data to be uploaded to a third-party Web site. All you need is Internet access to place orders, change prices on the fly, do payroll, view reports, manage inventory, create employee schedules, and perform any other task you previously had to be in the restaurant for.")
  - Firefly Technologies, *Put the Power of Phoenix Point-of-Sale*, Brochure ("Phoenix uses a Linux server for maximum stability and value. The Windows XP workstations use an Internet browser to access the Phoenix application through your own internal 'intranet.' The Web technology behind Phoenix is

the same technology that powers leading ecommerce sites which process millions of simultaneous transactions 24 hours a day, so you know you can rely on its stability. And the MySQL database structure is flexible and secure. Because it is Web based, you can access the program — with all its power — from anywhere you have high-speed Internet. So get out of the restaurant once in a while — secure in the knowledge that Phoenix is in control!")

- FireFly, *POS Software Showcase '07*, Supplement to Hospitality Technology, at 5 ("REMOTE ACCESS FOR REMOTE CONTROL: Web-based technology lets Phoenix Point-of-Sale users get more done in less time with easy anytime, anywhere access to their stores, without tying up an order terminal, without waiting for data updates, and without notifying onsite staff. Live access lets you update prices, run payroll, review reports, even take an order. Plus, proactive alerts to your email or cell phone keep you informed.")
- FireFly Technologies, *Manage Business Information at Your Fingertips– Anytime, Anywhere*, https://web.archive.org/web/20070705200350/http://www.fireflypos.com/phoenixpos/manage2.html (July 5, 2007) ("Just get me out of the restaurant! That's a goal come true for Phoenix owners. They enjoy full access to their system from anywhere in the world that they have Internet access. That's *full* access, without tying up a store terminal, without alerting the staff, and without waiting for data to be uploaded to a third-party Web site. Now you can check in from anywhere, update prices on the fly, do payroll from the beach. You can even configure the system to send text or e-mail messages to

your PDA or cell phone with daypart sales, end-of-day information, and employee overtime alerts. So get out and see the sun once in a while!")

- Simphony
    - Micros Simphony TM, https://web.archive.org/web/20071112050945/http://www.micros.com:80/Products/Simphony/ (2007) ("Although Simphony may be implemented at a single property, the true value can be found in its ability to scale to a business with thousands of workstations spread over many properties. This solution can be hosted using one of MICROS's four worldwide datacenters, or it can be self-hosted by a customer. The Enterprise Management Console allows for the management of the entire system from within a single application. Users are able to create and define the parameters of the enterprise, properties, and revenue centers from any PC that has access to the central server, thus allowing local users to make changes to their configuration if necessary.")
    - Micros Simphony, https://web.archive.org/web/20060324063129/http://www.micros.com/Products/Simphony/ (2006) ("Simphony is MICROS's latest enterprise class hospitality POS product designed specifically for the leisure and entertainment market. Simphony is capable of being centrally-hosted using MICROS's world-class hosting services or it can be self-hosted by a customer. Designed and built for scalability and flexible deployment, Simphony employs a modern services-oriented architecture. Simphony can handle a multi-property POS configuration consisting of thousands of

workstations generating terabytes of sales data, and its web-based reporting solution minimizes the technology footprint on property.").

5. Remote Manager that Can Access the Remote Web Server From Anywhere ("Remote Manager References")

The Asserted Patents purport to provide a solution to the long-recognized problem of managing business operations, and doing so in real time from outside the store or business, and particularly, from anywhere in the world. *E.g.*, '640 patent, 2:6-8.

The benefits of a remote manager that can access the remote web server from anywhere were known long before the Asserted Patents. Implementing a POS system that allows a manager to access the server remotely from anywhere in the world allows increased flexibility in corporate operations, because the manager does not have to be physically located in the same place as the server. In addition, it allows rapid response to problems that may arise, because a manager can log into the server from home or when on a business trip or vacation, and does not have to come into the office.

Remote Managers are ubiquitous in the prior art, for example as described in the following ("Remote Manager References"):

- Woycik at [0013] ("The administration tool application includes a menu editor. The menu editor enables the administrator to create and edit the interactive menu screens provided by the self-order application. The menu editor accesses a library of menu templates containing predefined groupings of buttons, and uses the menu templates, as directed by the administrator, during creation and editing of the interactive menu screens."), [0095] ("The local server kiosk (16, FIG. 1; 82, FIG. 3) includes an administrative tool (98, FIG. 3) to run reports, check the system status, add or remove kiosks from the system, or edit the screen displays that the

customers see. All of these settings are stored on the local server 16, 82. The administrative tool 98 allows an administrator to modify these settings via a local kiosk touch screen. The administrative tool 98 is built using the same enhanced vector graphics approach (e.g., FLASH and .NET) described above for the self-order application interface so that administrators can setup menus and items, and change configuration items primarily via a kiosk touch screen. Therefore, the administrative tool 98 provides a graphical representation of menu options, button selections, images, text fields, etc. on the kiosk. Those items are readily editable through the graphical administrative tool 98 as will be discussed in more detail below. An advantage of this approach is that it enables administrators and store owners to make changes to the store offerings and customer interface without requiring any knowledge of the underlying code. In addition, administration is available from any client kiosk (30, FIG. 1; 80, FIG. 3) configured to communicate with the server 16, 82. This can be implemented by, for example, storing the administrative tool 98 on each client kiosk 30, 80 or by providing the system with the ability to execute the administrative tool 98 on the server 16, 82 using an application interface located on the client kiosks 30, 80. Where the administrative tool 98 is separately stored on each client kiosk 30, 80 and run locally on the client, the tool can be configured with the server IP address of the server kiosk 16, 82 so that, prior to giving the administrator access to change the kiosk, menu, and item setups, it verifies that there are no other instances of the administrative tool 98 running. Also, in this implementation, the configuration data for the kiosks, menus, and items can be stored on the server kiosk 16, 82 and

downloaded to the client kiosk 30, 80 when the administrative tool 98 is run at the client. Changes to the configuration are then uploaded back to the server kiosk 16, 82 when the administrator is done. Thus, there is no need to go to the server 16, 82 in order to use the administrative tools 98 or to modify any of the settings in the client interface. Moreover, remote administration using the administrative tool 98 can be carried out in any of the various ways discussed further above."), [0120] ("As another example, the system includes the ability to perform software updates to either add new functionality or as patches to improve or fix existing software. This software updating can be carried out using the process shown in FIG. 24. Software updates and patches are initially stored on the remote server (block 530). Local servers check the remote software for software updates and patches, preferably during periods of little or no customer activity (block 532). The local server downloads the updates and patches stored on the remote server (block 534). The local server installs the updates and patches that apply to the software running on the local server (block 536). Administrators can choose when to download and install the updates and patches to ensure that installation does not interfere with customer use. Likewise, updates and patches that apply to the clients are pushed to the clients from the local servers and are installed on the clients during inactive periods or at times selected by the administrator (blocks 538, 540)."), [0122] ("The remote server can be used for store owners, franchisees, etc. to update multiple local servers, such as at multiple store locations, from one central location. By tracking the IP addresses or names of each local server registered with the remote server, all local servers can have

updates happen from the central location, eliminating the need for administrators to make the changes at each local server individually. FIG. 25 illustrates one example of one method of updating local servers. For instance, a franchiser can create a new item or modify an existing item with all of its associated attributes and save it to the remote server (blocks 550, 552). This can be done, for example, by using the administrative tool on the remote server to configure the menus and items locally stored there. As previously described above, each local server associated with the remote server can synchronize itself with the information stored on the remote server, and this can be done by having the local server check for changes to the configuration of the self-order application, or the remote server can automatically send the changes to the local server (blocks 554, 556). In the event that not all store locations are intended to use the same menus or pricing, then the changes can be sent to less than all stores and/or local servers. If desired, the administrator of each local server can be given the option to accept or reject each change on each local server using the administrative tools (block 558). Accepted changes can then be updated on each client registered with the local server (block 560).”), Fig. 10

- Manno at [0020] (“The WPOS provides a single, centralized management point for store web pages, and for in-house operations. Store management can change prices, look up stock quantities, check customer databases, in real time from any web-connected computer or PDA. Customers or store clerks can print invoices over the Internet, and customers can check appointments and can even be notified via text messaging to PDA or cell phone when a scheduled job is completed.”),

[0029] (“WPOS can provide staff online training, as determined from data mining results of management back end. Buying patterns of customers, either in-store sales or Internet sales, can be provided easily. Also staff can view real-time store performance via Internet or wireless PDA or cell phone. Mobile retailers (e.g., installers and technicians) can keep instant track of job status and requests. An integrated online scheduler allows for customers to change times and dates of scheduled job orders. Store database and web database can be shared, allowing for management to make changes in store quantities, prices, etc., with such changes being updated instantly at the stores and at the store's web site. Also, management can select which products are sold via Internet, at the store, or both, at the time the product is entered into stock.”)

- Tengler at [0073] (“A manager station 70 allows a manager of the restaurant 2 to view food production and service activity in the restaurant. In some examples, a manager station 70 displays a graphical user interface 190, as illustrated in FIG. 7. The manager station 70 can use a broadband data line 72 to link the manager to the Internet. Referring to FIGS. 7A and 7B, managers can view information about the multiple restaurants 2 using interfaces 192 and 196 over the Internet. These managers can also modify menus and change prices using interfaces 192 and 196. An interface 192 shows a recipe for coffee with options at a particular restaurant 2. A button 194 enables a manager to modify the recipe. An interface 196 shows hot beverages, cold beverages, and shakes available at a particular restaurant 2. The manager's interface is linked directly to the database in the restaurant (without the intervention of any intermediary database or process), so that

information presented to the manager is derived directly from the database of the restaurant and changes made by the manager through the interface are committed directly to the database in the restaurant."), [0103] ("A drive-through display service 610 shows the current customer order on the drive-through displays 394, 396, 418, and 420. The drive-through display service 610 also presents promotional imagery on the display and listens to orders. An automatic beverage dispenser interface 612 interfaces with the automatic beverage service 410 to pour beverages ordered in the drive-through (or from any other configurable source) and listens to orders. A user interface designer 614 allows management to edit the user interface of the register and self-service applications and also saves the specifications in the database 602. In this manner, the user interface designer 614 enables the restaurant management software to be configured for different stores and different brands without having to do a complete redesign. A video system interface 616 uploads digital images from cameras 398, 400, 414, and 416 in coordination with order entry and transmits these images to the RMS system server 602. A self service ordering application 618 handles customer service kiosks 74 that enable customers to enter their own orders. The self service ordering application 618 is a platform for promotions and loyalty programs and can be multi-lingual on a station-by-station basis. A proxy 619 is a proxy between the RMS server 604 and the self-service ordering application 618. A store resident web server 620 enables external managers to view restaurant status information in the database 602 using the interface 190. The store resident web server 620 also allows managers to edit the user interface of the register and self-service

applications and also saves the specifications in the database 602. A printer service 622 drives serial printers 308, 324, and 412. The printer service 622 prints receipts or grill slips on demand, may be shared by multiple applications, and respects the language of client application.")

- Cusack at [0030] ("Application server 152 is coupled to central database 154 and is configured so that systems and processes running with and on application server 152 can make real-time inserts, updates, queries and deletions, and perform other database operations in relation to data stored in central database 154. For example, such real-time database operations may include real-time price changes related to particular items about which POS transactions may relate (e.g., sale transactions, etc.). In particular, the present invention and, in particular, system 150 now permits real-time changes to pricing data say for a large drink at perceived peak times during a business period to maximize revenue or sales, etc. In particular, the present invention now permits a user (e.g., a regional store manager associated with a multiple facility enterprise) to access a web connection and change pricing data, quantity and availability data, etc. to more effectively meet changing market conditions to enhance sales, increase revenues, etc. And, a price or other datum may be reduced to aid slow moving inventory based on real-time transaction data analysis. The present invention now provides real-time control over what may be a global enterprise via a common and well know user medium—the Internet and WWW.")
- Lukkari at 5 ("The restaurant employees access the SmartRestaurant service via a web browser. After logging in with an employee account the employee is

provided with the main menu, which provides access to following functions: configuration of the service, status page of orders (used by chef), listing of all orders (used by cashier) and reports (used by cashier). Fig. 4(a) shows the configuration page, which allows for specifying the daily menu (meals and their prices) and the attributes of daily production process (opening hours, serving hours, delivery period, preparation time of a single order, and the maximum number of orders per delivery period). Based on these attributes, the SmartRestaurant automatically adjusts the sales with the production capacity, by keeping a record of booked delivery times and offering only available delivery times for new orders. In the screenshot the attributes are set so that at most 10 orders can be sold per a single delivery period of 15 minutes.")

- Blinn (US5897622) at 6:9-25 ("Focusing now on the client 100, the client is a general purpose computer. In a preferred embodiment, the client 100 is a conventional personal computer equipped with an operating system supporting Internet communication protocols, such as Microsoft Windows 95 and Microsoft Windows NT, a browser, such as Microsoft Explorer or Netscape Navigator, to access the Merchant System and a conventional modem for access to the Internet 104. In other embodiments, the client 100 could, for example, be a computer workstation, a local area network of computers, an interactive television, an interactive kiosk, a personal digital assistant, an interactive wireless communications device or the like which can interact with the network. While the operating systems may differ in such systems, they will continue to provide the

appropriate communications protocols needed to establish communication links with the network 104.")

- Showghi (US6920431) at 4:24-28 ("FIG. 2 is a diagram of a remote ordering system according to this invention which is based on the public wireless telecommunications and Internet infrastructures using cell phones, two-way pagers, PDAs and the like wireless, hand-held communication devices"); 6:5-22 ("FIG. 2 illustrates a system that is based on the public wireless telecommunications and Internet infrastructures. In this system conventional hand-held wireless communication devices, such as two-way pagers 22 (FIG. 3 a), cell phones 24 (FIG. 3 b) and PDAs 26 (FIG. 3 c) would also be useful as remote ordering devices. Here, a base transceiver station 28, such as a cell tower, transmits menu information to the wireless device and, receives orders with payment information from the wireless device. These signals would be routed by the telecommunications service provider 30 via the Internet 32 to menu/order server 31 which manages the order taking process and financial transaction software applications. When the order is complete, it is transmitted via the Internet 32 to the venue fulfillment center server 16 and order processing computers 18. Depending on the size of the venue and the number of fulfillment centers within it, there may be one or more order processing computers, and there may or may not be a need for a separate server 16."); claim 2 ("The system of

claim 1, where said order-taking computer is an Internet connected server hosting an order-taking application"); FIG. 2;

- Blinn II (US6,058,373) at 7:50-63 ("II. Overview Of The Preferred Electronic Merchandising System This section provides an overview of the preferred electronic merchandising system 100 as illustrated in FIG. 1. The preferred electronic merchandising system 100 is a client/server system which allows merchants to provide a virtual store which processes sales transactions. The electronic merchandising system 100 includes a consumer client 102 (the client) and a merchant system 104 which contains at least one store server process 106 (the server). The consumer client 102 and store server process 106 are in communication with each other by use of a communications medium 108."); 7:64-8:11 ("The consumer client 102 contains a consumer browser 110. The consumer browser 110 communicates with the store server process 106 and displays the web documents created by the store server process 106. Each store server process 106 provides a server architecture that supports the presentation and administration of a virtual store. Preferably, the store server process 106 comprises a number of components including a dynamic page generator 120, an action manager 122, one or more orders 124 and an order processing module 126. Furthermore, in communication with the store server process 106 is a storage device such as a hard disk which contains HTML structures 128 which define the layout of different HTML pages. In addition, the store server process 106 is in

communication with one or more databases 130 with a database module 132.”); FIG. 1

- Costello (WO 01/65427 A1) at Abstract (“A method of conducting a point of sale (POS) transaction includes providing a POS device at the location of the transaction, providing a server at a location remote from the POS device, providing a means of communication between the POS device and the server and downloading a POS program from the server to POS device. The POS program controls the operation of the POS device to conduct a sale. The POS device then receives details of the transaction transmitting details of the POS transaction from POS device to the server. The details of the POS transaction are then stored in a database within the server. A point of sale (POS) system, includes a plurality of POS devices for conducting POS transactions and a server.”); 2:8-28 (“According to a first aspect of the present invention there is provided a method of conducting a point of sale (POS) transaction, including; providing a POS device at the location of the transaction; providing a server at a location remote from the POS device; providing a means of communication between the POS device and the server; providing a POS program for the POS device, the POS program controlling the operation of the POS device to conduct a sale; the POS device receiving details of the transaction; transmitting details of the POS transaction from POS device to the server; and storing the details of the POS transaction in a database within the server. According to a second aspect of the present invention there is a point of sale (POS) system, including; a plurality of POS devices for conducting POS transactions, the POS devices configured to be loaded with a

POS program for controlling the operation of the POS device to conduct the POS transaction and provide the server with details of the POS transactions; and a server arranged to communicate with the POS devices, the server configured to store the details of each POS transaction in a database storage means. Preferably the server downloads the POS device with updates to the POS program"); 5:16-24 ("Referring to Figure 1, there is shown a point of sale (POS) system 10 which includes a server 12, a plurality of POS devices 14, 15A to 15B and a virtual private network 16 facilitating communication between each of the POS devices 14, 15A to 15B and the server 12. The server 12 is provided with a firewall 18 to provide secure access to the server and protect data held within. The POS device 14 is a "thin client" of the server 12, that runs a virtual environment, such as a Java virtual machine that allows uploaded Java language programs to be executed within the virtual machine environment. The server 12 provides an Accounting Backend 15 accessible via a browser running on the POS device 14 or an external PC or laptop."); 5:26-6:9 ("Referring to Figure 2, the POS device 14 is a terminal 20 located at the point the sale is made. It may be loaded with two main components, a point of sale client component and a back office access component. Each of these components is a Java program downloaded from the server 12 through the virtual private network 16 to the POS device 14 and executed within the Java virtual machine. The POS device 14 can be provided with special facilities to cater for a specific industry the POS device 14 is used within. These include input and output means 24. This may be for example, a screen, a keyboard, a weight measuring scale, a bar code reader, a touch screen, a receipt

printer, a bar code scanner or a mouse. The POS device I/O communicates with the substantive POS program 28 using a Java Application Programming Interface (API) called "JavaComm". A smart card device 26 or an EFTPOS terminal may also be included. The smart card device 26 communicates with the POS program 28 using API's called "Opencard" and "SwingGUI". The EFTPOS facility is provided by an appropriate card reader and proprietary Java software for interfacing the card reader with the POS program 28"); 7:1-7 ("The POS program 28 uses a "Store and Forward Messaging" API 36, to communicate with the server 12 using Java Message Service (JMS) protocol and Extensible Markup Language (XML) over the virtual private network 16. This API receives and transmits messages and XML from the server 12. An "Inventory and User Changes Input" API 38 also communicates with the server 12 over the virtual private network by JMS/XML. This API receives reference data including inventory prices from the server 12."); 7:19-30 ("Referring to Figure 1, the backend system is provided within the server 12. This includes a "POS Backend" 13 and an "Accounting Backend" 15. The POS Backend 13 provides business logic, data access, authentication, configuration and gateway services to the "client" POS device 14. It also provides the web browser client user interface via standard web page technology. The "Accounting Backend" 15 provides business logic, web client interface, data access and gateway services to implement an accounting system. The Accounting Backend also provides capabilities for managing multi-site customers, managing the data and reporting requirements for tax compliance, and a data warehouse that provides storage of information obtained from POS sales in

order to provide market data. The data may be provided in a generic form so that privacy issues can be overcome. The backend access is via either the POS device 14 using a web browser client 22 or via a computer running a standard web browser."); 13:23-29 ("The transfer of data between a point of sale device 14 and the server 12 is shown undertaking a POS program update, an update of reference data and recording of a POS transaction on the server. The POS device 14 contacts the server 12. In reply, the server 12 transfers updates to the POS program to the POS device 14. The POS program then downloads the updated reference data. The POS device 14 acknowledges receipt of updates to the POS program and reference data."); Claim 6 ("A method according to claim 1, wherein the server is provided with a POS backend server that downloads the POS program to the POS device."); FIGs. 1, 3, 4 and 11.

- Thompson (WO 02/39226 A2) at Abstract (A system and method for consummating a financial transaction at a POS (12) based on a secure electronic communication over an active, full-time public network such as the Internet (36). The invention comprises at least one POS (12), at least one network appliance (38), at least one primary server (40), and at least one financial service provider (14), the network appliance (38), primary server (40), and financial service provider (14) being in electronic communication over an active full-time public network (36), whereby a financial transaction is consummated at the POS (12) based on the electronic communications over the network (36)."); 12 ("In the Internet paradigm, the present invention is preferably implemented using a computer program that is operative at a web server such as the primary server 40,

which operates a so called "web-site" by known techniques. A flow diagram for such program is attached as Figure 3. An advantage of this embodiment is that all hardware for implementing the invention can be obtained commercially off the shelf."); 12-13 ("The network server program 48 of the primary server 40 is preferably stored in the internal memory device 44 and executed upon initialization of the primary server 40. It preferably includes appropriate display routines for generating a set of display screens that together comprise the user interface for the site, as will be displayed at the POS 12 via the network appliance 38. To that regard, and as noted above, the preferred embodiment of the invention is preferably implemented within at least one primary server 40. Thus, the invention does not require any modifications to conventional POS 12 hardware or software. Furthermore, by housing the software applications on the primary server 40, the single-POS businesses 12 need not afford the cost of the equipment or software or staff to implement the system functionality that is being described and had previously been reserved for the multi-POS businesses 14."); *see also* claim 1, FIG. 3.

- Redmond (WO 03/065178 A2) at [20] ("Fig. 2 illustrates an exemplary embodiment of the physical components of the POS Portal. The POS Portal can be conceptually categorized into three system domains, namely, the POS devices or terminals 20, in-store appliance servers 28 and a services syndication hub 30, each of which will be further described below. It should be understood that the components shown herein merely illustrate an exemplary system configuration and system interactions for each specific domain. Alternatively, for example,

some application services of the POS Portal could be deployed to multiple in-store appliance servers 28 each supporting one or more POS devices 20 for performance purposes or other reasons. Based on the disclosure and teachings provided herein, a person of ordinary skill in the art will know of other ways to implement the POS Portal in accordance with the present invention."); [21] ("The POS devices 20 are located at a merchant or point-of-sale location. Each of the POS devices is configured to deliver one or more of the subscribed services to a merchant's customers and/or employees. In addition, the POS devices 20 are capable of performing a number of functions including, for example, managing PIN entry, signature capturing, content rendering, providing a cash register interface, and functioning as a card reader, etc."); *see also* Claim 1, Claim 25 and FIG. 2.

- Halo
    - NRN, *Juice It Up! moves to Web-based POS*, https://www.nrn.com/print/20670 ("They said Halo's secure architecture will support Juice It Up! management and franchisees in making strategic decisions based on real-time information about different regions, menu items and store performance, while letting them view reports and change menu items from anywhere at any time using a Web browser.")
    - halo, *Halo for Restaurants*, https://web.archive.org/web/20061120125846/http://www.myhalo.com/section.asp?catid=267 (November 20, 2006) ("With Halo for Restaurants, you have the ability to monitor and manage a single store or your entire

chain from anywhere you have access to a web browser. Change prices, add menu items, manage your touch screen terminals and drill down to the transaction level of detail in real time.")

- Medina thesis at 3 ("Halo's main features include: … Halo Enterprise Manager, which allows the restaurant to configure the POS system by setting menu prices and promotions. Restaurants can operate the feature through any web browser."), Fig. 2
- halo, *Updates Return to HomePage Restaurant POS Systems – Halo*, https://web.archive.org/web/20080612142133/http://www.myhalo.com:80/restaurant/restaurant_POS/ (June 12, 2008) ("With the Halo point of sale restaurant system, you have the ability to monitor and manage a single store or your entire chain from anywhere you have access to a web browser. Change prices, add menu items, manage your touch screen terminals and drill down to the transaction level of detail in real time.")
- Green Sheet, *Look to the light: Retail POS system can replace ECR*, The Green Sheet Online Edition Issue 7:06:01 (June 7, 2007) ("Halo includes a touch-screen terminal connected to the Internet. The system tracks details of every transaction (cash, credit and debit) and sends that information via the Internet to a secure data center off-site. Real-time department, category and product sales performance data are accessible by store owners and managers from a Web browser anywhere, anytime. Even when the Internet connection goes down, the Halo offline manager keeps store operations running. And since the data is housed off-site on the Halo secure server,

merchants don't have to worry about losing information in a hard drive crash, theft or natural disaster, such as fire or flood.")

- InfoGenesis
    - William Atkinson, *Point of Service: Now and Later of Restaurant POS* (Apr. 9, 2007) ("Infogenesis hosts Revelation as an application service provider (ASP) solution from a secure data center in California and maintains Internet connectivity with each of the three restaurants, processing all outlet and site transactions in a real-time environment. This approach allows the restaurant to receive the benefits of POS without having to hire a support staff or install server hardware. In addition, the system integrates with those the restaurants have from other vendors.").
- Squirrel
    - Point of Sale, https://web.archive.org/web/20060101034917/http://www.squirrelsystems.com/solutions/pos.html (Jan. 1, 2006) ("Ideal for operations where a back office might not exist, Presto! Gives operators all the functionality and speed of a full point of sale system, paired affordably with basic management functions. Every sales and management task can be performed right in one spot.")
    - SquirrelOne Solutions, https://web.archive.org/web/20051201101248/http://www.squirrelsystems.com/solutions/index.html (Dec. 1, 2005) ("SquirrelOne's open architecture, built on a Microsoft SQL relational database, provides on-

demand access to real time information. Make menu and price changes instantly, at your convenience. Review and analyze live data from any location. Know what's happening in your operations now, not tomorrow. While open architecture also means you can add third-party foodservice programs, you probably won't need to. SquirrelOne is already integrated with leading partner solutions for maximum capability in every management function: inventory control, labor cost, staff scheduling, purchasing, video surveillance, kitchen management, and more.")

- Corporate POS, https://web.archive.org/web/20060327174026/http://www.squirrelsystems.com/solutions/corppos.html (Mar. 27, 2006) ("Corporate POS Manager facilitates the centralization of menu management, freeing restaurant managers from the burden of making menu changes. New information can be sent to locations at any time and can be automatically incorporated into the local database, whenever it's convenient. Built on the latest Microsoft SQL Server relational database which powers some of the largest enterprise systems today. Squirrel's POS Manager, with a familiar browser style desktop is able to be accessed directly from the central PC or remotely via the internet. It has an open architecture Ethernet network communication which is a world wide standard.")

6. Browser Access to Point of Sale Screens ("Browser References")

The Asserted Patents purport to provide a solution to the long-recognized problem of providing point of sale screens instantly viewable anywhere in the world via the world wide web. '640 patent, 5:27-32.

The benefits of browser access to point-of-sale screens were known long before the Asserted Patents. Among other benefits, browsers are standard and commoditized software applications, so no special software or hardware is required. In addition, browsers enable the use of "thin clients" that do not need to run applications (other than the browser) locally because the processing is largely done at the remote server. Such thin clients have reduced requirements for memory and processing power, as compared to clients that need to run the applications locally. Browser access to point-of-sale screens also allows customers to place orders from their own phones, or remotely.

Browser access to point-of-sale screens are ubiquitous in the prior art, for example as described in the following ("Browser References"):

- Woycik at [0075] ("Remote access also allows the chain operator 20 to remotely run the administrative tool. This can be done several ways. One is for the chain operator 20 to have the administrative tool loaded on a remote computer. For this, the remote computer can access the configuration information (menu screens, items, and other settings) at the local server 16, then allow the operator to make changes, and then update the local server 16 with the new configuration information. A second approach is to have the administrative tool loaded on the central server 22 and then provide the chain operator 20 with web access to the central server 22. In this approach, the central server 22 then accesses and stores updated configuration information on the local server 16. A third approach is to provide a web interface to the administrative tool on the local server 16 so that the chain operator 20 can access this interface from any Internet connected general purpose computer using the IP address of the local server 16. This can be done

using any of a number of different remote access and control approaches known to those skilled in the art. Other means of providing remote administration of the kiosks can be utilized."); [0093] ("The system also includes remote web service to allow customers to place orders using a standard web browser via the Internet.")

- Manno at [0015] ("In accordance with an aspect of this invention, a web-based point-of-sale system, or WPOS system, is provided in which a computer at the retail store is programmed and connected with a computer network, i.e., a LAN, a WAN, and/or the Internet, to create a fully functional WPOS that allows general purpose small computers, i.e., personal computers, to connect to the store interface and to serve as point-of-sale stations, and also permits the retail business manager to oversee sales, inventory, and other retail operations over the Internet."), [0016] ("In any of several embodiments, a network based point-of-sale system employs at least one point-of-sale terminal based on a small computer. The computer including the usual means for entering and processing data, means for connecting to said network, one or more ports to which a POS device (i.e., point-of-sale peripheral device) may be connected, and basic software including an operating system and a web browser."), [0017] ("In a multiple-store retail environment, a point-of-sale system can involve a main central computer, i.e., a "Mainframe", a network permitting communication between devices on the network, and also at each store location (i.e., a LAN, a WAN, or the Internet), and at least one point-of-sale terminal at each store location, where the POS terminal employs a computer as described earlier, with a web browser; and one or more associated POS devices to be operated by the computer in connection with a sales

transaction of merchandise or service."), [0046] ("An off-site management facility 54 permits store management personnel to monitor store activity and obtain management reports from the store server(s) 12 and/or from the mainframe 40. Here, a management laptop computer 56 is coupled through the Internet 18 to the system 10, or alternatively, the management functions can be carried out from the remote site via a web-enabled cellular phone or PDA 58, or from another computer 60, each being coupled over the Internet to the system 10. The facility 54 can be at any location, and the computer devices 56, 58, 60 can be wired or wireless. This allows management to check inventory levels, sales activity, and other store activity, as well as historical sales data, customer demographics, or other valuable trade information, from any remote location. Access is provided through the web browser of the computer 56 or other device, without requiring any additional or special software to be installed at the remote location. All necessary software is provided from the server through the web browser. A security functionality requires special access codes and PIN numbers for management level access."), [0051] ("A management functionality for a multiple-store configuration is illustrated in FIG. 4. Here, as discussed in respect to FIG. 1, there is a Store 1 server 12, a Store 2 server 12, and a headquarters office mainframe computer 40, which connect with one another over the Internet 18, together with local network or networks 14 (not shown here). At initial installation of the store servers 12, each store registers with the headquarters main frame computer 40. Then, once the store servers are registered, the information stored on each of the servers 12 is available on the system for management

purposes. At a management location 54, which can be anywhere where there is access to the Internet 18 (or to a LAN at either store location or at the headquarters location), an authorized management employee or officer can access the WPOS system using a standard Internet or web-enabled device such as laptop computer 56, PDA or cell phone 58, or desktop computer 60. As with other implementations previously discussed, the computer device accesses the WPOS system through the computer's web browser. The management employee or officer is provided with a higher level of access, via an access code, PIN, or the like. At the management location, it is possible to view (and print or save) daily or other periodic management reports, as well as to obtain specific management related information. The management employee or officer can write and send email messages to staff or customer, or receive same; to create or alter web pages on the store web sites; to obtain sales and demographic information, i.e., 'data mining'; or to view any other commercial data for either store location in real time.")

- Tengler at [0073] ("The manager station 70 can use a broadband data line 72 to link the manager to the Internet. Referring to FIGS. 7A and 7B, managers can view information about the multiple restaurants 2 using interfaces 192 and 196 over the Internet. These managers can also modify menus and change prices using interfaces 192 and 196.")
- Cusack at [0032] ("To achieve such functionality, application server 152 hosts at least one web page that forms a user interface to the POS system (described below with reference to FIGS. 3A-3D). Such web pages provide in software all the

various functions expected in a POS system (e.g., item selection, pricing, sub and total computations, etc.). Such an application server arrangement is well known, with the front end of a system being stored and served on an application server coupled electronically (via common protocols, for example, SQLNET) to the database, or backend, of the system. One having ordinary skill in the art will immediately understand the configuration described herein."), [0033] ("Each POS terminal 162 can provide standard POS functionality, such as touch screens, cash drawers, etc., and is configured to execute a common web browser, access the global network 175 (e.g., via modem or another network interface, etc.) and download the web pages that make the system's user interface, hosted by application server 152. Accordingly, a store can arrange its POS terminals 162 on a Local Area Network (LAN), as stand alone PC's, dedicated type POS terminals. Such as those manufactured and marketed by JAVELIN (VIPER machines including touch screen entry devices), etc. attached to network 175 via a modem or other common network connection, or in any other well known arrangement such that each POS terminal is provided Internet network access. For example, POS6 is shown directly accessing an ISP and POS5 is shown directly accessing network 175. Each POS terminal 162 may be stationed throughout, such as throughout a retail restaurant establishment, and may be placed anywhere that is accessible to network 175.")

- Olson at Fig 1, [0012] ("The objects of this invention are achieved by point of sale system with a web-based back-office system comprised of one or more point of sale (POS) terminals and one or more web servers. The POS terminals and web

servers communicate via commonly used, standardized internet protocols. The POS terminals are implemented with personal computer hardware and software using one or more relational databases. The web server is implemented with web server hardware and software, also using one or more relational databases."), [0017] ("FIG. 1 shows a high level diagram of this invention. Each point of sale (POS) terminal such as POS 21 represents a single cash register in a store, retail location, or business location. POS 21 is in STORE 1 and POS 2 (22) is in store 2. Each POS includes personal computer hardware and software. Additional POS terminals beyond the two shown, as well as additional stores beyond the two shown, are within the scope of the invention. Each POS normally operates with a hardware/software connection 23 to the Internet or Web. However, if the web goes down, the POS terminal continues to operate. There is a loose coupling of the POS to the back office (BO). The POS to BO connection is not required for the basic business functions of the POS. All transaction data is stored in a relational database on the hard drive in the POS."), [0026] ("Remote and local web browsers 33 illustrate a method used by store managers, corporate administrators, or system administers to access and view a company & its store's instantaneous performance from minute to minute. The POS, BO, and Web Browsers all communicate via the Web. 34. The protocols between the POS and Web browsers and BO servers is HTTPs, XMURPC. 35. XML-RPC is a remote procedure call protocol which uses XML to encode its calls and HTTP as a transport mechanism. It is a very simple protocol which defines only a handful of data types and commands.")

- Lukkari at 2-3 (“Restaurant employees operate the SmartRestaurant service with a laptop equipped with WLAN connectivity and a standard web browser. Laptops with wireless connectivity can be placed freely at the restaurant premises and they also allow the employees be mobile.”), 5 (“The restaurant employees access the SmartRestaurant service via a web browser. After logging in with an employee account the employee is provided with the main menu, which provides access to following functions: configuration of the service, status page of orders (used by chef), listing of all orders (used by cashier) and reports (used by cashier).”)
- Costello (WO 01/65427 A1) at 3:21-24 (“Preferably, each POS device is also configured to act as a browser for interacting with data stored on the server. Preferably, the POS device is preloaded with a browser application. Alternatively, the server downloads a browser program to the POS device to operate each POS device as the browser.”); 4:3-5 (“Preferably, the POS backend application also facilitates the updating or modification of the reference data stored in the database 5 by the browser operating on a POS device.”); 5:22-24 (“The server 12 provides an Accounting Backend 15 accessible via a browser running on the POS device 14 or an external PC or laptop.”); 6:23-31 (“Back office access may be through a standard web browser 30 installed on the POS device 14. Alternatively, a web browser 34 may be accessed on a desktop or laptop 25 computer 22 that has a screen and keyboard 24. The computer 22 accesses the back office access facilities in the form of World Wide Web pages using Hypertext Transfer Protocol (HTTP) through public access Internet 23. A web browser 34 acts an alternative to using the POS device to provide

maintenance facilities including maintaining the POS backend files, such as pricing, products, customers, employees, as 3 o well as access to accounting functions on the server such as, cash book, purchases and sales reports."); 7:29-30 ("The backend access is via either the POS device 14 using a web browser client 22 or via a computer running a standard web browser."); 14:4-8 ("The POS device contacts the server from a browser session. When a particular 5 report is selected a request is sent from the POS device to the server and ·the report assembled in Hypertext Markup Language format from the details stored in the database this report page is downloaded to the browser running on the POS device. The POS device then acknowledges receipt to the browser"); "Claim 9 ("A method according to claim 2, wherein the POS device is provided with a browser program."); claim 10 ("A method according to claim 2, wherein a browser program is downloaded form the server to the POS device, whereby the POS device is operated as a browser"); *see also* claim 11, 12, 28, 29, 30 and 35.

- Redmond (WO 03/065178 A2) at [23] ("The POS device 20 is the primary service delivery component of the POS Portal. The POS device 20 can be configured to deliver specific services subscribed to by a particular merchant. The POS device 20 is an intelligent network device, i.e., it can host applications and process transactions from a wide range of systems and services such as the cash register 22, the in-store appliance server 28, the merchant's back office applications 24 and third party services from affiliated providers via the Internet 31. An exemplary embodiment of the POS device 20 has the following operating characteristics: an embedded real time operating system; a built-in Internet

browser; a touch sensitive screen; audio supports a number of peripherals (magnetic stripe, smart card reader and check readers, keyboards, printers, PIN pad, etc.); supports high resolution color graphics and signature image capture; supports Internet protocols and networking (TCP/IP, Ethernet and Wireless 802.1 lb, Bluetooth, etc.); interfaces to the electronic cash register environment (ECR); supports field upgradeable and network loadable system and application modules; supports EMN specifications; supports PKI based security specifications.")

- 21st Century
    - Hospitality Net, *Ameranth Wireless(TM) Announces Award of Patent for Its Revolutionary 21st Century Restaurant(TM) Products*, https://www.hospitalitynet.org/news/4011916.html (May 16, 2002) ("Keith McNally, Ameranth's principal founder and the lead patent inventor said, 'A principal element of this patent involves the methodology by which hand-held operator displays are synchronized from and with the large displays of fixed, legacy computing systems. The patent covers the use of HTML, XML and similar messaging methodologies to instantly create the displays and GUI on the hand-held's smaller screens – with no custom programming required for each restaurant/hotel or similar application. Another element of the invention, while closely related to the first, addresses the ability for web, wireless and fixed computing systems to be synchronized with each other so that an action on any of the system nodes automatically updates all the other nodes. We believe that this invention will lead to the widespread adoption of wireless and web based

solutions in our markets and that this patent – and others soon to follow – will position us very strongly to retain our leadership position.'")

- Vending Marketwatch, *Kiosk Technology Will Give Vending New Capabilities*, https://www.vendingmarketwatch.com/home/article/10272967/kiosk-technology-will-give-vending-new-capabilities (Dec. 7, 2007) ("The solution operates on Microsoft Windows Mobile-based Pocket PCs and is based on familiar POS touch screen terminals.")

- Phoenix
  - Firefly Technologies, *Put the Power of Phoenix Point-of-Sale*, Brochure ("The Windows XP workstations use an Internet browser to access the Phoenix application through your own internal "intranet.")

- Simphony
  - micros, S*i*mphony™, https://web.archive.org/web/20071112050945/http://www.micros.com:80/Products/Simphony/ (2007) ("Although SimphonyTM may be implemented at a single property, the true value can be found in its ability to scale to a business with thousands of workstations spread over many properties. This solution can be hosted using one of MICROS's four worldwide datacenters, or it can be self-hosted by a customer. The Enterprise Management Console allows for the management of the entire system from within a single application. Users are able to create and define the parameters of the enterprise, properties, and revenue centers

from any PC that has access to the central server, thus allowing local users to make changes to their configuration if necessary.")

7. Operating Point of Sale Terminals Independent of Remote Server ("Independent Terminals References")

The Asserted Patents purport to provide a solution to the long-recognized problem of improved point-of-sale (POS) systems for business operations, "from a single branch to a large chain of stores or branches." *See* '012 patent, 2:12-15. The Asserted Patents describe that "[e]ach POS normally operates with a hardware/software connection 35 to the Internet or Web. However, if the web goes down, the POS terminal continues to operate. There is a 'loose coupling' of the POS to the back office (BO): the POS to BO connection is not required for the basic business functions of the POS. All transaction data is stored in a relational database on the hard drive in the POS." *See* '012 patent, 3:58-65.

The benefits of operating POS terminals independent of remote server(s) were known long before the Asserted Patents. Among other benefits, such POS terminals can continue operating—at least for a time—even if they lose their connection to the server.

POS systems using such independent terminals are ubiquitous in the prior art, for example as described in the following ("Independent Terminals References"):

- Woycik at [0078] ("FIG. 3 illustrates the various software layers running on the client kiosks 80, client/server kiosk 82, and central server 84. As mentioned above, in use the client program is accessed via a user interface that is provided using the touch screen to enable customers to use the system. This is carried out using a multimedia authoring tool to develop a graphical user interface application 90 that runs atop a web services layer 92 that provides the underlying client functionality. At the bottom software layer is the base operating system 94, such as Microsoft Windows™ XP or

WePOS. This operating system includes the .NET Framework. VS.NET, ASP.NET and/or Windows Forms are used for programming to provide the basic client functionality. The user interface to this .NET application is preferably Macromedia Flash™ content (e.g., .swffiles) or some other enhanced vector graphics file format such as SVG. The use of Flash content developed through, for example, ActionScript and as a graphical user interface to Microsoft.NET programming is known to those skilled in the art. Thus, in the illustrated embodiment, the software stored on each kiosk 80 includes a combination of a client computer program 92 (e.g., written in .NET) and user interface content 90 stored in an enhanced vector graphics file format, with this content being used by a suitable player or virtual machine (e.g., Macromedia Flash Player) to generate the user interface on the touch screen.")

- Manno at [0022] ("The WPOS employs web-based architecture, that is, the WPOS employs an in-store web server, and a database connected with the server that allows each client (i.e., each remote POS station) to connect with the server(s). Each in-store server is then registered with the organizational mainframe system. This allows any store location in the system to query any other store's server that is registered with the main frame. When any machine or server fails, management can simply point their store to another server (at another store or at the headquarters) until repairs are made. The computer will automatically update the down system upon restoration. This creates a seamless network of information and minimal down time.")
- Tengler at [0090] ("Each of the cashier stations 302, 304, 382, 402, and 406 has a restaurant management system client application 606 and a database 607. The database 607 stores information about cash and transactions for each cashier station in

addition to information about operators at the cashier station. The databases 607 ensure that each of the cashier stations 302, 304, 382, 402, and 406 is capable of running in a standalone mode by operating on stored or cached data if any of the cashier stations become disconnected from network 462. It is essential that records of cash at cashier stations 302, 304, 382, 402, and 406 be maintained even if any cashier stations become disconnected from network 462. This ensures that theft of cash from cashier stations can be detected. The databases 607 maintain such cash records. The register-specific database 607 includes a shoe-box database that is a version of the shoe-box data maintained by the RMS server 604.")

- Olson at [0017] ("FIG. 1 shows a high level diagram of this invention. Each point of sale (POS) terminal such as POS 21 represents a single cash register in a store, retail location, or business location. POS 21 is in STORE 1 and POS 2 (22) is in store 2. Each POS includes personal computer hardware and software. Additional POS terminals beyond the two shown, as well as additional stores beyond the two shown, are within the scope of the invention. Each POS normally operates with a hardware/software connection 23 to the Internet or Web. However, if the web goes down, the POS terminal continues to operate. There is a loose coupling of the POS to the back office (BO). The POS to BO connection is not required for the basic business functions of the POS. All transaction data is stored in a relational database on the hard drive in the POS.")
- Costello (WO 01/65427 A1) at 6:17-19 (In case of communication difficulties with the server 12 the details are buffered and can be re-sent when communication is re-established.); 7:13-17 ("The POS device 14 is able to operate autonomously in the

event of communication failure by buffering the reference data such as price tables and storing sales information. The information is forwarded upon re-establishment of communication. Full functionality of the POS device 14 may not have to be provided if there is communication problem."); 13:21-23 ("Referring to Figure 11, the POS device can conduct POS transactions without connecting to the server. These transactions are stored in the POS device until a connection to the server is initiated and a connection established.");

- Redmond (WO 03/065178 A2) at [22] ("At the merchant location, the POS 20 devices are implemented such that they are capable of operating in conjunction with a number of existing physical merchant devices and peripherals. For example, the POS devices 20 are able to interface with an in-store network and, alternatively, deliver stand-alone operations in the event of network unavailability. In Fig. 2, a POS device 20 is, for instance, connected to both a cash register 22 and the merchant's back office applications 24 via a local area network 26. The system security protection for the POS Portal is provided by firewalls supporting SSL or virtual private network (VPN), and security modules in the POS device 20. POS devices 20 and the in-store appliance server 28 are digitally certified. Certificate-based authentication is used to authenticate the POS devices 20 before they can connect to the in-store appliance server 28 and subsequently to the services syndication hub 30. The POS Portal leverages standard Internet protocols such as TCP/IP, HTTPS, SOAP, WSDL. The POS Portal also utilizes the network transport common to the Internet including, for example, Digital Subscriber Lines (DSL), Integrated Services Digital Network (ISDN), cable modem or wireless broadband connectivity. These network access

methods provide the digital data bandwidth in the range of 64Kbps to 1.544 Mbps and higher, readily available at most merchant locations.").

- Flashpoint
    - FlashPoint System Setup Manual, 96-98 ("Running in Backup mode . . . If the system does not find the backup server, you will need to run in Stand-alone mode.")
- InfoGenesis
    - InfoGenesis Point-of-Sale, https://web.archive.org/web/20051210182507/http://www.infogenesis.com:80/products/revelation.aspx (Dec. 10, 2005) ("Offline Functionality Keep your operations up and running, even if server connectivity is unavailable . . . What's more, our intelligent POS terminals provide 100% uptime and complete data security. If for any reason a terminal is disconnected from the system server, users can continue to ring up sales with no system downtime. All data can be captured off-line at the POS terminal and transmitted to the server upon reconnection.")
    - William Atkinson, Point of Service: Now and Later of Restaurant POS (Apr. 9, 2007) ("'We also like the off-LAN functionality,' adds Eric Forward, chief financial officer. 'If the connection goes down and is not able to communicate to the server, we can still process all of our orders. We don't need to use 'crash kits'").

- Halo
    - Ahmed H. Medina, *An Analysis of Market Development Strategy of a Point-of-Sale Solutions Provider's Market Research Database* (Summer 2007) (M.B.A. thesis, Simon Fraser University) at 3
        - HALO, which Vivonet introduced in 2004, is an ASP (Application Service Provider) based product which only requires a POS terminal connected via Internet to a central computer server maintained by Vivonet. The ASP technology allows the application logic and data to exist on a remotely located computer server, which the application users access over the internet. In Halo's case, ASP allows multiple outlets of a restaurant chain to share information and lowers the complexity of maintaining the POS system. As of 2006, Halo's customer base has grown to 200 customers.
        - Halo's main features include:

          A touch-screen terminal (shown in Figure 1) installed at the customer site. Halo does not require any computer server at the restaurant location to host the POS software; instead, the touch screen terminal connects via the public Internet to a central application server located at Vivonet's office. In case of a problem with the Internet connection, the Halo terminal at the customer site can switch to offline mode and process POS transactions locally until the connection is restored. . . .

- *FAQs*, HALO, https://web.archive.org/web/20061120125911/http://www.myhalo.com/section.asp?catid=274 at 1
    - 2. What happens if the Internet goes down; do I lose my POS system? No, the Halo Off Line Manager (OLM) keeps store operations running in the event of an Internet outage. The OLM buffers transactions and automatically resynchronizes with the hosted data warehouse once the Internet connection is re-established. In offline mode all POS functions are available as well as POS-based reports that will allow you to perform your end of day and cash out servers and cashiers.

- Simphony
    - Simphony, https://web.archive.org/web/20081014070030/http://www.micros.com/Products/Simphony/ (Oct. 14, 2008) ("The Simphony™ client is fully resilient, capable of performing mission critical operations in the event of an upstream failure. SOA allows for the deployment of local services like property management and credit card interfaces, reporting, and the paperless kitchen display system at the individual property or revenue center to ensure continuous system operation.")
    - Simphony, https://web.archive.org/web/20071112050945/http://www.micros.com/Products/Simphony/ (Nov. 12, 2007) ("The Simphony™ client is fully resilient, capable of performing mission critical operations in the event of an upstream

failure. SOA allows for the deployment of local services like property management and credit card interfaces, reporting, and the paperless kitchen display system at the individual property or revenue center to ensure continuous system operation.")

- Simphony, https://web.archive.org/web/20060324063129/http://www.micros.com/Products/Simphony/ (Mar. 24, 2006) ("Simphony has been designed with security in mind. In the central data center, Simphony is hosted on load-balance-capable application servers along with enterprise database facilities for central management of system configuration and reporting. On property, the workstation software is resilient to temporary network outages, minimizing the impact of network problems on daily operations.").
- MICROS Delivers Harmony to POS Technology with Simphony(TM), http://www.hotel-online.com/News/PR2007_4th/Oct07_MicrosHarmony.html (Oct. 10, 2007) ("In addition to its advanced architecture, Simphony(TM) also features high levels of scalability, resiliency, flexibility, and integration. Simphony(TM) is capable of scaling from a single site to thousands of properties and its multiple levels of resiliency enable the customer to perform mission-critical operations in the event of an upstream technical failure. Simphony(TM)'s flexible configuration and deployment options fulfill the customer's needs regardless of the technical deployment requirements or the size of the enterprise.")

B. Obviousness Combinations

The accompanying claim charts identify how each prior art reference discloses the limitations of the asserted claims on a limitation-by-limitation basis, and illustrative combinations are identified below. If CloudofChange argues that any particular prior art reference lacks any feature for which no combining references are provided in the relevant claim chart, a POSITA as of the Asserted Patents' purported invention date would at a minimum have been motivated to modify the reference to include the allegedly missing feature, or to combine it with other references that include that feature, as discussed below.

1. '640 Patent

Prior art references rendering the asserted claims of the '640 patent obvious, alone or in combination with other references, are discussed below and included in Exhibits A-'640-1 to A-'640-21. Further reasons to combine the references identified in Exhibits A-'640-1 to A-'640-21 include the nature of the problem being solved, the express, implied and inherent teachings of the prior art, the knowledge of persons of ordinary skill in the art, that such combinations would have yielded predictable results, and that such combinations would have represented known alternatives to a POSITA.

Illustrative obviousness combinations are listed below (it should be understood that each listed combination can also include the knowledge of a POSITA). The Asserted Claims of the '640 patent are obvious over:

- Woycik alone or in combination with one or more SaaS References
- Manno alone or in combination with one or more POS Builder References
- Tengler alone or in combination with one or more POS Builder References and one or more SaaS References

- Cusack alone or in combination with one or more POS Builder References and one or more SaaS References
- Olson alone or in combination with one or more POS Builder References
- Lukkari alone or in combination with one or more POS Builder References and one or more SaaS References
- Flashpoint alone or in combination with one or more Remote Web Server References and one or more POS Builder References and one or more SaaS References and one or more Browser References
- Halo alone or in combination with one or more POS Builder References
- InfoGenesis alone or in combination with one or more POS Builder References and one or more Browser References
- Simphony alone or in combination with one or more POS Builder References
- Squirrel alone or in combination with one or more Remote Manager References and one or more SaaS References and one or more POS Builder References and one or more Browser References

a. Woycik Alone or in Combination With One or More SaaS References

Woycik alone renders obvious each of the features of the ’640 patent, as shown in the attached claim chart. *See* Exhibit A-’640-8.

i. SaaS

The ’640 patent claims recite providing a web server as a vendor subscription service (i.e., SaaS). *See* ’640 patent, claim 1 limitation 1[e] (“wherein said web servers are provided as a vendor subscription service wherein web server software resides and is hosted on said vendor's remote servers and wherein subscriber company's POS terminals access and repeatedly interact

with said web server software from said vendor's remote servers, in order to perform the subscriber's desired terminal function, over a network, wherein the network comprises the Internet."), claim 6 ("The web-based point of sale (POS) builder system of claim 1 wherein said POS builder is useable to deliver POS application solutions using software as a subscription (SAAS)."). To the extent that CloudofChange contends that Woycik fails to disclose this feature, a POSITA would have been aware of and would have been motivated to modify the systems and methods of Woycik to incorporate this feature, or alternatively to combine various teachings from the systems and methods described in one or more of the SaaS references with Woycik into one system or method.

This feature is taught by and/or would have been obvious in view of Woycik. For example, Woycik relates generally to "[a] computer-based ordering system for ordering goods and services," and "includes a plurality of self-service client terminals and a server." Woycik, Abstract. Woycik also describes, "[s]elf-service POS systems typically use a dedicated server to communicate to each self-service client terminal. The server can be a general-purpose computer located somewhere on the premises or remotely, and is programmed with software that is used to provide each client with product information for customer ordering. Changes to the products offered via the self-service terminals usually require access to and use of an administration program on the server using common input devices such as a keyboard and mouse to change the product offerings or information about those products (e.g., price)." Woycik, [0008]. Woycik additionally states, "Remote access also allows the chain operator 20 to remotely run the administrative tool. This can be done several ways. One is for the chain operator 20 to have the administrative tool loaded on a remote computer. For this, the remote computer can access the configuration information (menu screens, items, and other settings) at the local server 16, then

allow the operator to make changes, and then update the local server 16 with the new configuration information. A second approach is to have the administrative tool loaded on the central server 22 and then provide the chain operator 20 with web access to the central server 22." Woycik, [0075]. Furthermore, providing the central server and its software as a shared subscription service was an established business model that was well-known for its ability to reduce subscriber cost and information system complexity. Accordingly, a POSITA would have been motivated to offer Woycik's remote web servers as a subscription service, and would have had a reasonable expectation of success in doing so.

To the extent CloudofChange contends that Woycik fails to disclose or render obvious this feature, a POSITA would have been motivated to provide Woycik's remote web servers as a subscription service, as disclosed in the SaaS references. *See* Section VI.A.1 [The Claimed Concepts Were Known, Software as a Subscription / Software as a Service (SaaS)].

Adding SaaS would have realized the many benefits of SaaS. For example, adding SaaS would improve Woycik's system, including Woycik's remote server and POS terminals, by making Woycik's system easier to manage, lowering the cost of maintenance, reducing or eliminating special training to manage, lowering hardware costs, making it easier to update, making it easier to expand to multiple restaurants, making it easier to generate reports, improving the ability to manage from anywhere, and improving the ability to order from anywhere. *See* Section VI.A.1 [The Claimed Concepts Were Known, Software as a Subscription / Software as a Service (SaaS)].

Moreover, adding SaaS would have been predictable to a POSITA because it would have been a combination of prior art elements according to known methods to yield predictable results, because a POSITA would have known of the general availability of SaaS and would have

understood that it would have been beneficial and desirable to add SaaS to a networked computer system, such as a POS system, like the one disclosed in Woycik.

Adding SaaS would have been a simple substitution of one known element (a local or remote server that is maintained by the individual owner) for another (SaaS, where a SaaS provider maintains a server) to obtain predictable results (the benefits of SaaS).

Moreover, adding SaaS would have been obvious to try. When implementing a server, an administrator only has limited options—making the server local or remote, using individually managed hardware or subscribing to hardware, and using individually managed software or subscribing to software (SaaS). Each solution has its known advantages and disadvantages, and a POSITA would have understood the benefits and advantages of SaaS and been motivated to implement SaaS in order to achieve those benefits and advantages.

Furthermore, SaaS was a known approach for computing hardware and software in POS systems, and particularly remote server hardware and software, so a POSITA would have been motivated to implement the teachings of SaaS in the same field (e.g., the teachings of SaaS in the SaaS references, which are in the same field as Woycik) based on design incentives or market forces, including those described in the SaaS references. *See* Section VI.A.1 [The Claimed Concepts Were Known, Software as a Subscription / Software as a Service (SaaS)].

Additionally, a POSITA would have had a reasonable expectation of success in adding this feature to Woycik, as Woycik already discloses a remote server that provides software services from the remote server and allows for software to be updated on the remote server directly, two features that are common with SaaS. In other words, applying SaaS would have merely involved the use of a known technique to improve similar systems in the same way.

b. Manno alone or in combination with one or more POS Builder References

Manno alone renders obvious each of the features of the ’640 patent, as shown in the attached claim chart. *See* Exhibit A-’640-2.

i. POS Builder

The ’640 patent claims recite providing a POS building tool that allows creating menus with no special training. *See* ’640 patent, claim limitation 1[d] (“point of sale builder software that runs on said web server, wherein said local or remote workstations are utilized to build or edit said POS terminals in real time, from anywhere in the world and over the worldwide web”), claim 3 (“The web-based point of sale (POS) builder system of claim 1 wherein use of said builder requires no special training”), claim 8 (“The web-based point of sale (POS) builder system of claim 1 wherein said POS builder adds new POS screens”), claim 9 (“The web-based point of sale (POS) builder system of claim 1 wherein said POS builder edits existing POS screens”), claim 10 (“The web-based point of sale (POS) builder system of claim 1 wherein said POS builder specifies and displays the number, shape and arrangement of selection keys or buttons on said POS screens”), claim 11 (“The web-based point of sale (POS) builder system of claim 1 wherein said POS builder adds items to be associated with said keys or buttons”), claim 12 (“The web-based point of sale (POS) builder system of claim 11 wherein said POS builder assigns attributes such as item name, price, cost, group, taxable and inventory status to said items”), claim 13 (“The web-based point of sale (POS) builder system of claim 12 wherein said attributes are stored, retrievable and changeable from a back office database stored on said web server”). To the extent that CloudofChange contends that Manno fails to disclose this feature, a POSITA would have been aware of and would have been motivated to modify the systems and methods of Manno to incorporate this feature, or alternatively to combine various teachings from

the systems and methods described in one or more of the POS Builder references with Manno into one system or method.

This feature is taught by and/or would have been obvious in view of Manno. For example, Manno relates generally to "[a] network or web-based point of sale (WPOS) system [that] employs a store server and point-of-sale client computers that are connected over a store LAN or over the Internet to the server. The server provides all necessary operating drivers for cash drawer, scanners, and other point-of-sale peripherals via the web browser of the client computers. An instant kiosk retail solution can involve a remote computer terminal installed via the Internet." *See* Manno, Abstract. Mano also discloses, "The WPOS provides a single, centralized management point for store web pages, and for in-house operations. Store management can change prices, look up stock quantities, check customer databases, in real time from any web-connected computer or PDA. Customers or store clerks can print invoices over the Internet, and customers can check appointments and can even be notified via text messaging to PDA or cell phone when a scheduled job is completed." Manno, [0020]. Manno further discloses, "WPOS can provide staff online training, as determined from data mining results of management back end. Buying patterns of customers, either in-store sales or Internet sales, can be provided easily. Also staff can view real-time store performance via Internet or wireless PDA or cell phone. Mobile retailers (e.g., installers and technicians) can keep instant track of job status and requests. An integrated online scheduler allows for customers to change times and dates of scheduled job orders. Store database and web database can be shared, allowing for management to make changes in store quantities, prices, etc., with such changes being updated instantly at the stores and at the store's web site. Also, management can select which products are sold via Internet, at the store, or both, at the time the product is entered into stock." Manno,

[0029]. Furthermore, a POS building tool that allows creating and modifying menus with no special training was a well-known feature of POS systems prior to the patent priority date, and was well-known for its ability to reduce implementation and maintenance costs of POS systems by reducing or eliminating the requirement for specialized IT support (e.g., programmers) to set up or maintain a POS system. Accordingly, a POSITA would have been motivated to provide Manno's POS system with a POS builder tool that allows creating and modifying menus with no special training, and would have had a reasonable expectation of success in doing so.

To the extent CloudofChange contends that Manno fails to disclose or render obvious this feature, a POSITA would have been motivated to add a POS builder tool to Manno's system, as disclosed in the POS Builder references. *See* Section VI.A.2 [The Claimed Concepts Were Known, POS Builder References].

Adding a POS builder tool would have realized the many benefits of POS builder tools. For example, adding a POS builder tool would improve Manno's system by making Manno's system—including Manno's POS terminals—easier to install and manage, lowering the cost of installation and maintenance, reducing or eliminating special training to manage (e.g., allowing any employee—rather than just IT—to add or update menus or other aspects of the POS terminals), and lowering IT costs. *See* Section VI.A.2 [The Claimed Concepts Were Known, POS Builder References].

Moreover, the ability to add a POS building tool, and the benefits of doing so, would have been predictable to a POSITA because it would have been a combination of prior art elements according to known methods to yield predictable results, because a POSITA would have known of the general availability of POS builder tools, and would have understood that it

would have been beneficial and desirable to add a POS builder tool to a POS system, like the one disclosed in Manno.

Adding a POS builder tool would have been a simple substitution of one known element (a POS system that requires programmers to manually build) for another (a POS system with a POS builder tool that anyone can use without special training) to obtain predictable results (the benefits of a POS builder tool).

Moreover, adding a POS builder tool would have been obvious to try. When implementing a POS system, an administrator only has limited options—configuring menus by hiring a programmer or other IT professional to manually program them, or configuring menus with a POS builder tool that requires no special training, and thus is usable by any employee, including said manager. Each solution has its known advantages, and a POSITA would have understood the benefits and advantages of a POS builder tool and been motivated to implement a POS builder tool in order to achieve those benefits and advantages.

Furthermore, using a POS builder tool was a known approach for POS systems, so a POSITA would have been motivated to implement the teachings of a POS builder tool in the same field (e.g., the teachings of POS builder tools in the POS Builder References, which are in the same field as Manno) based on design incentives or market forces, including those described in the POS Builder References. *See* Section VI.A.2 [The Claimed Concepts Were Known, POS Builder References].

Additionally, a POSITA would have had a reasonable expectation of success in adding this feature to Manno, as Manno already discloses that with its POS system, "management [is able] to make changes in store quantities, prices, etc., with such changes being updated instantly at the stores and at the store's web site. Also, management can select which products are sold via

Internet, at the store, or both . . . ." Manno, [0029]. Manno's disclosure of these features is in common with a system that includes a POS builder tool that requires no special training. In other words, adding a POS builder tool would have merely involved the use of a known technique to improve similar systems in the same way.

c. Tengler alone or in combination with one or more POS Builder References and one or more SaaS References

Tengler alone renders obvious each of the features of the '640 patent, as shown in the attached claim chart. *See* Exhibit A-'640-4.

i. POS Builder

The '640 patent claims recite providing a POS building tool that allows creating menus with no special training. *See* '640 patent, claim limitation 1[d] ("point of sale builder software that runs on said web server, wherein said local or remote workstations are utilized to build or edit said POS terminals in real time, from anywhere in the world and over the worldwide web"), claim 3 ("The web-based point of sale (POS) builder system of claim 1 wherein use of said builder requires no special training"), claim 8 ("The web-based point of sale (POS) builder system of claim 1 wherein said POS builder adds new POS screens"), claim 9 ("The web-based point of sale (POS) builder system of claim 1 wherein said POS builder edits existing POS screens"), claim 10 ("The web-based point of sale (POS) builder system of claim 1 wherein said POS builder specifies and displays the number, shape and arrangement of selection keys or buttons on said POS screens"), claim 11 ("The web-based point of sale (POS) builder system of claim 1 wherein said POS builder adds items to be associated with said keys or buttons"), claim 12 ("The web-based point of sale (POS) builder system of claim 11 wherein said POS builder assigns attributes such as item name, price, cost, group, taxable and inventory status to said items"), claim 13 ("The web-based point of sale (POS) builder system of claim 12 wherein said

attributes are stored, retrievable and changeable from a back office database stored on said web server"). To the extent that CloudofChange contends that Tengler fails to disclose this feature, a POSITA would have been aware of and would have been motivated to modify the systems and methods of Tengler to incorporate this feature, or alternatively to combine various teachings from the systems and methods described in one or more of the POS Builder references with Tengler into one system or method.

This feature is taught by and/or would have been obvious in view of Tengler. For example, Tengler relates generally to "an apparatus that includes stations in a quick-serve commercial establishment, the stations enabling users to enter orders, check the status of orders, and assemble orders for delivery, a network interconnecting the stations, each of the stations comprising storage for data about the orders, and software to cause copies of the data to be cached at multiple ones of the stations." Tengler, [0011]. Tengler further discloses "A manager station 70 allows a manager of the restaurant 2 to view food production and service activity in the restaurant. In some examples, a manager station 70 displays a graphical user interface 190, as illustrated in FIG. 7. The manager station 70 can use a broadband data line 72 to link the manager to the Internet. Referring to FIGS. 7A and 7B, managers can view information about the multiple restaurants 2 using interfaces 192 and 196 over the Internet. These managers can also modify menus and change prices using interfaces 192 and 196. An interface 192 shows a recipe for coffee with options at a particular restaurant 2. A button 194 enables a manager to modify the recipe. An interface 196 shows hot beverages, cold beverages, and shakes available at a particular restaurant 2. The manager's interface is linked directly to the database in the restaurant (without the intervention of any intermediary database or process), so that information presented to the manager is derived directly from the database of the restaurant and changes made by the

manager through the interface are committed directly to the database in the restaurant." Tengler, [0073]. Tengler also discloses, "In some examples, the restaurant management software can be configured to use the information about the location of the customer placing the order to modify the prices charged for each item in a food order and displayed in interface 100. For instance, prices could be lower for customers ordering food in the drive-through 26 or the zoom-through 32 than for customers in the dining area 12. The software could also adjust other aspects of the order based on the identity of the order point." Tengler, [0079]. Furthermore, a POS building tool that allows creating and modifying menus with no special training was a well-known feature of POS systems prior to the patent priority date, and was well-known for its ability to reduce implementation and maintenance costs of POS systems by reducing or eliminating the requirement for specialized IT support (e.g., programmers) to set up or maintain a POS system. Accordingly, a POSITA would have been motivated to provide Tengler's POS system with a POS builder tool that allows creating and modifying menus with no special training, and would have had a reasonable expectation of success in doing so.

To the extent CloudofChange contends that Tengler fails to disclose or render obvious this feature, a POSITA would have been motivated to add a POS builder tool to Tengler's system, as disclosed in the POS Builder references. *See* Section VI.A.2 [The Claimed Concepts Were Known, POS Builder References].

Adding a POS builder tool would have realized the many benefits of POS builder tools. For example, adding a POS builder tool would improve Tengler's system by making Tengler's system—including Tengler's POS terminals—easier to install and manage, lowering the cost of installation and maintenance, reducing or eliminating special training to manage (e.g., allowing any employee—rather than just IT—to add or update menus or other aspects of the POS

terminals), and lowering IT costs. *See* Section VI.A.2 [The Claimed Concepts Were Known, POS Builder References].

Moreover, the ability to add a POS building tool, and the benefits of doing so, would have been predictable to a POSITA because it would have been a combination of prior art elements according to known methods to yield predictable results, because a POSITA would have known of the general availability of POS builder tools, and would have understood that it would have been beneficial and desirable to add a POS builder tool to a POS system, like the one disclosed in Tengler.

Adding a POS builder tool would have been a simple substitution of one known element (a POS system that requires programmers to manually build) for another (a POS system with a POS builder tool that anyone can use without special training) to obtain predictable results (the benefits of a POS builder tool).

Moreover, adding a POS builder tool would have been obvious to try. When implementing a POS system, an administrator only has limited options—configuring menus by hiring a programmer or other IT professional to manually program them, or configuring menus with a POS builder tool that requires no special training, and thus is usable by any employee, including said manager. Each solution has its known advantages, and a POSITA would have understood the benefits and advantages of a POS builder tool and been motivated to implement a POS builder tool in order to achieve those benefits and advantages.

Furthermore, using a POS builder tool was a known approach for POS systems, so a POSITA would have been motivated to implement the teachings of a POS builder tool in the same field (e.g., the teachings of POS builder tools in the POS Builder References, which are in the same field as Tengler) based on design incentives or market forces, including those described

in the POS Builder References. *See* Section VI.A.2 [The Claimed Concepts Were Known, POS Builder References].

Additionally, a POSITA would have had a reasonable expectation of success in adding this feature to Tengler. Tengler's disclosure of these features is in common with a system that includes a POS builder tool that requires no special training. In other words, adding a POS builder tool would have merely involved the use of a known technique to improve similar systems in the same way.

ii. SaaS

The '640 patent claims recite providing a web server as a vendor subscription service (i.e., SaaS). *See* '640 patent, claim 1 limitation 1[e] ("wherein said web servers are provided as a vendor subscription service wherein web server software resides and is hosted on said vendor's remote servers and wherein subscriber company's POS terminals access and repeatedly interact with said web server software from said vendor's remote servers, in order to perform the subscriber's desired terminal function, over a network, wherein the network comprises the Internet."), claim 6 ("The web-based point of sale (POS) builder system of claim 1 wherein said POS builder is useable to deliver POS application solutions using software as a subscription (SAAS)."). To the extent that CloudofChange contends that Tengler fails to disclose this feature, a POSITA would have been aware of and would have been motivated to modify the systems and methods of Tengler to incorporate this feature, or alternatively to combine various teachings from the systems and methods described in one or more of the SaaS references with Tengler into one system or method.

This feature is taught by and/or would have been obvious in view of Tengler. For example, Tengler relates generally to "[a]n apparatus that includes stations in a quick-serve commercial establishment, the stations enabling users to enter orders, check the status of orders,

and assemble orders for delivery, a network interconnecting the stations, each of the stations comprising storage for data about the orders, and software to cause copies of the data to be cached at multiple ones of the stations." Tengler, [0011]. Tengler also discloses "[a] system server 68 manages interactions for a central computer database for the restaurant management software." Tengler, [0072]. Additionally, Tengler describes that restaurant management software "includes various components that are executed on different computers" and "is not limited to any specific operating system. In some examples, the restaurant management software 600 is implemented in a Java programming language that enables different components of the software 600 to run on different computer platforms and communicate via application programmer interfaces (API). In some examples, these APIs are provided by high level software tools . . . ." Tengler, [0115]. Furthermore, providing the central server and its software as a shared subscription service was an established business model that was well-known for its ability to reduce subscriber cost and information system complexity. Accordingly, a POSITA would have been motivated to offer Tengler's remote web servers as a subscription service, and would have had a reasonable expectation of success in doing so.

To the extent CloudofChange contends that Tengler fails to disclose or render obvious this feature, a POSITA would have been motivated to provide Tengler's remote web servers as a subscription service, as disclosed in the SaaS references. *See* Section VI.A.1 [The Claimed Concepts Were Known, Software as a Subscription / Software as a Service (SaaS)].

Adding SaaS would have realized the many benefits of SaaS. For example, adding SaaS would improve Tengler's system, including Tengler's remote server and POS terminals, by making Tengler's system easier to manage, lowering the cost of maintenance, reducing or eliminating special training to manage, lowering hardware costs, making it easier to update,

making it easier to expand to multiple restaurants, making it easier to generate reports, improving the ability to manage from anywhere, and improving the ability to order from anywhere. *See* Section VI.A.1 [The Claimed Concepts Were Known, Software as a Subscription / Software as a Service (SaaS)].

Moreover, adding SaaS would have been predictable to a POSITA because it would have been a combination of prior art elements according to known methods to yield predictable results, because a POSITA would have known of the general availability of SaaS and would have understood that it would have been beneficial and desirable to add SaaS to a networked computer system, such as a POS system, like the one disclosed in Tengler.

Adding SaaS would have been a simple substitution of one known element (a local or remote server that is maintained by the individual owner) for another (SaaS, where a SaaS provider maintains a server) to obtain predictable results (the benefits of SaaS).

Moreover, adding SaaS would have been obvious to try. When implementing a server, an administrator only has limited options—making the server local or remote, using individually managed hardware or subscribing to hardware, and using individually managed software or subscribing to software (SaaS). Each solution has its known advantages and disadvantages, and a POSITA would have understood the benefits and advantages of SaaS and been motivated to implement SaaS in order to achieve those benefits and advantages.

Furthermore, SaaS was a known approach for computing hardware and software in POS systems, and particularly remote server hardware and software, so a POSITA would have been motivated to implement the teachings of SaaS in the same field (e.g., the teachings of SaaS in the SaaS references, which are in the same field as Tengler) based on design incentives or market

forces, including those described in the SaaS references. *See* Section VI.A.1 [The Claimed Concepts Were Known, Software as a Subscription / Software as a Service (SaaS)].

Additionally, a POSITA would have had a reasonable expectation of success in adding this feature to Tengler, as Tengler already discloses a remote server that provides software services from the remote server and allows for software to be updated on the remote server directly, two features that are common with SaaS. In other words, applying SaaS would have merely involved the use of a known technique to improve similar systems in the same way.

d. Cusack alone or in combination with one or more POS Builder References and one or more SaaS References

Cusack alone renders obvious each of the features of the ’640 patent, as shown in the attached claim chart. *See* Exhibit A-’640-1.

i. POS Builder

The ’640 patent claims recite providing a POS building tool that allows creating menus with no special training. *See* ’640 patent, claim limitation 1[d] (“point of sale builder software that runs on said web server, wherein said local or remote workstations are utilized to build or edit said POS terminals in real time, from anywhere in the world and over the worldwide web”), claim 3 (“The web-based point of sale (POS) builder system of claim 1 wherein use of said builder requires no special training”), claim 8 (“The web-based point of sale (POS) builder system of claim 1 wherein said POS builder adds new POS screens”), claim 9 (“The web-based point of sale (POS) builder system of claim 1 wherein said POS builder edits existing POS screens”), claim 10 (“The web-based point of sale (POS) builder system of claim 1 wherein said POS builder specifies and displays the number, shape and arrangement of selection keys or buttons on said POS screens”), claim 11 (“The web-based point of sale (POS) builder system of claim 1 wherein said POS builder adds items to be associated with said keys or buttons”), claim

12 (“The web-based point of sale (POS) builder system of claim 11 wherein said POS builder assigns attributes such as item name, price, cost, group, taxable and inventory status to said items”), claim 13 (“The web-based point of sale (POS) builder system of claim 12 wherein said attributes are stored, retrievable and changeable from a back office database stored on said web server”). To the extent that CloudofChange contends that Cusack fails to disclose this feature, a POSITA would have been aware of and would have been motivated to modify the systems and methods of Cusack to incorporate this feature, or alternatively to combine various teachings from the systems and methods described in one or more of the POS Builder references with Cusack into one system or method.

This feature is taught by and/or would have been obvious in view of Cusack. For example, Cusack relates generally to a “[s]ystem and method for facilitating real-time, web based point of sale operations such as sales, reporting, etc. The system and method include and involve a central database management facility storing and managing data related to at least one user interface form operable within a browser application and to items which may be sold or otherwise processed at a point of sale. Also included and involved is a plurality of client data processing facilities which are coupled to the central database management facility via an electronic data network. Each client data processing system is located at a remote point of sale and is coupled to the central database management facility via a network connection over the electronic data network. Each client data processing facility is configured to execute the browser application, to access the central database management facility via the network connection to retrieve and process at least one user interface form within the browser application, to access the central database management facility in real time to retrieve data related to at least one of the items to facilitate a remote point of sale related operation, and to cause the central database

management facility to manage the data based on the remote point of sale related operation." Cusack, Abstract. Cusack also discloses, "[a]pplication server 152 is coupled to central database 154 and is configured so that systems and processes running with and on application server 152 can make real-time inserts, updates, queries and deletions, and perform other database operations in relation to data stored in central database 154. For example, such real-time database operations may include real-time price changes related to particular items about which POS transactions may relate (e.g., sale transactions, etc.). In particular, the present invention and, in particular, system 150 now permits real-time changes to pricing data say for a large drink at perceived peak times during a business period to maximize revenue or sales, etc. In particular, the present invention now permits a user (e.g., a regional store manager associated with a multiple facility enterprise) to access a web connection and change pricing data, quantity and availability data, etc. to more effectively meet changing market conditions to enhance sales, increase revenues, etc. And, a price or other datum may be reduced to aid slow moving inventory based on real-time transaction data analysis. The present invention now provides real-time control over what may be a global enterprise via a common and well know user medium—the Internet and WWW." Cusack, [0030]. Furthermore, a POS building tool that allows creating and modifying menus with no special training was a well-known feature of POS systems prior to the patent priority date, and was well-known for its ability to reduce implementation and maintenance costs of POS systems by reducing or eliminating the requirement for specialized IT support (e.g., programmers) to set up or maintain a POS system. Accordingly, a POSITA would have been motivated to provide Cusack's POS system with a POS builder tool that allows creating and modifying menus with no special training, and would have had a reasonable expectation of success in doing so.

To the extent CloudofChange contends that Cusack fails to disclose or render obvious this feature, a POSITA would have been motivated to add a POS builder tool to Cusack's system, as disclosed in the POS Builder references. *See* Section VI.A.2 [The Claimed Concepts Were Known, POS Builder References].

Adding a POS builder tool would have realized the many benefits of POS builder tools. For example, adding a POS builder tool would improve Cusack's system by making Cusack's system—including Cusack's POS terminals—easier to install and manage, lowering the cost of installation and maintenance, reducing or eliminating special training to manage (e.g., allowing any employee—rather than just IT—to add or update menus or other aspects of the POS terminals), and lowering IT costs. *See* Section VI.A.2 [The Claimed Concepts Were Known, POS Builder References].

Moreover, the ability to add a POS building tool, and the benefits of doing so, would have been predictable to a POSITA because it would have been a combination of prior art elements according to known methods to yield predictable results, because a POSITA would have known of the general availability of POS builder tools, and would have understood that it would have been beneficial and desirable to add a POS builder tool to a POS system, like the one disclosed in Cusack.

Adding a POS builder tool would have been a simple substitution of one known element (a POS system that requires programmers to manually build) for another (a POS system with a POS builder tool that anyone can use without special training) to obtain predictable results (the benefits of a POS builder tool).

Moreover, adding a POS builder tool would have been obvious to try. When implementing a POS system, an administrator only has limited options—configuring menus by

hiring a programmer or other IT professional to manually program them, or configuring menus with a POS builder tool that requires no special training, and thus is usable by any employee, including said manager. Each solution has its known advantages, and a POSITA would have understood the benefits and advantages of a POS builder tool and been motivated to implement a POS builder tool in order to achieve those benefits and advantages.

Furthermore, using a POS builder tool was a known approach for POS systems, so a POSITA would have been motivated to implement the teachings of a POS builder tool in the same field (e.g., the teachings of POS builder tools in the POS Builder References, which are in the same field as Cusack) based on design incentives or market forces, including those described in the POS Builder References. *See* Section VI.A.2 [The Claimed Concepts Were Known, POS Builder References].

Additionally, a POSITA would have had a reasonable expectation of success in adding this feature to Cusack. Cusack's disclosure of these features is in common with a system that includes a POS builder tool that requires no special training. In other words, adding a POS builder tool would have merely involved the use of a known technique to improve similar systems in the same way.

ii. SaaS

The '640 patent claims recite providing a web server as a vendor subscription service (i.e., SaaS). *See* '640 patent, claim 1 limitation 1[e] ("wherein said web servers are provided as a vendor subscription service wherein web server software resides and is hosted on said vendor's remote servers and wherein subscriber company's POS terminals access and repeatedly interact with said web server software from said vendor's remote servers, in order to perform the subscriber's desired terminal function, over a network, wherein the network comprises the Internet."), claim 6 ("The web-based point of sale (POS) builder system of claim 1 wherein said

POS builder is useable to deliver POS application solutions using software as a subscription (SAAS).”). To the extent that CloudofChange contends that Cusack fails to disclose this feature, a POSITA would have been aware of and would have been motivated to modify the systems and methods of Cusack to incorporate this feature, or alternatively to combine various teachings from the systems and methods described in one or more of the SaaS references with Cusack into one system or method.

This feature is taught by and/or would have been obvious in view of Cusack. For example, Cusack relates generally to a “[s]ystem and method for facilitating real-time, web based point of sale operations such as sales, reporting, etc. The system and method include and involve a central database management facility storing and managing data related to at least one user interface form operable within a browser application and to items which may be sold or otherwise processed at a point of sale. Also included and involved is a plurality of client data processing facilities which are coupled to the central database management facility via an electronic data network. Each client data processing system is located at a remote point of sale and is coupled to the central database management facility via a network connection over the electronic data network. Each client data processing facility is configured to execute the browser application, to access the central database management facility via the network connection to retrieve and process at least one user interface form within the browser application, to access the central database management facility in real time to retrieve data related to at least one of the items to facilitate a remote point of sale related operation, and to cause the central database management facility to manage the data based on the remote point of sale related operation.” Cusack, Abstract. Cusack discloses “POS system 150 further includes an application server 152 and a central database 154 which can be integrated with back office systems 156. Application

server 152 can be any well know web server that is coupled to a network 175 (e.g., a globally accessible network such as the Internet and World Wide Web (WWW), a network incorporating open-standards based technologies and protocols, etc.) and is configured to serve and process content such as web pages . . . ." Cusack, [0029]. Cusack discloses that "application server 152 hosts at least one web page that forms a user interface to the POS system (described below with reference to FIGS. 3A-3D). Such web pages provide in software all the various functions expected in a POS system (e.g., item selection, pricing, sub and total computations, etc.). Such an application server arrangement is well known, with the front end of a system being stored and served on an application server coupled electronically (via common protocols, for example, SQLNET) to the database, or backend, of the system." Cusack, [0032]. Additionally, Cusack describes, "a management operation related to POS transaction may involve accessing central data storage facilities via the WWW and engaging in what may otherwise be considered an on-line query which is processed within system 150." Cusack, [0065]. Furthermore, providing the central server and its software as a shared subscription service was an established business model that was well-known for its ability to reduce subscriber cost and information system complexity. Accordingly, a POSITA would have been motivated to offer Cusack's remote web servers as a subscription service, and would have had a reasonable expectation of success in doing so.

To the extent CloudofChange contends that Cusack fails to disclose or render obvious this feature, a POSITA would have been motivated to provide Cusack's remote web servers as a subscription service, as disclosed in the SaaS references. *See* Section VI.A.1 [The Claimed Concepts Were Known, Software as a Subscription / Software as a Service (SaaS)].

Adding SaaS would have realized the many benefits of SaaS. For example, adding SaaS would improve Cusack's system, including Cusack's remote server and POS terminals, by

making Cusack's system easier to manage, lowering the cost of maintenance, reducing or eliminating special training to manage, lowering hardware costs, making it easier to update, making it easier to expand to multiple restaurants, making it easier to generate reports, improving the ability to manage from anywhere, and improving the ability to order from anywhere. *See* Section VI.A.1 [The Claimed Concepts Were Known, Software as a Subscription / Software as a Service (SaaS)].

Moreover, adding SaaS would have been predictable to a POSITA because it would have been a combination of prior art elements according to known methods to yield predictable results, because a POSITA would have known of the general availability of SaaS and would have understood that it would have been beneficial and desirable to add SaaS to a networked computer system, such as a POS system, like the one disclosed in Cusack.

Adding SaaS would have been a simple substitution of one known element (a local or remote server that is maintained by the individual owner) for another (SaaS, where a SaaS provider maintains a server) to obtain predictable results (the benefits of SaaS).

Moreover, adding SaaS would have been obvious to try. When implementing a server, an administrator only has limited options—making the server local or remote, using individually managed hardware or subscribing to hardware, and using individually managed software or subscribing to software (SaaS). Each solution has its known advantages and disadvantages, and a POSITA would have understood the benefits and advantages of SaaS and been motivated to implement SaaS in order to achieve those benefits and advantages.

Furthermore, SaaS was a known approach for computing hardware and software in POS systems, and particularly remote server hardware and software, so a POSITA would have been motivated to implement the teachings of SaaS in the same field (e.g., the teachings of SaaS in the

SaaS references, which are in the same field as Cusack) based on design incentives or market forces, including those described in the SaaS references. *See* Section VI.A.1 [The Claimed Concepts Were Known, Software as a Subscription / Software as a Service (SaaS)].

Additionally, a POSITA would have had a reasonable expectation of success in adding this feature to Cusack, as Cusack already discloses a remote server that provides software services from the remote server and allows for software to be updated on the remote server directly, two features that are common with SaaS. In other words, applying SaaS would have merely involved the use of a known technique to improve similar systems in the same way.

e. Olson alone or in combination with one or more POS Builder References

Olson alone renders obvious each of the features of the ’640 patent, as shown in the attached claim chart. *See* Exhibit A-’640-10.

i. POS Builder

The ’640 patent claims recite providing a POS building tool that allows creating menus with no special training. *See* ’640 patent, claim limitation 1[d] (“point of sale builder software that runs on said web server, wherein said local or remote workstations are utilized to build or edit said POS terminals in real time, from anywhere in the world and over the worldwide web”), claim 3 (“The web-based point of sale (POS) builder system of claim 1 wherein use of said builder requires no special training”), claim 8 (“The web-based point of sale (POS) builder system of claim 1 wherein said POS builder adds new POS screens”), claim 9 (“The web-based point of sale (POS) builder system of claim 1 wherein said POS builder edits existing POS screens”), claim 10 (“The web-based point of sale (POS) builder system of claim 1 wherein said POS builder specifies and displays the number, shape and arrangement of selection keys or buttons on said POS screens”), claim 11 (“The web-based point of sale (POS) builder system of

claim 1 wherein said POS builder adds items to be associated with said keys or buttons"), claim 12 ("The web-based point of sale (POS) builder system of claim 11 wherein said POS builder assigns attributes such as item name, price, cost, group, taxable and inventory status to said items"), claim 13 ("The web-based point of sale (POS) builder system of claim 12 wherein said attributes are stored, retrievable and changeable from a back office database stored on said web server"). To the extent that CloudofChange contends that Olson fails to disclose this feature, a POSITA would have been aware of and would have been motivated to modify the systems and methods of Olson to incorporate this feature, or alternatively to combine various teachings from the systems and methods described in one or more of the POS Builder references with Olson into one system or method.

This feature is taught by and/or would have been obvious in view of Olson. For example, Olson relates generally to "the managing of the points of sale (POS) with Web-based back-office systems. POS transaction engine terminals are implemented on personal computer, PC hardware using PC software with a relational database. The transaction engine architecture provides cost-effective application customization. Web-based back-office software is implemented on servers. POS and BO systems are monitored and supported via the web using protocol and open software components." Olson, Abstract. Olson further discloses "the BO interface requests update from the BO such as new items, price changes, employees, etc. The POS terminals communicate via HTTP protocol (hypertext transfer protocol) 23 with Back-office BO software, which is implemented on web servers 24, which can be located anywhere in the world. In addition, the BO software and data can be viewed from any store employee at any PC who has Internet access 26 and a password." Olson, [0019]. Olson also discloses, "A client who is the store manager or owner selects a promotion type, associates it with an item, department, etc, then sets the

parameters that control how that promotion works. These transaction transmissions between the POS and the BO can be encrypted to insure privacy and security. A typical encryption method is 128 bit SSL (secure sockets layer). A further element of security is that each BO client (individual POS, store or multi-store owner) gets their own instance of a database. When they log into the BO they are attached to their own relational database associated and validated via their user login and password." Olson, [0020]. Furthermore, a POS building tool that allows creating and modifying menus with no special training was a well-known feature of POS systems prior to the patent priority date, and was well-known for its ability to reduce implementation and maintenance costs of POS systems by reducing or eliminating the requirement for specialized IT support (e.g., programmers) to set up or maintain a POS system. Accordingly, a POSITA would have been motivated to provide Olson's POS system with a POS builder tool that allows creating and modifying menus with no special training, and would have had a reasonable expectation of success in doing so.

To the extent CloudofChange contends that Olson fails to disclose or render obvious this feature, a POSITA would have been motivated to add a POS builder tool to Olson's system, as disclosed in the POS Builder references. *See* Section VI.A.2 [The Claimed Concepts Were Known, POS Builder References].

Adding a POS builder tool would have realized the many benefits of POS builder tools. For example, adding a POS builder tool would improve Olson's system by making Olson's system—including Olson's POS terminals—easier to install and manage, lowering the cost of installation and maintenance, reducing or eliminating special training to manage (e.g., allowing any employee—rather than just IT—to add or update menus or other aspects of the POS

terminals), and lowering IT costs. *See* Section VI.A.2 [The Claimed Concepts Were Known, POS Builder References].

Moreover, the ability to add a POS building tool, and the benefits of doing so, would have been predictable to a POSITA because it would have been a combination of prior art elements according to known methods to yield predictable results, because a POSITA would have known of the general availability of POS builder tools, and would have understood that it would have been beneficial and desirable to add a POS builder tool to a POS system, like the one disclosed in Olson.

Adding a POS builder tool would have been a simple substitution of one known element (a POS system that requires programmers to manually build) for another (a POS system with a POS builder tool that anyone can use without special training) to obtain predictable results (the benefits of a POS builder tool).

Moreover, adding a POS builder tool would have been obvious to try. When implementing a POS system, an administrator only has limited options—configuring menus by hiring a programmer or other IT professional to manually program them, or configuring menus with a POS builder tool that requires no special training, and thus is usable by any employee, including said manager. Each solution has its known advantages, and a POSITA would have understood the benefits and advantages of a POS builder tool and been motivated to implement a POS builder tool in order to achieve those benefits and advantages.

Furthermore, using a POS builder tool was a known approach for POS systems, so a POSITA would have been motivated to implement the teachings of a POS builder tool in the same field (e.g., the teachings of POS builder tools in the POS Builder References, which are in the same field as Olson) based on design incentives or market forces, including those described

in the POS Builder References. *See* Section VI.A.2 [The Claimed Concepts Were Known, POS Builder References].

Additionally, a POSITA would have had a reasonable expectation of success in adding this feature to Olson. Olson's disclosure of these features is in common with a system that includes a POS builder tool that requires no special training. In other words, adding a POS builder tool would have merely involved the use of a known technique to improve similar systems in the same way.

f. Lukkari alone or in combination with one or more POS Builder References and one or more SaaS References

Lukkari alone renders obvious each of the features of the '640 patent, as shown in the attached claim chart. *See* Exhibit A-'640-11.

i. POS Builder

The '640 patent claims recite providing a POS building tool that allows creating menus with no special training. *See* '640 patent, claim limitation 1[d] ("point of sale builder software that runs on said web server, wherein said local or remote workstations are utilized to build or edit said POS terminals in real time, from anywhere in the world and over the worldwide web"), claim 3 ("The web-based point of sale (POS) builder system of claim 1 wherein use of said builder requires no special training"), claim 8 ("The web-based point of sale (POS) builder system of claim 1 wherein said POS builder adds new POS screens"), claim 9 ("The web-based point of sale (POS) builder system of claim 1 wherein said POS builder edits existing POS screens"), claim 10 ("The web-based point of sale (POS) builder system of claim 1 wherein said POS builder specifies and displays the number, shape and arrangement of selection keys or buttons on said POS screens"), claim 11 ("The web-based point of sale (POS) builder system of claim 1 wherein said POS builder adds items to be associated with said keys or buttons"), claim

12 (“The web-based point of sale (POS) builder system of claim 11 wherein said POS builder assigns attributes such as item name, price, cost, group, taxable and inventory status to said items”), claim 13 (“The web-based point of sale (POS) builder system of claim 12 wherein said attributes are stored, retrievable and changeable from a back office database stored on said web server”). To the extent that CloudofChange contends that Lukkari fails to disclose this feature, a POSITA would have been aware of and would have been motivated to modify the systems and methods of Lukkari to incorporate this feature, or alternatively to combine various teachings from the systems and methods described in one or more of the POS Builder references with Lukkari into one system or method.

This feature is taught by and/or would have been obvious in view of Lukkari. For example, Lukkari relates generally to “SmartRestaurant service, which allows customers to use mobile devices for ordering and paying lunches from a nearby campus restaurant beforehand. Further, SmartRestaurant provides the restaurant with means of adjusting the sales with production capacity and prior knowledge of upcoming orders.” Lukkari, 1. Lukkari also discloses, “SmartRestaurant is designed for a context-aware mobile service system such as SmartRotuaari [11], allowing customers to use mobile devices for ordering and paying lunches from a nearby restaurant beforehand. … The SmartRestaurant service is a new feature in the SmartRotuaari service system, which offers context aware mobile multimedia services to consumers at a city center.” Lukkari, 2. Lukkari also discloses, “restaurant employees access the SmartRestaurant service via a web browser. After logging in with an employee account the employee is provided with the main menu, which provides access to following functions: configuration of the service, status page of orders (used by chef), listing of all orders (used by cashier) and reports (used by cashier).” Lukkari, 5. Lukkari discloses a “configuration page,

which allows for specifying the daily menu (meals and their prices) and the attributes of daily production process (opening hours, serving hours, delivery period, preparation time of a single order, and the maximum number of orders per delivery period)." Lukkari, 5. Furthermore, a POS building tool that allows creating and modifying menus with no special training was a well-known feature of POS systems prior to the patent priority date, and was well-known for its ability to reduce implementation and maintenance costs of POS systems by reducing or eliminating the requirement for specialized IT support (e.g., programmers) to set up or maintain a POS system. Accordingly, a POSITA would have been motivated to provide Lukkari's POS system with a POS builder tool that allows creating and modifying menus with no special training, and would have had a reasonable expectation of success in doing so.

To the extent CloudofChange contends that Lukkari fails to disclose or render obvious this feature, a POSITA would have been motivated to add a POS builder tool to Lukkari's system, as disclosed in the POS Builder references. *See* Section VI.A.2 [The Claimed Concepts Were Known, POS Builder References].

Adding a POS builder tool would have realized the many benefits of POS builder tools. For example, adding a POS builder tool would improve Lukkari's system by making Lukkari's system—including Lukkari's POS terminals—easier to install and manage, lowering the cost of installation and maintenance, reducing or eliminating special training to manage (e.g., allowing any employee—rather than just IT—to add or update menus or other aspects of the POS terminals), and lowering IT costs. *See* Section VI.A.2 [The Claimed Concepts Were Known, POS Builder References].

Moreover, the ability to add a POS building tool, and the benefits of doing so, would have been predictable to a POSITA because it would have been a combination of prior art

elements according to known methods to yield predictable results, because a POSITA would have known of the general availability of POS builder tools, and would have understood that it would have been beneficial and desirable to add a POS builder tool to a POS system, like the one disclosed in Lukkari.

Adding a POS builder tool would have been a simple substitution of one known element (a POS system that requires programmers to manually build) for another (a POS system with a POS builder tool that anyone can use without special training) to obtain predictable results (the benefits of a POS builder tool).

Moreover, adding a POS builder tool would have been obvious to try. When implementing a POS system, an administrator only has limited options—configuring menus by hiring a programmer or other IT professional to manually program them, or configuring menus with a POS builder tool that requires no special training, and thus is usable by any employee, including said manager. Each solution has its known advantages, and a POSITA would have understood the benefits and advantages of a POS builder tool and been motivated to implement a POS builder tool in order to achieve those benefits and advantages.

Furthermore, using a POS builder tool was a known approach for POS systems, so a POSITA would have been motivated to implement the teachings of a POS builder tool in the same field (e.g., the teachings of POS builder tools in the POS Builder References, which are in the same field as Lukkari) based on design incentives or market forces, including those described in the POS Builder References. *See* Section VI.A.2 [The Claimed Concepts Were Known, POS Builder References].

Additionally, a POSITA would have had a reasonable expectation of success in adding this feature to Lukkari. Lukkari's disclosure of these features is in common with a system that

includes a POS builder tool that requires no special training. In other words, adding a POS builder tool would have merely involved the use of a known technique to improve similar systems in the same way.

ii. SaaS

The ’640 patent claims recite providing a web server as a vendor subscription service (i.e., SaaS). *See* ’640 patent, claim 1 limitation 1[e] (“wherein said web servers are provided as a vendor subscription service wherein web server software resides and is hosted on said vendor's remote servers and wherein subscriber company's POS terminals access and repeatedly interact with said web server software from said vendor's remote servers, in order to perform the subscriber's desired terminal function, over a network, wherein the network comprises the Internet.”), claim 6 (“The web-based point of sale (POS) builder system of claim 1 wherein said POS builder is useable to deliver POS application solutions using software as a subscription (SAAS).”). To the extent that CloudofChange contends that Lukkari fails to disclose this feature, a POSITA would have been aware of and would have been motivated to modify the systems and methods of Lukkari to incorporate this feature, or alternatively to combine various teachings from the systems and methods described in one or more of the SaaS references with Lukkari into one system or method.

This feature is taught by and/or would have been obvious in view of Lukkari. For example, Lukkari relates generally to “SmartRestaurant service, which allows customers to use mobile devices for ordering and paying lunches from a nearby campus restaurant beforehand. Further, SmartRestaurant provides the restaurant with means of adjusting the sales with production capacity and prior knowledge of upcoming orders.” Lukkari, 1. Lukkari also describes, “The SmartRestaurant service is a new feature in the SmartRotuaari service system, which offers context aware mobile multimedia services to consumers at a city center.” Lukkari,

2. Lukkari also discloses "SmartRestaurant's run-time view is presented in Fig. 2(a). rotuaari.net is a multi-processor host that runs the SmartRestaurant service and other server components: Apache web server, Tomcat application server and MySQL database." Lukkari, 3. Furthermore, providing the central server and its software as a shared subscription service was an established business model that was well-known for its ability to reduce subscriber cost and information system complexity, and it would have made sense to charge restaurants that use SmartRestaurant a subscription fee for the ability to access the SmartRotuaari service system. Accordingly, a POSITA would have been motivated to offer Lukkari's remote web servers as a subscription service, and would have had a reasonable expectation of success in doing so.

To the extent CloudofChange contends that Lukkari fails to disclose or render obvious this feature, a POSITA would have been motivated to provide Lukkari's remote web servers as a subscription service, as disclosed in the SaaS references. *See* Section VI.A.1 [The Claimed Concepts Were Known, Software as a Subscription / Software as a Service (SaaS)].

Adding SaaS would have realized the many benefits of SaaS. For example, adding SaaS would improve Lukkari's system, including Lukkari's remote server and POS terminals, by making Lukkari's system easier to manage, lowering the cost of maintenance, reducing or eliminating special training to manage, lowering hardware costs, making it easier to update, making it easier to expand to multiple restaurants, making it easier to generate reports, improving the ability to manage from anywhere, and improving the ability to order from anywhere. *See* Section VI.A.1 [The Claimed Concepts Were Known, Software as a Subscription / Software as a Service (SaaS)].

Moreover, adding SaaS would have been predictable to a POSITA because it would have been a combination of prior art elements according to known methods to yield predictable

results, because a POSITA would have known of the general availability of SaaS and would have understood that it would have been beneficial and desirable to add SaaS to a networked computer system, such as a POS system, like the one disclosed in Lukkari.

Adding SaaS would have been a simple substitution of one known element (a local or remote server that is maintained by the individual owner) for another (SaaS, where a SaaS provider maintains a server) to obtain predictable results (the benefits of SaaS).

Moreover, adding SaaS would have been obvious to try. When implementing a server, an administrator only has limited options—making the server local or remote, using individually managed hardware or subscribing to hardware, and using individually managed software or subscribing to software (SaaS). Each solution has its known advantages and disadvantages, and a POSITA would have understood the benefits and advantages of SaaS and been motivated to implement SaaS in order to achieve those benefits and advantages.

Furthermore, SaaS was a known approach for computing hardware and software in POS systems, and particularly remote server hardware and software, so a POSITA would have been motivated to implement the teachings of SaaS in the same field (e.g., the teachings of SaaS in the SaaS references, which are in the same field as Lukkari) based on design incentives or market forces, including those described in the SaaS references. *See* Section VI.A.1 [The Claimed Concepts Were Known, Software as a Subscription / Software as a Service (SaaS)].

Additionally, a POSITA would have had a reasonable expectation of success in adding this feature to Lukkari, as Lukkari already discloses a remote server that provides software services from the remote server and allows for software to be updated on the remote server directly, two features that are common with SaaS. In other words, applying SaaS would have merely involved the use of a known technique to improve similar systems in the same way.

g. Flashpoint alone or in combination with one or more Remote Web Server References and one or more POS Builder References and one or more SaaS References and one or more Browser References

Flashpoint alone renders obvious each of the features of the ’640 patent, as shown in the attached claim chart. *See* Exhibit A-’640-13.

i. Remote Web Server

The ’640 patent claims recite a web server that is remote from one or more point of sale terminals. *See* ’640 patent, claim 1 limitation 1[b] (“an internet connection from said one or more point of sale terminals to a web server”), 1[e] (“wherein said web servers are provided as a vendor subscription service wherein web server software resides and is hosted on said vendor's remote servers and wherein subscriber company's POS terminals access and repeatedly interact with said web server software from said vendor's remote servers, in order to perform the subscriber's desired terminal function, over a network, wherein the network comprises the Internet.”).

This feature is taught by and/or would have been obvious in view of Flashpoint. For example, Flashpoint relates generally to a POS system that offers “integrated software” including “mission-critical point of sale applications and back office management tools” as well as “enterprise-wide management solutions.” Flashpoint, https://web.archive.org/web/20071130010505/http://www.trimpos.net:80/ (Nov. 3, 2007). Restaurants using Flashpoint could use “corporate software” on a remote server at the corporate office to update menus on the remote server. *See* FLASHPOINT™ P.O.S. Software, Flashpoint Operations Guide, System Setup Manual at 4 (“Once the System Set Up button has been selected the following screen will appear, allowing you to complete many of the setups for Flashpoint. If you are using the corporate software, you must select the location to setup before going into

system setup."). Flashpoint also discloses a local server computer. *See, e.g.*, Flashpoint, *Quick Food Service*, https://web.archive.org/web/20071031033856/http:/www.trimpos.net/sitefiles/index.html (Oct. 31, 2007) ("Use the back office station for confidential reporting and daily management needs, allowing the POS stations to be dedicated to order processing."); FLASHPOINT™ P.O.S. Software, Flashpoint Operations Guide, System Setup Manual at 48 ("Back Office Station – If set, the station will automatically display the system menu once the employee enters their id. Use this option when the station is sitting in the office and only used to change prices, run reports etc."). Remote servers were well known at the time, as was conducting business over the web. Therefore, this feature would have also been obvious in view of Flashpoint to move a local server to a remote location and shift the system to be web-based.

To the extent CloudofChange contends that Flashpoint fails to disclose or render obvious this feature, a POSITA would have been motivated to modify Flashpoint's system to make the server remote and web-based, as disclosed in the Remote Web Server References. *See* Section VI.A.4 [The Claimed Concepts Were Known, Remote Web Server / Remote Web Server].

Making Flashpoint's server remote and web-based would have realized many benefits. For example, making the server remote would allow a single manager to manage multiple restaurants, reducing hardware costs and simplifying management of multiple restaurants, ensure uniformity across the multiple restaurants, improve the ability to manage from anywhere, and improve the ability to order from anywhere. *See* Section VI.A.4 [The Claimed Concepts Were Known, Remote Web Server / Remote Web Server]. In addition, because web-servers were well-known long before the Asserted Patents, making the remote server web-based would have allowed Flashpoint's system to use standardized hardware and software, or training by using

HTML standard protocol, and lower the cost of POS terminals. Making Flashpoint's server a remote web server would also allow a single server to manage multiple restaurants, reducing hardware costs and simplifying management of multiple restaurants, ensure uniformity across the multiple restaurants, improve the ability to manage from anywhere, and improve the ability to order from anywhere.

Moreover, the ability to add a remote web server, and the benefits of doing so, would have been predictable to a POSITA because it would have been a combination of prior art elements according to known methods to yield predictable results, because a POSITA would have known of the general availability of remote web servers and would have understood that it would have been beneficial and desirable to add a remote web server to a networked computer system, such as a POS system, like the one disclosed in Flashpoint.

Modifying Flashpoint to use a remote web server rather than a local web server would have been a simple substitution of one known element (a local server that is maintained by the individual owner) for another (a remote web server) to obtain predictable results (the benefits of using a remote web server).

Moreover, modifying Flashpoint to use a remote web server would have been obvious to try. When implementing a server, an administrator only has limited options—making the server local or remote, making the server web-based or non-web-based. Each solution has its known advantages and disadvantages, and a POSITA would have understood the benefits and advantages of a remote web server and been motivated to implement a remote web server in order to achieve those benefits and advantages.

Furthermore, using a remote web server was a known approach for computing hardware and software in POS systems, and particularly remote server hardware and software, so a

POSITA would have been motivated to implement a remote web server, as taught in the Remote Web References in the same field, based on design incentives or market forces, including those described in the Remote Web Server References. *See* Section VI.A.4 [The Claimed Concepts Were Known, Remote Web Server / Remote Web Server].

Additionally, a POSITA would have had a reasonable expectation of success in adding this feature to Flashpoint, as Flashpoint already discloses a corporate software that could manage servers remotely. In other words, modifying Flashpoint to use a remote web server would have merely involved the use of a known technique to improve similar systems in the same way.

ii. POS Builder

The '640 patent claims recite providing a POS building tool that allows creating menus with no special training. *See* '640 patent, claim limitation 1[d] ("point of sale builder software that runs on said web server, wherein said local or remote workstations are utilized to build or edit said POS terminals in real time, from anywhere in the world and over the worldwide web"), claim 3 ("The web-based point of sale (POS) builder system of claim 1 wherein use of said builder requires no special training"), claim 8 ("The web-based point of sale (POS) builder system of claim 1 wherein said POS builder adds new POS screens"), claim 9 ("The web-based point of sale (POS) builder system of claim 1 wherein said POS builder edits existing POS screens"), claim 10 ("The web-based point of sale (POS) builder system of claim 1 wherein said POS builder specifies and displays the number, shape and arrangement of selection keys or buttons on said POS screens"), claim 11 ("The web-based point of sale (POS) builder system of claim 1 wherein said POS builder adds items to be associated with said keys or buttons"), claim 12 ("The web-based point of sale (POS) builder system of claim 11 wherein said POS builder assigns attributes such as item name, price, cost, group, taxable and inventory status to said items"), claim 13 ("The web-based point of sale (POS) builder system of claim 12 wherein said

attributes are stored, retrievable and changeable from a back office database stored on said web server"). To the extent that CloudofChange contends that Flashpoint fails to disclose this feature, a POSITA would have been aware of and would have been motivated to modify the systems and methods of Flashpoint to incorporate this feature, or alternatively to combine various teachings from the systems and methods described in one or more of the POS Builder references with Flashpoint into one system or method.

This feature is taught by and/or would have been obvious in view of Flashpoint. For example, Flashpoint relates generally to a POS system that offers "integrated software" including "mission-critical point of sale applications and back office management tools" as well as "enterprise-wide management solutions." Flashpoint, https://web.archive.org/web/20071130010505/http://www.trimpos.net:80/ (Nov. 3, 2007). Flashpoint also discloses allowing a user "to add a new window or edit an existing one" in POS software menus. Flashpoint Operations Guide, Menu Setup Manual, http://www.trimpos.net/Ver16/Docs/Menu%20Setup.pdf (2013), 5. "In the Menu Setup screen," a user "may add new screens, add sales items, delete keys/screens, add modifiers, add call buttons etc." *Id.* at 3. Furthermore, a POS building tool that allows creating and modifying menus with no special training was a well-known feature of POS systems prior to the patent priority date, and was well-known for its ability to reduce implementation and maintenance costs of POS systems by reducing or eliminating the requirement for specialized IT support (e.g., programmers) to set up or maintain a POS system. Accordingly, a POSITA would have been motivated to provide Flashpoint's POS system with a POS builder tool that allows creating and modifying menus with no special training, and would have had a reasonable expectation of success in doing so.

To the extent CloudofChange contends that Flashpoint fails to disclose or render obvious this feature, a POSITA would have been motivated to add a POS builder tool to Flashpoint's system, as disclosed in the POS Builder references. *See* Section VI.A.2 [The Claimed Concepts Were Known, POS Builder References].

Adding a POS builder tool would have realized the many benefits of POS builder tools. For example, adding a POS builder tool would improve Flashpoint's system by making Flashpoint's system—including Flashpoint's POS terminals—easier to install and manage, lowering the cost of installation and maintenance, reducing or eliminating special training to manage (e.g., allowing any employee—rather than just IT—to add or update menus or other aspects of the POS terminals), and lowering IT costs. *See* Section VI.A.2 [The Claimed Concepts Were Known, POS Builder References].

Moreover, the ability to add a POS building tool, and the benefits of doing so, would have been predictable to a POSITA because it would have been a combination of prior art elements according to known methods to yield predictable results, because a POSITA would have known of the general availability of POS builder tools, and would have understood that it would have been beneficial and desirable to add a POS builder tool to a POS system, like the one disclosed in Flashpoint.

Adding a POS builder tool would have been a simple substitution of one known element (a POS system that requires programmers to manually build) for another (a POS system with a POS builder tool that anyone can use without special training) to obtain predictable results (the benefits of a POS builder tool).

Moreover, adding a POS builder tool would have been obvious to try. When implementing a POS system, an administrator only has limited options—configuring menus by

hiring a programmer or other IT professional to manually program them, or configuring menus with a POS builder tool that requires no special training, and thus is usable by any employee, including said manager. Each solution has its known advantages, and a POSITA would have understood the benefits and advantages of a POS builder tool and been motivated to implement a POS builder tool in order to achieve those benefits and advantages.

Furthermore, using a POS builder tool was a known approach for POS systems, so a POSITA would have been motivated to implement the teachings of a POS builder tool in the same field (e.g., the teachings of POS builder tools in the POS Builder References, which are in the same field as Flashpoint) based on design incentives or market forces, including those described in the POS Builder References. *See* Section VI.A.2 [The Claimed Concepts Were Known, POS Builder References].

Additionally, a POSITA would have had a reasonable expectation of success in adding this feature to Flashpoint. Flashpoint's disclosure of these features is in common with a system that includes a POS builder tool that requires no special training. In other words, adding a POS builder tool would have merely involved the use of a known technique to improve similar systems in the same way.

iii. SaaS

The '640 patent claims recite providing a web server as a vendor subscription service (i.e., SaaS). *See* '640 patent, claim 1 limitation 1[e] ("wherein said web servers are provided as a vendor subscription service wherein web server software resides and is hosted on said vendor's remote servers and wherein subscriber company's POS terminals access and repeatedly interact with said web server software from said vendor's remote servers, in order to perform the subscriber's desired terminal function, over a network, wherein the network comprises the Internet."), claim 6 ("The web-based point of sale (POS) builder system of claim 1 wherein said

POS builder is useable to deliver POS application solutions using software as a subscription (SAAS).”). To the extent that CloudofChange contends that Flashpoint fails to disclose this feature, a POSITA would have been aware of and would have been motivated to modify the systems and methods of Flashpoint to incorporate this feature, or alternatively to combine various teachings from the systems and methods described in one or more of the SaaS references with Flashpoint into one system or method.

This feature is taught by and/or would have been obvious in view of Flashpoint. For example, Flashpoint relates generally to a POS system that offers “integrated software” including “mission-critical point of sale applications and back office management tools” as well as “enterprise-wide management solutions.” Flashpoint, https://web.archive.org/web/20071130010505/http://www.trimpos.net:80/ (Nov. 3, 2007). Flashpoint provides a “comprehensive suite of . . . products and services” to “improve all facets of . . . hospitality operations.” Home page, https://web.archive.org/web/20071031033856/http://www.trimpos.net/sitefiles/index.html (Oct. 31, 2007). Furthermore, providing the central server and its software as a shared subscription service was an established business model that was well-known for its ability to reduce subscriber cost and information system complexity. Accordingly, a POSITA would have been motivated to offer Flashpoint’s remote web servers as a subscription service, and would have had a reasonable expectation of success in doing so.

To the extent CloudofChange contends that Flashpoint fails to disclose or render obvious this feature, a POSITA would have been motivated to provide Flashpoint’s remote web servers as a subscription service, as disclosed in the SaaS references. *See* Section VI.A.1 [The Claimed Concepts Were Known, Software as a Subscription / Software as a Service (SaaS)].

Adding SaaS would have realized the many benefits of SaaS. For example, adding SaaS would improve Flashpoint's system, including Flashpoint's remote server and POS terminals, by making Flashpoint's system easier to manage, lowering the cost of maintenance, reducing or eliminating special training to manage, lowering hardware costs, making it easier to update, making it easier to expand to multiple restaurants, making it easier to generate reports, improving the ability to manage from anywhere, and improving the ability to order from anywhere. *See* Section VI.A.1 [The Claimed Concepts Were Known, Software as a Subscription / Software as a Service (SaaS)].

Moreover, adding SaaS would have been predictable to a POSITA because it would have been a combination of prior art elements according to known methods to yield predictable results, because a POSITA would have known of the general availability of SaaS and would have understood that it would have been beneficial and desirable to add SaaS to a networked computer system, such as a POS system, like the one disclosed in Flashpoint.

Adding SaaS would have been a simple substitution of one known element (a local or remote server that is maintained by the individual owner) for another (SaaS, where a SaaS provider maintains a server) to obtain predictable results (the benefits of SaaS).

Moreover, adding SaaS would have been obvious to try. When implementing a server, an administrator only has limited options—making the server local or remote, using individually managed hardware or subscribing to hardware, and using individually managed software or subscribing to software (SaaS). Each solution has its known advantages and disadvantages, and a POSITA would have understood the benefits and advantages of SaaS and been motivated to implement SaaS in order to achieve those benefits and advantages.

Furthermore, SaaS was a known approach for computing hardware and software in POS systems, and particularly remote server hardware and software, so a POSITA would have been motivated to implement the teachings of SaaS in the same field (e.g., the teachings of SaaS in the SaaS references, which are in the same field as Flashpoint) based on design incentives or market forces, including those described in the SaaS references. *See* Section VI.A.1 [The Claimed Concepts Were Known, Software as a Subscription / Software as a Service (SaaS)].

Additionally, a POSITA would have had a reasonable expectation of success in adding this feature to Flashpoint, as Flashpoint already discloses a remote server that provides software services from the remote server and allows for software to be updated on the remote server directly, two features that are common with SaaS. In other words, applying SaaS would have merely involved the use of a known technique to improve similar systems in the same way.

iv. Browser

The ’640 patent claims recite access to point of sale screens with browser-based web access. *See* ’640 patent, claim 14 (“The web-based point of sale (POS) builder system of claim 1, wherein said POS screens are viewable in real-time at remote locations anywhere in the world via personal computers with browser based web access.”).

This feature is taught by and/or would have been obvious in view of Flashpoint. For example, Flashpoint relates generally to a POS system that offers “integrated software” including “mission-critical point of sale applications and back office management tools” as well as “enterprise-wide management solutions.” Flashpoint, https://web.archive.org/web/20071130010505/http://www.trimpos.net:80/ (Nov. 3, 2007). Flashpoint discloses using the web for online ordering. *See* FLASHPOINT™ P.O.S. Software, Flashpoint Operations Guide, System Setup Manual at 84 (“You can use the web button to setup your location for use with online ordering. For detailed information, see the Setting up your

location for online ordering manual."). It would have been obvious to use a browser when ordering on the web.

To the extent CloudofChange contends that Flashpoint fails to disclose or render obvious this feature, a POSITA would have been motivated to modify Flashpoint's system to use a browser, as disclosed in the Browser References. *See* Section VI.A.6 [The Claimed Concepts Were Known, Access to Point of Sale Screens / Browser].

Using Flashpoint's system with a browser would have realized many benefits. For example, using a browser would allow a business to use standard and commoditized software applications, instead of special software or hardware. In addition, browsers enable the use of "thin clients" that do not need to run applications (other than the browser) locally because the processing is largely done at the remote server. Such thin clients would have reduced requirements for memory and processing power, as compared to clients that need to run the applications locally. Browser access to point-of-sale screens would also allow customers to place orders from their own phones, or remotely. *See* Section VI.A.6 [The Claimed Concepts Were Known, Access to Point of Sale Screens / Browser]. In addition, because browsers were well-known long before the Asserted Patents, using a browser would have allowed Flashpoint's system to use standardized hardware and software, or training by using HTML standard protocol, and lower the cost of POS terminals.

Moreover, the ability to use a browser, and the benefits of doing so, would have been predictable to a POSITA because it would have been a combination of prior art elements according to known methods to yield predictable results, because a POSITA would have known of the general availability of browsers and would have understood that it would have been

beneficial and desirable to use browsers to a networked computer system, such as a POS system, like the one disclosed in Flashpoint.

Modifying Flashpoint to use a browser rather than a non-browser program would have been a simple substitution of one known element (a non-browser program) for another (a browser) to obtain predictable results (the benefits of using a browser).

Moreover, modifying Flashpoint to use a browser would have been obvious to try. When implementing a server, an administrator only has limited options—using a non-browser program or a browser. Each solution has its known advantages and disadvantages, and a POSITA would have understood the benefits and advantages of a browser and been motivated to implement a remote web server in order to achieve those benefits and advantages.

Furthermore, using a browser was a known approach for running hardware and software in POS systems, so a POSITA would have been motivated to implement a browser, as taught in the Browser References in the same field, based on design incentives or market forces, including those described in the Browser References. *See* Section VI.A.6 [The Claimed Concepts Were Known, Access to Point of Sale Screens / Browser].

Additionally, a POSITA would have had a reasonable expectation of success in adding this feature to Flashpoint, as Flashpoint already discloses ordering on the web. In other words, modifying Flashpoint to use a browser would have merely involved the use of a known technique to improve similar systems in the same way.

h. Halo alone or in combination with one or more POS Builder References

Halo alone renders obvious each of the features of the ’640 patent, as shown in the attached claim chart. *See* Exhibit A-’640-14.

i. POS Builder

The ’640 patent claims recite providing a POS building tool that allows creating menus with no special training. *See* ’640 patent, claim limitation 1[d] (“point of sale builder software that runs on said web server, wherein said local or remote workstations are utilized to build or edit said POS terminals in real time, from anywhere in the world and over the worldwide web”), claim 3 (“The web-based point of sale (POS) builder system of claim 1 wherein use of said builder requires no special training”), claim 8 (“The web-based point of sale (POS) builder system of claim 1 wherein said POS builder adds new POS screens”), claim 9 (“The web-based point of sale (POS) builder system of claim 1 wherein said POS builder edits existing POS screens”), claim 10 (“The web-based point of sale (POS) builder system of claim 1 wherein said POS builder specifies and displays the number, shape and arrangement of selection keys or buttons on said POS screens”), claim 11 (“The web-based point of sale (POS) builder system of claim 1 wherein said POS builder adds items to be associated with said keys or buttons”), claim 12 (“The web-based point of sale (POS) builder system of claim 11 wherein said POS builder assigns attributes such as item name, price, cost, group, taxable and inventory status to said items”), claim 13 (“The web-based point of sale (POS) builder system of claim 12 wherein said attributes are stored, retrievable and changeable from a back office database stored on said web server”). To the extent that CloudofChange contends that Halo fails to disclose this feature, a POSITA would have been aware of and would have been motivated to modify the systems and methods of Halo to incorporate this feature, or alternatively to combine various teachings from the systems and methods described in one or more of the POS Builder references with Halo into one system or method.

This feature is taught by and/or would have been obvious in view of Halo. For example, Halo relates generally to a POS system that “only requires a POS terminal connected via Internet

to a central computer server." Ahmed H. Medina, *An Analysis of Market Development Strategy of a Point-of-Sale Solutions Provider's Market Research Database* (Summer 2007) (M.B.A. thesis, Simon Fraser University), 3. "Halo Enterprise Manager, which allows the restaurant to configure its POS system by setting menu process and promotions. Restaurants can operate the feature through any web browser." *Id.* "Halo's secure architecture will support Juice It Up! management and franchisees in making strategic decisions based on real-time information about different regions, menu items and store performance, while letting them view reports and change menu items from anywhere at any time using a Web browser." *Juice It Up! Moves to Web-based POS*, NATION'S RESTAURANT NEWS (June 26, 2008), https://www.nrn.com/archive/juice-it-moves-web-based-pos, 1. "With Halo for Restaurants, you have the ability to monitor and manage a single store or your entire chain from anywhere you have access to a web browser. Change prices, add menu items, manage your touch screen terminals and drill down to the transaction level of detail in real time." *Halo For Restaurants*, HALO (Nov. 20, 2006), https://web.archive.org/web/20061120125846/http://www.myhalo.com/section.asp?catid=267, 1. "Halo also integrates multiple locations and enables merchants to compare them in real-time and change items and prices instantly." *Look to the Light: Retail POS System Can Replace ECR,* THE GREEN SHEET ONLINE EDITION, Issue 7:6:1 (June 7, 2007), www.greensheet.com/emagazine.php?article_id=136, 1. Furthermore, a POS building tool that allows creating and modifying menus with no special training was a well-known feature of POS systems prior to the patent priority date, and was well-known for its ability to reduce implementation and maintenance costs of POS systems by reducing or eliminating the requirement for specialized IT support (e.g., programmers) to set up or maintain a POS system. Accordingly, a POSITA would have been motivated to provide Halo's POS system with a POS

builder tool that allows creating and modifying menus with no special training, and would have had a reasonable expectation of success in doing so.

To the extent CloudofChange contends that Halo fails to disclose or render obvious this feature, a POSITA would have been motivated to add a POS builder tool to Halo's system, as disclosed in the POS Builder references. *See* Section VI.A.2 [The Claimed Concepts Were Known, POS Builder References].

Adding a POS builder tool would have realized the many benefits of POS builder tools. For example, adding a POS builder tool would improve Halo's system by making Halo's system—including Halo's POS terminals—easier to install and manage, lowering the cost of installation and maintenance, reducing or eliminating special training to manage (e.g., allowing any employee—rather than just IT—to add or update menus or other aspects of the POS terminals), and lowering IT costs. *See* Section VI.A.2 [The Claimed Concepts Were Known, POS Builder References].

Moreover, the ability to add a POS building tool, and the benefits of doing so, would have been predictable to a POSITA because it would have been a combination of prior art elements according to known methods to yield predictable results, because a POSITA would have known of the general availability of POS builder tools, and would have understood that it would have been beneficial and desirable to add a POS builder tool to a POS system, like the one disclosed in Halo.

Adding a POS builder tool would have been a simple substitution of one known element (a POS system that requires programmers to manually build) for another (a POS system with a POS builder tool that anyone can use without special training) to obtain predictable results (the benefits of a POS builder tool).

Moreover, adding a POS builder tool would have been obvious to try. When implementing a POS system, an administrator only has limited options—configuring menus by hiring a programmer or other IT professional to manually program them, or configuring menus with a POS builder tool that requires no special training, and thus is usable by any employee, including said manager. Each solution has its known advantages, and a POSITA would have understood the benefits and advantages of a POS builder tool and been motivated to implement a POS builder tool in order to achieve those benefits and advantages.

Furthermore, using a POS builder tool was a known approach for POS systems, so a POSITA would have been motivated to implement the teachings of a POS builder tool in the same field (e.g., the teachings of POS builder tools in the POS Builder References, which are in the same field as Halo) based on design incentives or market forces, including those described in the POS Builder References. *See* Section VI.A.2 [The Claimed Concepts Were Known, POS Builder References].

Additionally, a POSITA would have had a reasonable expectation of success in adding this feature to Halo. Halo's disclosure of these features is in common with a system that includes a POS builder tool that requires no special training. In other words, adding a POS builder tool would have merely involved the use of a known technique to improve similar systems in the same way.

i. InfoGenesis alone or in combination with one or more POS Builder References and one or more Browser References

InfoGenesis alone renders obvious each of the features of the '640 patent, as shown in the attached claim chart. *See* Exhibit A-'640-19.

i. POS Builder

The ’640 patent claims recite providing a POS building tool that allows creating menus with no special training. *See* ’640 patent, claim limitation 1[d] (“point of sale builder software that runs on said web server, wherein said local or remote workstations are utilized to build or edit said POS terminals in real time, from anywhere in the world and over the worldwide web”), claim 3 (“The web-based point of sale (POS) builder system of claim 1 wherein use of said builder requires no special training”), claim 8 (“The web-based point of sale (POS) builder system of claim 1 wherein said POS builder adds new POS screens”), claim 9 (“The web-based point of sale (POS) builder system of claim 1 wherein said POS builder edits existing POS screens”), claim 10 (“The web-based point of sale (POS) builder system of claim 1 wherein said POS builder specifies and displays the number, shape and arrangement of selection keys or buttons on said POS screens”), claim 11 (“The web-based point of sale (POS) builder system of claim 1 wherein said POS builder adds items to be associated with said keys or buttons”), claim 12 (“The web-based point of sale (POS) builder system of claim 11 wherein said POS builder assigns attributes such as item name, price, cost, group, taxable and inventory status to said items”), claim 13 (“The web-based point of sale (POS) builder system of claim 12 wherein said attributes are stored, retrievable and changeable from a back office database stored on said web server”). To the extent that CloudofChange contends that InfoGenesis fails to disclose this feature, a POSITA would have been aware of and would have been motivated to modify the systems and methods of InfoGenesis to incorporate this feature, or alternatively to combine various teachings from the systems and methods described in one or more of the POS Builder references with InfoGenesis into one system or method.

This feature is taught by and/or would have been obvious in view of InfoGenesis. For example, InfoGenesis relates generally to “point-of-sale software that combines powerful

reporting and configuration capabilities in the back office with an easy-to-use, touch screen Windows Terminal Application (WTA) that makes better service for frontline staff second nature." InfoGenesis, Point-of-Sale, https://web.archive.org/web/20070416203641/http://www.infogenesis.com/restaurants/pointofsalepos.aspx (Mar. 11, 2006), at 1. InfoGenesis also discloses a "'point and click' interface, drop-down lists, and drill down icons." InfoGenesis, Point-of-Sale, https://web.archive.org/web/20051210182507/http://www.infogenesis.com:80/products/revelation.aspx (Dec. 10, 2005), at 1-2. Furthermore, a POS building tool that allows creating and modifying menus with no special training was a well-known feature of POS systems prior to the patent priority date, and was well-known for its ability to reduce implementation and maintenance costs of POS systems by reducing or eliminating the requirement for specialized IT support (e.g., programmers) to set up or maintain a POS system. Accordingly, a POSITA would have been motivated to provide InfoGenesis's POS system with a POS builder tool that allows creating and modifying menus with no special training, and would have had a reasonable expectation of success in doing so.

To the extent CloudofChange contends that InfoGenesis fails to disclose or render obvious this feature, a POSITA would have been motivated to add a POS builder tool to InfoGenesis's system, as disclosed in the POS Builder references. *See* Section VI.A.2 [The Claimed Concepts Were Known, POS Builder References].

Adding a POS builder tool would have realized the many benefits of POS builder tools. For example, adding a POS builder tool would improve InfoGenesis's system by making InfoGenesis's system—including InfoGenesis's POS terminals—easier to install and manage, lowering the cost of installation and maintenance, reducing or eliminating special training to

manage (e.g., allowing any employee—rather than just IT—to add or update menus or other aspects of the POS terminals), and lowering IT costs. *See* Section VI.A.2 [The Claimed Concepts Were Known, POS Builder References].

Moreover, the ability to add a POS building tool, and the benefits of doing so, would have been predictable to a POSITA because it would have been a combination of prior art elements according to known methods to yield predictable results, because a POSITA would have known of the general availability of POS builder tools, and would have understood that it would have been beneficial and desirable to add a POS builder tool to a POS system, like the one disclosed in InfoGenesis.

Adding a POS builder tool would have been a simple substitution of one known element (a POS system that requires programmers to manually build) for another (a POS system with a POS builder tool that anyone can use without special training) to obtain predictable results (the benefits of a POS builder tool).

Moreover, adding a POS builder tool would have been obvious to try. When implementing a POS system, an administrator only has limited options—configuring menus by hiring a programmer or other IT professional to manually program them, or configuring menus with a POS builder tool that requires no special training, and thus is usable by any employee, including said manager. Each solution has its known advantages, and a POSITA would have understood the benefits and advantages of a POS builder tool and been motivated to implement a POS builder tool in order to achieve those benefits and advantages.

Furthermore, using a POS builder tool was a known approach for POS systems, so a POSITA would have been motivated to implement the teachings of a POS builder tool in the same field (e.g., the teachings of POS builder tools in the POS Builder References, which are in

the same field as InfoGenesis) based on design incentives or market forces, including those described in the POS Builder References. *See* Section VI.A.2 [The Claimed Concepts Were Known, POS Builder References].

Additionally, a POSITA would have had a reasonable expectation of success in adding this feature to InfoGenesis. InfoGenesis's disclosure of these features is in common with a system that includes a POS builder tool that requires no special training. In other words, adding a POS builder tool would have merely involved the use of a known technique to improve similar systems in the same way.

ii. Browser

The '640 patent claims recite access to point of sale screens with browser-based web access. *See* '640 patent, claim 14 ("The web-based point of sale (POS) builder system of claim 1, wherein said POS screens are viewable in real-time at remote locations anywhere in the world via personal computers with browser based web access.").

This feature is taught by and/or would have been obvious in view of InfoGenesis. For example, InfoGenesis relates generally to "point-of-sale software that combines powerful reporting and configuration capabilities in the back office with an easy-to-use, touch screen Windows Terminal Application (WTA) that makes better service for frontline staff second nature." InfoGenesis, Point-of-Sale, https://web.archive.org/web/20070416203641/http://www.infogenesis.com/restaurants/pointofsalepos.aspx (Mar. 11, 2006), at 1. InfoGenesis further discloses "touch screen Windows Terminal Application (WTA)." It would have been obvious to use a browser instead of a WTA, as both were known concepts of how to run a POS system. Furthermore, InfoGenesis discloses "a Web-based solution that [restaurants] employ to track sales, labor and food costs in a real-time environment." William Atkinson, *Point of Service: Now and Later of Restaurant POS* (Apr. 9,

2007). Thus, the InfoGenesis technology already used web-based solutions, which used a browser.

To the extent CloudofChange contends that InfoGenesis fails to disclose or render obvious this feature, a POSITA would have been motivated to modify InfoGenesis's system to use a browser, as disclosed in the Browser References. *See* Section VI.A.6 [The Claimed Concepts Were Known, Access to Point of Sale Screens / Browser].

Using InfoGenesis's system with a browser would have realized many benefits. For example, using a browser would allow a business to use standard and commoditized software applications, instead of special software or hardware. In addition, browsers enable the use of "thin clients" that do not need to run applications (other than the browser) locally because the processing is largely done at the remote server. Such thin clients would have reduced requirements for memory and processing power, as compared to clients that need to run the applications locally. Browser access to point-of-sale screens would also allow customers to place orders from their own phones, or remotely. *See* Section VI.A.6 [The Claimed Concepts Were Known, Access to Point of Sale Screens / Browser]. In addition, because browsers were well-known long before the Asserted Patents, using a browser would have allowed InfoGenesis's system to use standardized hardware and software, or training by using HTML standard protocol, and lower the cost of POS terminals.

Moreover, the ability to use a browser, and the benefits of doing so, would have been predictable to a POSITA because it would have been a combination of prior art elements according to known methods to yield predictable results, because a POSITA would have known of the general availability of browsers and would have understood that it would have been

beneficial and desirable to use browsers to a networked computer system, such as a POS system, like the one disclosed in InfoGenesis.

Modifying InfoGenesis to use a browser rather than a non-browser program would have been a simple substitution of one known element (a non-browser program) for another (a browser) to obtain predictable results (the benefits of using a browser).

Moreover, modifying InfoGenesis to use a browser would have been obvious to try. When implementing a server, an administrator only has limited options—using a non-browser program or a browser. Each solution has its known advantages and disadvantages, and a POSITA would have understood the benefits and advantages of a browser and been motivated to implement a remote web server in order to achieve those benefits and advantages.

Furthermore, using a browser was a known approach for running hardware and software in POS systems, so a POSITA would have been motivated to implement a browser, as taught in the Browser References in the same field, based on design incentives or market forces, including those described in the Browser References. *See* Section VI.A.6 [The Claimed Concepts Were Known, Access to Point of Sale Screens / Browser].

Additionally, a POSITA would have had a reasonable expectation of success in adding this feature to InfoGenesis, as InfoGenesis already discloses a WTA, as well as other web-based solutions. In other words, modifying InfoGenesis to use a browser would have merely involved the use of a known technique to improve similar systems in the same way.

j. Simphony alone or in combination with one or more POS Builder References

Simphony alone renders obvious each of the features of the ’640 patent, as shown in the attached claim chart. *See* Exhibit A-’640-20.

i. POS Builder

The ’640 patent claims recite providing a POS building tool that allows creating menus with no special training. *See* ’640 patent, claim limitation 1[d] (“point of sale builder software that runs on said web server, wherein said local or remote workstations are utilized to build or edit said POS terminals in real time, from anywhere in the world and over the worldwide web”), claim 3 (“The web-based point of sale (POS) builder system of claim 1 wherein use of said builder requires no special training”), claim 8 (“The web-based point of sale (POS) builder system of claim 1 wherein said POS builder adds new POS screens”), claim 9 (“The web-based point of sale (POS) builder system of claim 1 wherein said POS builder edits existing POS screens”), claim 10 (“The web-based point of sale (POS) builder system of claim 1 wherein said POS builder specifies and displays the number, shape and arrangement of selection keys or buttons on said POS screens”), claim 11 (“The web-based point of sale (POS) builder system of claim 1 wherein said POS builder adds items to be associated with said keys or buttons”), claim 12 (“The web-based point of sale (POS) builder system of claim 11 wherein said POS builder assigns attributes such as item name, price, cost, group, taxable and inventory status to said items”), claim 13 (“The web-based point of sale (POS) builder system of claim 12 wherein said attributes are stored, retrievable and changeable from a back office database stored on said web server”). To the extent that CloudofChange contends that Simphony fails to disclose this feature, a POSITA would have been aware of and would have been motivated to modify the systems and methods of Simphony to incorporate this feature, or alternatively to combine various teachings from the systems and methods described in one or more of the POS Builder references with Simphony into one system or method.

This feature is taught by and/or would have been obvious in view of Simphony. For example, Simphony relates generally to a “point-of-sale (POS) hospitality product.” Micros

Simphony TM, https://web.archive.org/web/20071112050945/http://www.micros.com:80/Products/Simphony/ (2007). Simphony also discloses "its ability to scale to a business with thousands of workstations spread over many properties. . . . The Enterprise Management Console allows for the management of the entire system from within a single application. Users are able to create and define the parameters of the enterprise, properties, and revenue centers from any PC that has access to the central server, thus allowing local users to make changes to their configuration if necessary." *Id.* Furthermore, a POS building tool that allows creating and modifying menus with no special training was a well-known feature of POS systems prior to the patent priority date, and was well-known for its ability to reduce implementation and maintenance costs of POS systems by reducing or eliminating the requirement for specialized IT support (e.g., programmers) to set up or maintain a POS system. Accordingly, a POSITA would have been motivated to provide Simphony's POS system with a POS builder tool that allows creating and modifying menus with no special training, and would have had a reasonable expectation of success in doing so.

To the extent CloudofChange contends that Simphony fails to disclose or render obvious this feature, a POSITA would have been motivated to add a POS builder tool to Simphony's system, as disclosed in the POS Builder references. *See* Section VI.A.2 [The Claimed Concepts Were Known, POS Builder References].

Adding a POS builder tool would have realized the many benefits of POS builder tools. For example, adding a POS builder tool would improve Simphony's system by making Simphony's system—including Simphony's POS terminals—easier to install and manage, lowering the cost of installation and maintenance, reducing or eliminating special training to

manage (e.g., allowing any employee—rather than just IT—to add or update menus or other aspects of the POS terminals), and lowering IT costs. *See* Section VI.A.2 [The Claimed Concepts Were Known, POS Builder References].

Moreover, the ability to add a POS building tool, and the benefits of doing so, would have been predictable to a POSITA because it would have been a combination of prior art elements according to known methods to yield predictable results, because a POSITA would have known of the general availability of POS builder tools, and would have understood that it would have been beneficial and desirable to add a POS builder tool to a POS system, like the one disclosed in Simphony.

Adding a POS builder tool would have been a simple substitution of one known element (a POS system that requires programmers to manually build) for another (a POS system with a POS builder tool that anyone can use without special training) to obtain predictable results (the benefits of a POS builder tool).

Moreover, adding a POS builder tool would have been obvious to try. When implementing a POS system, an administrator only has limited options—configuring menus by hiring a programmer or other IT professional to manually program them, or configuring menus with a POS builder tool that requires no special training, and thus is usable by any employee, including said manager. Each solution has its known advantages, and a POSITA would have understood the benefits and advantages of a POS builder tool and been motivated to implement a POS builder tool in order to achieve those benefits and advantages.

Furthermore, using a POS builder tool was a known approach for POS systems, so a POSITA would have been motivated to implement the teachings of a POS builder tool in the same field (e.g., the teachings of POS builder tools in the POS Builder References, which are in

the same field as Simphony) based on design incentives or market forces, including those described in the POS Builder References. *See* Section VI.A.2 [The Claimed Concepts Were Known, POS Builder References].

Additionally, a POSITA would have had a reasonable expectation of success in adding this feature to Simphony. Simphony's disclosure of these features is in common with a system that includes a POS builder tool that requires no special training. In other words, adding a POS builder tool would have merely involved the use of a known technique to improve similar systems in the same way.

k. Squirrel alone or in combination with one or more Remote Manager References and one or more SaaS References and one or more POS Builder References and one or more Browser References

Squirrel alone renders obvious each of the features of the '640 patent, as shown in the attached claim chart. *See* Exhibit A-'640-21.

i. Remote Manager

The '640 patent claims recite a remote manager that can access the remote web server from anywhere. *See* '640 patent, 1[d] ("point of sale builder software that runs on said web server, wherein said local or remote workstations are utilized to build or edit said POS terminals in real time, from anywhere in the world and over the worldwide web").

This feature is taught by and/or would have been obvious in view of Squirrel. For example, Squirrel relates generally to a "Restaurant Point of Sale and Management system." Squirrelsystems, *SquirrelOne Solutions*, https://web.archive.org/web/20051201101248/http://www.squirrelsystems.com:80/solutions/index.html (Dec. 1, 2005). Squirrel discloses that a "Corporate POS manager enables you to control every location's database from your corporate center." Squirrelsystems, *Corporate POS*,

https://web.archive.org/web/20060327174026/http://www.squirrelsystems.com/solutions/corppos.html (Mar. 27, 2006). Squirrel also discloses "local managers" that focus on "local tasks." *Id.* Remote managers were well known at the time, as was conducting business over the web. Therefore, this feature would have also been obvious in view of Squirrel to allow a manager to access the Corporate POS manager from a remote location, which would eliminate the requirement to have managers physically in the office to manage the system.

To the extent CloudofChange contends that Squirrel fails to disclose or render obvious this feature, a POSITA would have been motivated to modify Squirrel's system to use a remote manager, as disclosed in the Remote Manager References. *See* Section VI.A.5 [Remote Manager that Can Access the Remote Web Server From Anywhere / Remote Manager].

Using Squirrel's system with a remote manager would have realized many benefits. For example, using a remote manager would allow a manager to access the server remotely from anywhere in the world allows increased flexibility in corporate operations, because the manager does not have to be physically located in the same place as the server. In addition, it would allow rapid responses to problems that may arise, because a manager can log into the server from home or when on a business trip or vacation, and does not have to come into the office. *See* Section VI.A.5 [Remote Manager that Can Access the Remote Web Server From Anywhere / Remote Manager]. In addition, because having a remote manager was well-known long before the Asserted Patents, using a remote manager would have allowed Squirrel's system to use a single manager, rather than having a local manager at every location, thus lowering costs and providing consistency across businesses.

Moreover, the ability to use a remote manager, and the benefits of doing so, would have been predictable to a POSITA because it would have been a combination of prior art elements

according to known methods to yield predictable results, because a POSITA would have known of the general ability to use a remote manager and would have understood that it would have been beneficial and desirable to use remote managers of a POS system, like the one disclosed in Squirrel.

Modifying Squirrel to use a remote manager rather than a local manager would have been a simple substitution of one known element (a local manager) for another (a remote manager) to obtain predictable results (the benefits of using a remote manager).

Moreover, modifying Squirrel to use a remote manager would have been obvious to try. When implementing a server, an administrator only has limited options—using a local manager or a remote manager. Each solution has its known advantages and disadvantages, and a POSITA would have understood the benefits and advantages of a remote manager and been motivated to use a remote manager in order to achieve those benefits and advantages.

Furthermore, using a remote manager was a known approach for updating content in POS systems, so a POSITA would have been motivated to implement a remote manager, as taught in the Remote Manager References in the same field, based on design incentives or market forces, including those described in the Remote Manager References. *See* Section VI.A.5 [The Claimed Concepts Were Known, Remote Manager that Can Access the Remote Web Server From Anywhere / Remote Manager].

Additionally, a POSITA would have had a reasonable expectation of success in adding this feature to Squirrel, as Squirrel already discloses managing the content of a POS system. In other words, modifying Squirrel to use a remote manager would have merely involved the use of a known technique to improve similar systems in the same way.

ii. SaaS

The ’640 patent claims recite providing a web server as a vendor subscription service (i.e., SaaS). *See* ’640 patent, claim 1 limitation 1[e] (“wherein said web servers are provided as a vendor subscription service wherein web server software resides and is hosted on said vendor's remote servers and wherein subscriber company's POS terminals access and repeatedly interact with said web server software from said vendor's remote servers, in order to perform the subscriber's desired terminal function, over a network, wherein the network comprises the Internet.”), claim 6 (“The web-based point of sale (POS) builder system of claim 1 wherein said POS builder is useable to deliver POS application solutions using software as a subscription (SAAS).”). To the extent that CloudofChange contends that Squirrel fails to disclose this feature, a POSITA would have been aware of and would have been motivated to modify the systems and methods of Squirrel to incorporate this feature, or alternatively to combine various teachings from the systems and methods described in one or more of the SaaS references with Squirrel into one system or method.

This feature is taught by and/or would have been obvious in view of Squirrel. For example, Squirrel relates generally to a “Restaurant Point of Sale and Management system.” Squirrelsystems, *SquirrelOne Solutions*, https://web.archive.org/web/20051201101248/http://www.squirrelsystems.com:80/solutions/index.html (Dec. 1, 2005). Squirrel also discloses “[o]ur passion for your success is evident in every decision factor – in the system itself, but also in the virtually risk-and-obsolescence-free way that we sell it and support you.” *Id.* Furthermore, providing the central server and its software as a shared subscription service was an established business model that was well-known for its ability to reduce subscriber cost and information system complexity. Accordingly, a POSITA would

have been motivated to offer Squirrel's remote web servers as a subscription service, and would have had a reasonable expectation of success in doing so.

To the extent CloudofChange contends that Squirrel fails to disclose or render obvious this feature, a POSITA would have been motivated to provide Squirrel's remote web servers as a subscription service, as disclosed in the SaaS references. *See* Section VI.A.1 [The Claimed Concepts Were Known, Software as a Subscription / Software as a Service (SaaS)].

Adding SaaS would have realized the many benefits of SaaS. For example, adding SaaS would improve Squirrel's system, including Squirrel's remote server and POS terminals, by making Squirrel's system easier to manage, lowering the cost of maintenance, reducing or eliminating special training to manage, lowering hardware costs, making it easier to update, making it easier to expand to multiple restaurants, making it easier to generate reports, improving the ability to manage from anywhere, and improving the ability to order from anywhere. *See* Section VI.A.1 [The Claimed Concepts Were Known, Software as a Subscription / Software as a Service (SaaS)].

Moreover, adding SaaS would have been predictable to a POSITA because it would have been a combination of prior art elements according to known methods to yield predictable results, because a POSITA would have known of the general availability of SaaS and would have understood that it would have been beneficial and desirable to add SaaS to a networked computer system, such as a POS system, like the one disclosed in Squirrel.

Adding SaaS would have been a simple substitution of one known element (a local or remote server that is maintained by the individual owner) for another (SaaS, where a SaaS provider maintains a server) to obtain predictable results (the benefits of SaaS).

Moreover, adding SaaS would have been obvious to try. When implementing a server, an administrator only has limited options—making the server local or remote, using individually managed hardware or subscribing to hardware, and using individually managed software or subscribing to software (SaaS). Each solution has its known advantages and disadvantages, and a POSITA would have understood the benefits and advantages of SaaS and been motivated to implement SaaS in order to achieve those benefits and advantages.

Furthermore, SaaS was a known approach for computing hardware and software in POS systems, and particularly remote server hardware and software, so a POSITA would have been motivated to implement the teachings of SaaS in the same field (e.g., the teachings of SaaS in the SaaS references, which are in the same field as Squirrel) based on design incentives or market forces, including those described in the SaaS references. *See* Section VI.A.1 [The Claimed Concepts Were Known, Software as a Subscription / Software as a Service (SaaS)].

Additionally, a POSITA would have had a reasonable expectation of success in adding this feature to Squirrel, as Squirrel already discloses a remote server that provides software services from the remote server and allows for software to be updated on the remote server directly, two features that are common with SaaS. In other words, applying SaaS would have merely involved the use of a known technique to improve similar systems in the same way.

iii. POS Builder

The ’640 patent claims recite providing a POS building tool that allows creating menus with no special training. *See* ’640 patent, claim limitation 1[d] (“point of sale builder software that runs on said web server, wherein said local or remote workstations are utilized to build or edit said POS terminals in real time, from anywhere in the world and over the worldwide web”), claim 3 (“The web-based point of sale (POS) builder system of claim 1 wherein use of said builder requires no special training”), claim 8 (“The web-based point of sale (POS) builder

system of claim 1 wherein said POS builder adds new POS screens"), claim 9 ("The web-based point of sale (POS) builder system of claim 1 wherein said POS builder edits existing POS screens"), claim 10 ("The web-based point of sale (POS) builder system of claim 1 wherein said POS builder specifies and displays the number, shape and arrangement of selection keys or buttons on said POS screens"), claim 11 ("The web-based point of sale (POS) builder system of claim 1 wherein said POS builder adds items to be associated with said keys or buttons"), claim 12 ("The web-based point of sale (POS) builder system of claim 11 wherein said POS builder assigns attributes such as item name, price, cost, group, taxable and inventory status to said items"), claim 13 ("The web-based point of sale (POS) builder system of claim 12 wherein said attributes are stored, retrievable and changeable from a back office database stored on said web server"). To the extent that CloudofChange contends that Squirrel fails to disclose this feature, a POSITA would have been aware of and would have been motivated to modify the systems and methods of Squirrel to incorporate this feature, or alternatively to combine various teachings from the systems and methods described in one or more of the POS Builder references with Squirrel into one system or method.

This feature is taught by and/or would have been obvious in view of Squirrel. For example, Squirrel relates generally to a "Restaurant Point of Sale and Management system." Squirrelsystems, *SquirrelOne Solutions*, https://web.archive.org/web/20051201101248/http://www.squirrelsystems.com:80/solutions/index.html (Dec. 1, 2005). Squirrel also discloses "SquirrelOne's open architecture, built on a Microsoft SQL relational database, provides on-demand access to real time information. Make menu and price changes instantly, at your convenience. Review and analyze live data from any location. Know what's happening in your operations now, not tomorrow." *Id.* Squirrel

additionally discloses, "Corporate POS Manager facilitates the centralization of menu management, freeing restaurant managers from the burden of making menu changes. New information can be sent to locations at any time and can be automatically incorporated into the local database, whenever it's convenient. . . . You can easily make changes to the menu database at the corporate center, and download them to the restaurants which can in turn perform the up-dates at any time. Or, changes can be pre-scheduled to update automatically (unattended if you prefer)." Squirrelsystems: Corporate POS, https://web.archive.org/web/20060327174026/http://www.squirrelsystems.com/solutions/corppos.html (May 27, 2006). Furthermore, a POS building tool that allows creating and modifying menus with no special training was a well-known feature of POS systems prior to the patent priority date, and was well-known for its ability to reduce implementation and maintenance costs of POS systems by reducing or eliminating the requirement for specialized IT support (e.g., programmers) to set up or maintain a POS system. Accordingly, a POSITA would have been motivated to provide Squirrel's POS system with a POS builder tool that allows creating and modifying menus with no special training, and would have had a reasonable expectation of success in doing so.

To the extent CloudofChange contends that Squirrel fails to disclose or render obvious this feature, a POSITA would have been motivated to add a POS builder tool to Squirrel's system, as disclosed in the POS Builder references. *See* Section VI.A.2 [The Claimed Concepts Were Known, POS Builder References].

Adding a POS builder tool would have realized the many benefits of POS builder tools. For example, adding a POS builder tool would improve Squirrel's system by making Squirrel's system—including Squirrel's POS terminals—easier to install and manage, lowering the cost of

installation and maintenance, reducing or eliminating special training to manage (e.g., allowing any employee—rather than just IT—to add or update menus or other aspects of the POS terminals), and lowering IT costs. *See* Section VI.A.2 [The Claimed Concepts Were Known, POS Builder References].

Moreover, the ability to add a POS building tool, and the benefits of doing so, would have been predictable to a POSITA because it would have been a combination of prior art elements according to known methods to yield predictable results, because a POSITA would have known of the general availability of POS builder tools, and would have understood that it would have been beneficial and desirable to add a POS builder tool to a POS system, like the one disclosed in Squirrel.

Adding a POS builder tool would have been a simple substitution of one known element (a POS system that requires programmers to manually build) for another (a POS system with a POS builder tool that anyone can use without special training) to obtain predictable results (the benefits of a POS builder tool).

Moreover, adding a POS builder tool would have been obvious to try. When implementing a POS system, an administrator only has limited options—configuring menus by hiring a programmer or other IT professional to manually program them, or configuring menus with a POS builder tool that requires no special training, and thus is usable by any employee, including said manager. Each solution has its known advantages, and a POSITA would have understood the benefits and advantages of a POS builder tool and been motivated to implement a POS builder tool in order to achieve those benefits and advantages.

Furthermore, using a POS builder tool was a known approach for POS systems, so a POSITA would have been motivated to implement the teachings of a POS builder tool in the

same field (e.g., the teachings of POS builder tools in the POS Builder References, which are in the same field as Squirrel) based on design incentives or market forces, including those described in the POS Builder References. *See* Section VI.A.2 [The Claimed Concepts Were Known, POS Builder References].

Additionally, a POSITA would have had a reasonable expectation of success in adding this feature to Squirrel. Squirrel's disclosure of these features is in common with a system that includes a POS builder tool that requires no special training. In other words, adding a POS builder tool would have merely involved the use of a known technique to improve similar systems in the same way.

iv. Browser

The '640 patent claims recite access to point of sale screens with browser-based web access. *See* '640 patent, claim 14 ("The web-based point of sale (POS) builder system of claim 1, wherein said POS screens are viewable in real-time at remote locations anywhere in the world via personal computers with browser based web access.").

This feature is taught by and/or would have been obvious in view of Squirrel. For example, Squirrel relates generally to a "Restaurant Point of Sale and Management system." Squirrelsystems, *SquirrelOne Solutions*, https://web.archive.org/web/20051201101248/http://www.squirrelsystems.com:80/solutions/index.html (Dec. 1, 2005). Squirrel further discloses that its POS Manager was "[b]uilt on the latest Microsoft SQL Server relational database" "with a familiar browser style desktop [that] is able to be accessed directly from the central PC or remotely via the internet" Squirrelsystems, *Corporate POS*, https://web.archive.org/web/20060327174026/http://www.squirrelsystems.com/solutions/corppo

s.html (Mar. 27, 2006). It thus would have been obvious to use a browser to run the POS Manager.

To the extent CloudofChange contends that Squirrel fails to disclose or render obvious this feature, a POSITA would have been motivated to modify Squirrel's system to use a browser, as disclosed in the Browser References. *See* Section VI.A.6 [The Claimed Concepts Were Known, Access to Point of Sale Screens / Browser].

Using Squirrel's system with a browser would have realized many benefits. For example, using a browser would allow a business to use standard and commoditized software applications, instead of special software or hardware. In addition, browsers enable the use of "thin clients" that do not need to run applications (other than the browser) locally because the processing is largely done at the remote server. Such thin clients would have reduced requirements for memory and processing power, as compared to clients that need to run the applications locally. Browser access to point-of-sale screens would also allow customers to place orders from their own phones, or remotely. *See* Section VI.A.6 [The Claimed Concepts Were Known, Access to Point of Sale Screens / Browser]. In addition, because browsers were well-known long before the Asserted Patents, using a browser would have allowed Squirrel's system to use standardized hardware and software, or training by using HTML standard protocol, and lower the cost of POS terminals.

Moreover, the ability to use a browser, and the benefits of doing so, would have been predictable to a POSITA because it would have been a combination of prior art elements according to known methods to yield predictable results, because a POSITA would have known of the general availability of browsers and would have understood that it would have been

beneficial and desirable to use browsers to a networked computer system, such as a POS system, like the one disclosed in Squirrel.

Modifying Squirrel to use a browser rather than a non-browser program would have been a simple substitution of one known element (a non-browser program) for another (a browser) to obtain predictable results (the benefits of using a browser).

Moreover, modifying Squirrel to use a browser would have been obvious to try. When implementing a server, an administrator only has limited options—using a non-browser program or a browser. Each solution has its known advantages and disadvantages, and a POSITA would have understood the benefits and advantages of a browser and been motivated to implement a remote web server in order to achieve those benefits and advantages.

Furthermore, using a browser was a known approach for running hardware and software in POS systems, so a POSITA would have been motivated to implement a browser, as taught in the Browser References in the same field, based on design incentives or market forces, including those described in the Browser References. *See* Section VI.A.6 [The Claimed Concepts Were Known, Access to Point of Sale Screens / Browser].

Additionally, a POSITA would have had a reasonable expectation of success in adding this feature to Squirrel, as Squirrel already discloses a browser style desktop that is able to be accessed remotely via the Internet. In other words, modifying Squirrel to use a browser would have merely involved the use of a known technique to improve similar systems in the same way.

2. ’012 Patent

Prior art references rendering the asserted claims of the ’012 patent obvious, alone or in combination with other references, are discussed below and included in Exhibits B-’012-1 to B-’012-21. Further reasons to combine the references identified in Exhibits B-’012-1 to B-’012-21 include the nature of the problem being solved, the express, implied and inherent teachings of the

prior art, the knowledge of persons of ordinary skill in the art, that such combinations would have yielded predictable results, and that such combinations would have represented known alternatives to a POSITA.

Illustrative obviousness combinations are listed below (it should be understood that each listed combination can also include the knowledge of a POSITA). The Asserted Claims of the '012 patent are obvious over:

- Manno alone or in combination with one or more POS GUI References
- Tengler alone or in combination with one or more POS GUI References and one or more Browser References
- Cusack alone or in combination with one or more POS GUI References
- Cusack alone or in combination with one or more POS GUI References and one or more Independent Terminals References
- Olson alone or in combination with one or more POS GUI References
- Lukkari alone or in combination with one or more POS GUI References and one or more Independent Terminals References
- Flashpoint alone or in combination with one or more Remote Web Server References and one or more Browser References
- Halo alone or in combination with one or more Independent Terminals References
- NextPOS alone or in combination with one or more Remote Web Server References and one or more Browser References and one or more Independent Terminals References

- Point of Success alone or in combination with one or more Remote Web Server References and one or more Browser References and one or more Independent Terminals References
- InfoGenesis alone or in combination with one or more POS GUI References
- Squirrel alone or in combination with one or more POS GUI References and one or more Remote Web Server References and one or more Browser References and one or more Independent Terminals References

a. Manno alone or in combination with one or more POS GUI References

Manno alone renders obvious each of the features of the ’012 patent, as shown in the attached claim chart. *See* Exhibit B-’012-2.

i. POS GUI

The ’012 patent claims recite a POS building tool that allows creating the graphical user interface of a menu with no special training. *See* ’012 patent, claim 6 (“The web-based point of sale (POS) builder system of claim 1, wherein said POS builder software is configured to provide instructions to add new POS terminals”), claim 7 (“The web-based point of sale (POS) builder system of claim 1, wherein said POS builder software is configured to provide instructions to modify existing POS terminals”), claim 10 (“The web-based point of sale (POS) builder system of claim 1, wherein a number, shape and arrangement of selection keys or buttons for said one or more POS terminals are specified from the POS builder interface”), claim 11 (“The web-based point of sale (POS) builder system of claim 10, wherein items are associated with said selection keys or buttons from the POS builder interface”). To the extent that CloudofChange contends that Manno fails to disclose this feature, a POSITA would have been aware of and would have been motivated to modify the systems and methods of Manno to incorporate this feature, or

alternatively to combine various teachings from the systems and methods described in one or more of the POS GUI references with Manno into one system or method.

This feature is taught by and/or would have been obvious in view of Manno. For example, Manno relates generally to "[a] network or web-based point of sale (WPOS) system [that] employs a store server and point-of-sale client computers that are connected over a store LAN or over the Internet to the server. The server provides all necessary operating drivers for cash drawer, scanners, and other point-of-sale peripherals via the web browser of the client computers. An instant kiosk retail solution can involve a remote computer terminal installed via the Internet." *See* Manno, Abstract. Mano also discloses, "[t]he WPOS provides a single, centralized management point for store web pages, and for in-house operations. Store management can change prices, look up stock quantities, check customer databases, in real time from any web-connected computer or PDA. Customers or store clerks can print invoices over the Internet, and customers can check appointments and can even be notified via text messaging to PDA or cell phone when a scheduled job is completed." Manno, [0020]. Manno additionally discloses, "[b]ecause the WPOS system is web-based, management can set up terminals anywhere in the world and have real-time access to the store's database, and can provide the same look and feel as actually being at the store. A kiosk can be set up at a local mall without the need to install any software, and when there is an equipment failure at the point-of-sale, another computer can be substituted to act as the POS terminal, without loss of any sales data. The constant communication over the Internet (or other network) makes the system ideal for any mobile retailer." Manno, [0021]. Manno further discloses, "WPOS can provide staff online training, as determined from data mining results of management back end. Buying patterns of customers, either in-store sales or Internet sales, can be provided easily. Also staff can view real-

time store performance via Internet or wireless PDA or cell phone. Mobile retailers (e.g., installers and technicians) can keep instant track of job status and requests. An integrated online scheduler allows for customers to change times and dates of scheduled job orders. Store database and web database can be shared, allowing for management to make changes in store quantities, prices, etc., with such changes being updated instantly at the stores and at the store's web site. Also, management can select which products are sold via Internet, at the store, or both, at the time the product is entered into stock." Manno, [0029]. Furthermore, providing a POS building tool that allows creating the graphical user interface of a menu with no special training was a common feature in POS systems prior to the patent priority date and was well-known for its ability to reduce implementation and maintenance costs of POS systems, by avoiding requirements for specialized IT support (e.g., programmers) to set up or maintain the graphical user interface of a POS system menu. Accordingly, a POSITA would have been motivated to provide Manno's POS system with a POS building tool that allows creating the graphical user interface of a menu with no special training, and would have had a reasonable expectation of success in doing so.

To the extent CloudofChange contends that Manno fails to disclose or render obvious this feature, a POSITA would have been motivated to add a POS building tool that allows creating the graphical user interface of a menu with no special training to Manno's system, as disclosed in the POS GUI references. *See* Section VI.A.3 [The Claimed Concepts Were Known, POS GUI References].

Adding a POS building tool that allows creating the graphical user interface of a menu with no special training would have realized the many benefits of POS GUIs. For example, adding a POS builder tool that allows creating the graphical user interface of a menu with no

special training would improve Manno's system by making Manno's system—including Manno's POS terminals—easier to install and manage, lowering the cost of installation and maintenance, reducing or eliminating special training to manage (e.g., allowing any employee—rather than just IT—to add or update graphical user interfaces of menus or of other aspects of the POS terminals), and lowering IT costs. *See* Section VI.A.3 [The Claimed Concepts Were Known, POS GUI References].

Moreover, the ability to add a POS building tool that allows creating the graphical user interface of a menu with no special training, and the benefits of doing so, would have been predictable to a POSITA because it would have been a combination of prior art elements according to known methods to yield predictable results, because a POSITA would have known of the general availability of POS builder tools, the general availability of graphical user interfaces, and the general availability of easy-to-use tools for building graphical user interfaces, and would have understood that it would have been beneficial and desirable to add a POS building tool that allows creating the graphical user interface of a menu with no special training to a POS system, like the one disclosed in Manno.

Adding a POS building tool that allows creating the graphical user interface of a menu with no special training would have been a simple substitution of one known element (a POS system that requires programmers to manually build a user interface of a menu) for another (a POS system with a POS building tool that allows creating the graphical user interface of a menu with no special training) to obtain predictable results (the benefits of a POS building tool that allows creating the graphical user interface of a menu with no special training).

Moreover, adding a POS building tool that allows creating the graphical user interface of a menu with no special training would have been obvious to try. When implementing a POS

system, an administrator only has limited options—configuring menu interfaces by hiring a programmer or other IT professional to manually program them, or configuring graphical user interfaces of menus with a POS building tool that allows creating the graphical user interface of a menu with no special training, and thus is usable by any employee, including said manager. Each solution has its known advantages, and a POSITA would have understood the benefits and advantages of a POS building tool that allows creating the graphical user interface of a menu with no special training and been motivated to implement a POS building tool that allows creating the graphical user interface of a menu with no special training in order to achieve those benefits and advantages.

Furthermore, using a POS building tool that allows creating the graphical user interface of a menu with no special training was a known approach for POS systems, so a POSITA would have been motivated to implement the teachings of a POS building tool that allows creating the graphical user interface of a menu with no special training as described in the POS GUI References, which are in the same field as Manno, based on design incentives or market forces, including those described in the POS GUI References. *See* Section VI.A.3 [The Claimed Concepts Were Known, POS GUI References].

Additionally, a POSITA would have had a reasonable expectation of success in adding this feature to Manno, as Manno already discloses that with its POS system, "management [is able] to make changes in store quantities, prices, etc., with such changes being updated instantly at the stores and at the store's web site. Also, management can select which products are sold via Internet, at the store, or both . . . ." Manno, [0029]. Manno's disclosure of these features is in common with a system that includes a POS building tool that allows creating the graphical user interface of a menu with no special training. In other words, adding a POS building tool that

allows creating the graphical user interface of a menu with no special training would have merely involved the use of a known technique to improve similar systems in the same way.

b. Tengler alone or in combination with one or more POS GUI References and one or more Browser References

Tengler alone renders obvious each of the features of the ’012 patent, as shown in the attached claim chart. *See* Exhibit B-’012-4.

i. POS GUI

The ’012 patent claims recite a POS building tool that allows creating the graphical user interface of a menu with no special training. *See* ’012 patent, claim 6 (“The web-based point of sale (POS) builder system of claim 1, wherein said POS builder software is configured to provide instructions to add new POS terminals”), claim 7 (“The web-based point of sale (POS) builder system of claim 1, wherein said POS builder software is configured to provide instructions to modify existing POS terminals”), claim 10 (“The web-based point of sale (POS) builder system of claim 1, wherein a number, shape and arrangement of selection keys or buttons for said one or more POS terminals are specified from the POS builder interface”), claim 11 (“The web-based point of sale (POS) builder system of claim 10, wherein items are associated with said selection keys or buttons from the POS builder interface”). To the extent that CloudofChange contends that Tengler fails to disclose this feature, a POSITA would have been aware of and would have been motivated to modify the systems and methods of Tengler to incorporate this feature, or alternatively to combine various teachings from the systems and methods described in one or more of the POS GUI references with Tengler into one system or method.

This feature is taught by and/or would have been obvious in view of Tengler. For example, Tengler relates generally to “an apparatus that includes stations in a quick-serve commercial establishment, the stations enabling users to enter orders, check the status of orders,

and assemble orders for delivery, a network interconnecting the stations, each of the stations comprising storage for data about the orders, and software to cause copies of the data to be cached at multiple ones of the stations." Tengler, [0011]. Tengler also discloses "[a] manager station 70 allows a manager of the restaurant 2 to view food production and service activity in the restaurant. In some examples, a manager station 70 displays a graphical user interface 190, as illustrated in FIG. 7. The manager station 70 can use a broadband data line 72 to link the manager to the Internet. Referring to FIGS. 7A and 7B, managers can view information about the multiple restaurants 2 using interfaces 192 and 196 over the Internet. These managers can also modify menus and change prices using interfaces 192 and 196. An interface 192 shows a recipe for coffee with options at a particular restaurant 2. A button 194 enables a manager to modify the recipe. An interface 196 shows hot beverages, cold beverages, and shakes available at a particular restaurant 2. The manager's interface is linked directly to the database in the restaurant (without the intervention of any intermediary database or process), so that information presented to the manager is derived directly from the database of the restaurant and changes made by the manager through the interface are committed directly to the database in the restaurant." Tengler, [0073]. Furthermore, providing a POS building tool that allows creating the graphical user interface of a menu with no special training was a common feature in POS systems prior to the patent priority date and was well-known for its ability to reduce implementation and maintenance costs of POS systems, by avoiding requirements for specialized IT support (e.g., programmers) to set up or maintain the graphical user interface of a POS system menu. Accordingly, a POSITA would have been motivated to provide Tengler's POS system with a POS building tool that allows creating the graphical user interface of a menu with no special training, and would have had a reasonable expectation of success in doing so.

To the extent CloudofChange contends that Tengler fails to disclose or render obvious this feature, a POSITA would have been motivated to add a POS building tool that allows creating the graphical user interface of a menu with no special training to Tengler's system, as disclosed in the POS GUI references. *See* Section VI.A.3 [The Claimed Concepts Were Known, POS GUI References].

Adding a POS building tool that allows creating the graphical user interface of a menu with no special training would have realized the many benefits of POS GUIs. For example, adding a POS builder tool that allows creating the graphical user interface of a menu with no special training would improve Tengler's system by making Tengler's system—including Tengler's POS terminals—easier to install and manage, lowering the cost of installation and maintenance, reducing or eliminating special training to manage (e.g., allowing any employee—rather than just IT—to add or update graphical user interfaces of menus or of other aspects of the POS terminals), and lowering IT costs. *See* Section VI.A.3 [The Claimed Concepts Were Known, POS GUI References].

Moreover, the ability to add a POS building tool that allows creating the graphical user interface of a menu with no special training, and the benefits of doing so, would have been predictable to a POSITA because it would have been a combination of prior art elements according to known methods to yield predictable results, because a POSITA would have known of the general availability of POS builder tools, the general availability of graphical user interfaces, and the general availability of easy-to-use tools for building graphical user interfaces, and would have understood that it would have been beneficial and desirable to add a POS building tool that allows creating the graphical user interface of a menu with no special training to a POS system, like the one disclosed in Tengler.

Adding a POS building tool that allows creating the graphical user interface of a menu with no special training would have been a simple substitution of one known element (a POS system that requires programmers to manually build a user interface of a menu) for another (a POS system with a POS building tool that allows creating the graphical user interface of a menu with no special training) to obtain predictable results (the benefits of a POS building tool that allows creating the graphical user interface of a menu with no special training).

Moreover, adding a POS building tool that allows creating the graphical user interface of a menu with no special training would have been obvious to try. When implementing a POS system, an administrator only has limited options—configuring menu interfaces by hiring a programmer or other IT professional to manually program them, or configuring graphical user interfaces of menus with a POS building tool that allows creating the graphical user interface of a menu with no special training, and thus is usable by any employee, including said manager. Each solution has its known advantages, and a POSITA would have understood the benefits and advantages of a POS building tool that allows creating the graphical user interface of a menu with no special training and been motivated to implement a POS building tool that allows creating the graphical user interface of a menu with no special training in order to achieve those benefits and advantages.

Furthermore, using a POS building tool that allows creating the graphical user interface of a menu with no special training was a known approach for POS systems, so a POSITA would have been motivated to implement the teachings of a POS building tool that allows creating the graphical user interface of a menu with no special training as described in the POS GUI References, which are in the same field as Tengler, based on design incentives or market forces,

including those described in the POS GUI References. *See* Section VI.A.3 [The Claimed Concepts Were Known, POS GUI References].

Additionally, a POSITA would have had a reasonable expectation of success in adding this feature to Tengler. Tengler's disclosure of these features is in common with a system that includes a POS building tool that allows creating the graphical user interface of a menu with no special training. In other words, adding a POS building tool that allows creating the graphical user interface of a menu with no special training would have merely involved the use of a known technique to improve similar systems in the same way.

ii. Browser

The '012 patent claims recite access to point of sale screens with browser-based web access. *See* claim 8 ("The web-based point of sale (POS) builder system of claim 1, wherein said one or more POS terminals are configured to be accessible by a web browser.").

This feature is taught by and/or would have been obvious in view of Tengler. For example, Tengler relates generally to "an apparatus that includes stations in a quick-serve commercial establishment, the stations enabling users to enter orders, check the status of orders, and assemble orders for delivery, a network interconnecting the stations, each of the stations comprising storage for data about the orders, and software to cause copies of the data to be cached at multiple ones of the stations." Tengler, [0011]. Tengler further discloses a "manager station" that can use a "broadband data line" to "link the manager to the Internet." Tengler, [0073]. Tengler discloses that "managers can view information about the multiple restaurants" using interfaces "over the Internet." *Id.* Tengler also discloses that, while "a main server computer" "runs the system server software," "the database," "the store-resident web server software," "the production display software," and "the drive-through display service," there is a

"computer" that "runs a web browser." Tengler, [0016]. Therefore, it would have been obvious to use a browser for the POS system.

To the extent CloudofChange contends that Tengler fails to disclose or render obvious this feature, a POSITA would have been motivated to modify Tengler's system to use a browser, as disclosed in the Browser References. *See* Section VI.A.6 [The Claimed Concepts Were Known, Access to Point of Sale Screens / Browser].

Using Tengler's system with a browser would have realized many benefits. For example, using a browser would allow a business to use standard and commoditized software applications, instead of special software or hardware. In addition, browsers enable the use of "thin clients" that do not need to run applications (other than the browser) locally because the processing is largely done at the remote server. Such thin clients would have reduced requirements for memory and processing power, as compared to clients that need to run the applications locally. Browser access to point-of-sale screens would also allow customers to place orders from their own phones, or remotely. *See* Section VI.A.6 [The Claimed Concepts Were Known, Access to Point of Sale Screens / Browser]. In addition, because browsers were well-known long before the Asserted Patents, using a browser would have allowed Tengler's system to use standardized hardware and software, or training by using HTML standard protocol, and lower the cost of POS terminals.

Moreover, the ability to use a browser, and the benefits of doing so, would have been predictable to a POSITA because it would have been a combination of prior art elements according to known methods to yield predictable results, because a POSITA would have known of the general availability of browsers and would have understood that it would have been

beneficial and desirable to use browsers to a networked computer system, such as a POS system, like the one disclosed in Tengler.

Modifying Tengler to use a browser rather than a non-browser program would have been a simple substitution of one known element (a non-browser program) for another (a browser) to obtain predictable results (the benefits of using a browser).

Moreover, modifying Tengler to use a browser would have been obvious to try. When implementing a server, an administrator only has limited options—using a non-browser program or a browser. Each solution has its known advantages and disadvantages, and a POSITA would have understood the benefits and advantages of a browser and been motivated to implement a remote web server in order to achieve those benefits and advantages.

Furthermore, using a browser was a known approach for running hardware and software in POS systems, so a POSITA would have been motivated to implement a browser, as taught in the Browser References in the same field, based on design incentives or market forces, including those described in the Browser References. *See* Section VI.A.6 [The Claimed Concepts Were Known, Access to Point of Sale Screens / Browser].

Additionally, a POSITA would have had a reasonable expectation of success in adding this feature to Tengler, as Tengler already discloses running a POS system over the internet, and computers using web browsers. In other words, modifying Tengler to use a browser would have merely involved the use of a known technique to improve similar systems in the same way.

c. Cusack alone or in combination with one or more POS GUI References

Cusack alone renders obvious each of the features of the ’012 patent, as shown in the attached claim chart. *See* Exhibit B-’012-1.

i. POS GUI

The ’012 patent claims recite a POS building tool that allows creating the graphical user interface of a menu with no special training. *See* ’012 patent, claim 6 (“The web-based point of sale (POS) builder system of claim 1, wherein said POS builder software is configured to provide instructions to add new POS terminals”), claim 7 (“The web-based point of sale (POS) builder system of claim 1, wherein said POS builder software is configured to provide instructions to modify existing POS terminals”), claim 10 (“The web-based point of sale (POS) builder system of claim 1, wherein a number, shape and arrangement of selection keys or buttons for said one or more POS terminals are specified from the POS builder interface”), claim 11 (“The web-based point of sale (POS) builder system of claim 10, wherein items are associated with said selection keys or buttons from the POS builder interface”). To the extent that CloudofChange contends that Cusack fails to disclose this feature, a POSITA would have been aware of and would have been motivated to modify the systems and methods of Cusack to incorporate this feature, or alternatively to combine various teachings from the systems and methods described in one or more of the POS GUI references with Cusack into one system or method.

This feature is taught by and/or would have been obvious in view of Cusack. For example, Cusack relates generally to a “[s]ystem and method for facilitating real-time, web based point of sale operations such as sales, reporting, etc. The system and method include and involve a central database management facility storing and managing data related to at least one user interface form operable within a browser application and to items which may be sold or otherwise processed at a point of sale. Also included and involved is a plurality of client data processing facilities which are coupled to the central database management facility via an electronic data network. Each client data processing system is located at a remote point of sale and is coupled to the central database management facility via a network connection over the

electronic data network. Each client data processing facility is configured to execute the browser application, to access the central database management facility via the network connection to retrieve and process at least one user interface form within the browser application, to access the central database management facility in real time to retrieve data related to at least one of the items to facilitate a remote point of sale related operation, and to cause the central database management facility to manage the data based on the remote point of sale related operation." Cusack, Abstract. Cusack also discloses "[u]ser interfaces such as that shown in FIG. 3A can be designed using common web design techniques, such as those used to design and implement HTML or Active Server Pages, and will be readily understood by those having ordinary skill in the art. Moreover, although the present invention is described in terms of HTML documents and the like, the present invention is not limited such designs, and a user interface may be designed using other methods and techniques that enable an application interface to be accessible via a web browser and perform the functionality described herein. It will also be appreciated that the interface shown here is only exemplary, and a user interface may be designed to be more or less user-friendly and to provide more or less functionality as a company's business practices may require." Cusack, [0042]. Furthermore, providing a POS building tool that allows creating the graphical user interface of a menu with no special training was a common feature in POS systems prior to the patent priority date and was well-known for its ability to reduce implementation and maintenance costs of POS systems, by avoiding requirements for specialized IT support (e.g., programmers) to set up or maintain the graphical user interface of a POS system menu. Accordingly, a POSITA would have been motivated to provide Cusack's POS system with a POS building tool that allows creating the graphical user interface of a menu with no special training, and would have had a reasonable expectation of success in doing so.

To the extent CloudofChange contends that Cusack fails to disclose or render obvious this feature, a POSITA would have been motivated to add a POS building tool that allows creating the graphical user interface of a menu with no special training to Cusack's system, as disclosed in the POS GUI references. *See* Section VI.A.3 [The Claimed Concepts Were Known, POS GUI References].

Adding a POS building tool that allows creating the graphical user interface of a menu with no special training would have realized the many benefits of POS GUIs. For example, adding a POS builder tool that allows creating the graphical user interface of a menu with no special training would improve Cusack's system by making Cusack's system—including Cusack's POS terminals—easier to install and manage, lowering the cost of installation and maintenance, reducing or eliminating special training to manage (e.g., allowing any employee—rather than just IT—to add or update graphical user interfaces of menus or of other aspects of the POS terminals), and lowering IT costs. *See* Section VI.A.3 [The Claimed Concepts Were Known, POS GUI References].

Moreover, the ability to add a POS building tool that allows creating the graphical user interface of a menu with no special training, and the benefits of doing so, would have been predictable to a POSITA because it would have been a combination of prior art elements according to known methods to yield predictable results, because a POSITA would have known of the general availability of POS builder tools, the general availability of graphical user interfaces, and the general availability of easy-to-use tools for building graphical user interfaces, and would have understood that it would have been beneficial and desirable to add a POS building tool that allows creating the graphical user interface of a menu with no special training to a POS system, like the one disclosed in Cusack.

Adding a POS building tool that allows creating the graphical user interface of a menu with no special training would have been a simple substitution of one known element (a POS system that requires programmers to manually build a user interface of a menu) for another (a POS system with a POS building tool that allows creating the graphical user interface of a menu with no special training) to obtain predictable results (the benefits of a POS building tool that allows creating the graphical user interface of a menu with no special training).

Moreover, adding a POS building tool that allows creating the graphical user interface of a menu with no special training would have been obvious to try. When implementing a POS system, an administrator only has limited options—configuring menu interfaces by hiring a programmer or other IT professional to manually program them, or configuring graphical user interfaces of menus with a POS building tool that allows creating the graphical user interface of a menu with no special training, and thus is usable by any employee, including said manager. Each solution has its known advantages, and a POSITA would have understood the benefits and advantages of a POS building tool that allows creating the graphical user interface of a menu with no special training and been motivated to implement a POS building tool that allows creating the graphical user interface of a menu with no special training in order to achieve those benefits and advantages.

Furthermore, using a POS building tool that allows creating the graphical user interface of a menu with no special training was a known approach for POS systems, so a POSITA would have been motivated to implement the teachings of a POS building tool that allows creating the graphical user interface of a menu with no special training as described in the POS GUI References, which are in the same field as Cusack, based on design incentives or market forces,

including those described in the POS GUI References. *See* Section VI.A.3 [The Claimed Concepts Were Known, POS GUI References].

Additionally, a POSITA would have had a reasonable expectation of success in adding this feature to Cusack. Cusack's disclosure of these features is in common with a system that includes a POS building tool that allows creating the graphical user interface of a menu with no special training. In other words, adding a POS building tool that allows creating the graphical user interface of a menu with no special training would have merely involved the use of a known technique to improve similar systems in the same way.

d. Cusack alone or in combination with one or more POS GUI References and one or more Independent Terminals References

Cusack alone renders obvious each of the features of the '012 patent, as shown in the attached claim chart. *See* Exhibit B-'012-1.

i. POS GUI

The '012 patent claims recite a POS building tool that allows creating the graphical user interface of a menu with no special training. *See* '012 patent, claim 6 ("The web-based point of sale (POS) builder system of claim 1, wherein said POS builder software is configured to provide instructions to add new POS terminals"), claim 7 ("The web-based point of sale (POS) builder system of claim 1, wherein said POS builder software is configured to provide instructions to modify existing POS terminals"), claim 10 ("The web-based point of sale (POS) builder system of claim 1, wherein a number, shape and arrangement of selection keys or buttons for said one or more POS terminals are specified from the POS builder interface"), claim 11 ("The web-based point of sale (POS) builder system of claim 10, wherein items are associated with said selection keys or buttons from the POS builder interface"). To the extent that CloudofChange contends that Cusack fails to disclose this feature, a POSITA would have been aware of and would have

been motivated to modify the systems and methods of Cusack to incorporate this feature, or alternatively to combine various teachings from the systems and methods described in one or more of the POS GUI references with Cusack into one system or method.

This feature is taught by and/or would have been obvious in view of Cusack. For example, Cusack relates generally to a "[s]ystem and method for facilitating real-time, web based point of sale operations such as sales, reporting, etc. The system and method include and involve a central database management facility storing and managing data related to at least one user interface form operable within a browser application and to items which may be sold or otherwise processed at a point of sale. Also included and involved is a plurality of client data processing facilities which are coupled to the central database management facility via an electronic data network. Each client data processing system is located at a remote point of sale and is coupled to the central database management facility via a network connection over the electronic data network. Each client data processing facility is configured to execute the browser application, to access the central database management facility via the network connection to retrieve and process at least one user interface form within the browser application, to access the central database management facility in real time to retrieve data related to at least one of the items to facilitate a remote point of sale related operation, and to cause the central database management facility to manage the data based on the remote point of sale related operation." Cusack, Abstract. Cusack also discloses "[u]ser interfaces such as that shown in FIG. 3A can be designed using common web design techniques, such as those used to design and implement HTML or Active Server Pages, and will be readily understood by those having ordinary skill in the art. Moreover, although the present invention is described in terms of HTML documents and the like, the present invention is not limited such designs, and a user interface may be designed

using other methods and techniques that enable an application interface to be accessible via a web browser and perform the functionality described herein. It will also be appreciated that the interface shown here is only exemplary, and a user interface may be designed to be more or less user-friendly and to provide more or less functionality as a company's business practices may require." Cusack, [0042]. Furthermore, providing a POS building tool that allows creating the graphical user interface of a menu with no special training was a common feature in POS systems prior to the patent priority date and was well-known for its ability to reduce implementation and maintenance costs of POS systems, by avoiding requirements for specialized IT support (e.g., programmers) to set up or maintain the graphical user interface of a POS system menu. Accordingly, a POSITA would have been motivated to provide Cusack's POS system with a POS building tool that allows creating the graphical user interface of a menu with no special training, and would have had a reasonable expectation of success in doing so.

To the extent CloudofChange contends that Cusack fails to disclose or render obvious this feature, a POSITA would have been motivated to add a POS building tool that allows creating the graphical user interface of a menu with no special training to Cusack's system, as disclosed in the POS GUI references. *See* Section VI.A.3 [The Claimed Concepts Were Known, POS GUI References].

Adding a POS building tool that allows creating the graphical user interface of a menu with no special training would have realized the many benefits of POS GUIs. For example, adding a POS builder tool that allows creating the graphical user interface of a menu with no special training would improve Cusack's system by making Cusack's system—including Cusack's POS terminals—easier to install and manage, lowering the cost of installation and maintenance, reducing or eliminating special training to manage (e.g., allowing any employee—

rather than just IT—to add or update graphical user interfaces of menus or of other aspects of the POS terminals), and lowering IT costs. *See* Section VI.A.3 [The Claimed Concepts Were Known, POS GUI References].

Moreover, the ability to add a POS building tool that allows creating the graphical user interface of a menu with no special training, and the benefits of doing so, would have been predictable to a POSITA because it would have been a combination of prior art elements according to known methods to yield predictable results, because a POSITA would have known of the general availability of POS builder tools, the general availability of graphical user interfaces, and the general availability of easy-to-use tools for building graphical user interfaces, and would have understood that it would have been beneficial and desirable to add a POS building tool that allows creating the graphical user interface of a menu with no special training to a POS system, like the one disclosed in Cusack.

Adding a POS building tool that allows creating the graphical user interface of a menu with no special training would have been a simple substitution of one known element (a POS system that requires programmers to manually build a user interface of a menu) for another (a POS system with a POS building tool that allows creating the graphical user interface of a menu with no special training) to obtain predictable results (the benefits of a POS building tool that allows creating the graphical user interface of a menu with no special training).

Moreover, adding a POS building tool that allows creating the graphical user interface of a menu with no special training would have been obvious to try. When implementing a POS system, an administrator only has limited options—configuring menu interfaces by hiring a programmer or other IT professional to manually program them, or configuring graphical user interfaces of menus with a POS building tool that allows creating the graphical user interface of a

menu with no special training, and thus is usable by any employee, including said manager. Each solution has its known advantages, and a POSITA would have understood the benefits and advantages of a POS building tool that allows creating the graphical user interface of a menu with no special training and been motivated to implement a POS building tool that allows creating the graphical user interface of a menu with no special training in order to achieve those benefits and advantages.

Furthermore, using a POS building tool that allows creating the graphical user interface of a menu with no special training was a known approach for POS systems, so a POSITA would have been motivated to implement the teachings of a POS building tool that allows creating the graphical user interface of a menu with no special training as described in the POS GUI References, which are in the same field as Cusack, based on design incentives or market forces, including those described in the POS GUI References. *See* Section VI.A.3 [The Claimed Concepts Were Known, POS GUI References].

Additionally, a POSITA would have had a reasonable expectation of success in adding this feature to Cusack. Cusack's disclosure of these features is in common with a system that includes a POS building tool that allows creating the graphical user interface of a menu with no special training. In other words, adding a POS building tool that allows creating the graphical user interface of a menu with no special training would have merely involved the use of a known technique to improve similar systems in the same way.

ii. Independent Terminals

Claim 9 of the '012 patent recites the ability of the POS terminals to continue functioning—at least temporarily—even if they lose their connection to the server. *See* '012 patent, claim 9 ("The web-based point of sale (POS) builder system of claim 1, wherein said one or more POS terminals, once generated, are configured to operate independently of said web

server.”). To the extent that CloudofChange contends that Cusack fails to disclose this feature, a POSITA would have been aware of and would have been motivated to modify the systems and methods of Cusack to incorporate this feature, or alternatively to combine various teachings from the systems and methods described in one or more of the Independent Terminals references with Cusack into one system or method.

This feature is taught by and/or would have been obvious in view of Cusack. For example, Cusack relates generally to a “[s]ystem and method for facilitating real-time, web based point of sale operations such as sales, reporting, etc. The system and method include and involve a central database management facility storing and managing data related to at least one user interface form operable within a browser application and to items which may be sold or otherwise processed at a point of sale. Also included and involved is a plurality of client data processing facilities which are coupled to the central database management facility via an electronic data network. Each client data processing system is located at a remote point of sale and is coupled to the central database management facility via a network connection over the electronic data network. Each client data processing facility is configured to execute the browser application, to access the central database management facility via the network connection to retrieve and process at least one user interface form within the browser application, to access the central database management facility in real time to retrieve data related to at least one of the items to facilitate a remote point of sale related operation, and to cause the central database management facility to manage the data based on the remote point of sale related operation.” Cusack, Abstract. Cusack also discloses “[e]ach POS terminal is capable of running an entire POS system in accordance with the present invention.” Cusack, [0028]. Cusack explains that it was known in the prior art to have “a POS network containing a plurality of POS terminals 104

linked together via a LAN, such as one incorporating an Ethernet backbone. Each POS terminal contains data relating to the items (goods, services, etc.) to be sold in the store, and is configured to perform standard POS functions, such as executing purchases (cash, credit card, etc.), enabling coupon redemption, providing returns and refunds, performing credit card transactions, engaging in credit card settlement transactions, permitting cash/drawer transactions, etc. During business hours, POS terminals are the center of business transactions, storing transaction data locally as transactions (e.g., sales, returns, etc.) occur. After business hours, a store manager, for example, may place the POS systems into a reconciliation mode to reconcile the day's business across the POS network." Cusack, [0007]. Furthermore, providing the ability of the POS terminals to continue functioning—at least temporarily—even if they lose their connection to the server was an established feature of POS systems prior to the patent priority date that was well-known for its ability to reduce the costs of downtime for POS systems, by ensuring that restaurants, retail locations, or other businesses implementing POS systems would be able to rely on their chosen POS system to continue to enter transactions and/or take orders even if a network connection or server was down. Accordingly, a POSITA would have been motivated to provide Cusack's POS system with the ability of the POS terminals to continue functioning—at least temporarily—even if they lose their connection to the server, and would have had a reasonable expectation of success in doing so.

To the extent CloudofChange contends that Cusack fails to disclose or render obvious this feature, a POSITA would have been motivated to add the ability of the POS terminals to continue functioning—at least temporarily—even if they lose their connection to the server, as disclosed in the Independent Terminals references. *See* Section VI.A.7 [The Claimed Concepts Were Known, Independent Terminals References].

Modifying Cusack to enable its POS terminals to continue operating independently would have realized many benefits. For example, adding independent terminals would improve Cusack's system by making Cusack's system—including Cusack's POS terminals—more reliable, increasing revenue (e.g., because of reduced downtime), and making the system work faster (e.g., because menu processing is done locally instead of over a network). *See* Section VI.A.7 [The Claimed Concepts Were Known, Independent Terminals References].

Moreover, the ability to add independent terminals, and the benefits of doing so, would have been predictable to a POSITA because it would have been a combination of prior art elements according to known methods to yield predictable results, because a POSITA would have known of the ability to make POS terminals operate independently from servers (e.g., like all POS terminals pre-dating the Internet), of the general availability of independent terminals, and would have understood that it would have been beneficial and desirable to add independent terminals to a POS system, like the one disclosed in Cusack.

Adding independent terminals would have been a simple substitution of one known element (a POS system with POS terminals that do not have the ability to operate independently of a server) for another (a POS system with POS terminals that have the ability to operate independently of a server) to obtain predictable results (the benefits of POS terminals that can continue to operate even if a connection to a server is lost).

Moreover, adding independent terminals would have been obvious to try. A POSITA implementing a POS system would have known that before the advent of computer networking and the Internet, POS terminals were independent terminals (e.g., cash register), and so POS terminals could be operated without a connection to a server. Moreover, a POSITA would have understood that most or all networks sometimes experience connectivity issues, and similarly

servers sometimes experience downtime, so it would have been obvious to try implementing a failsafe option to allow POS terminals to operate independently of a server, so that the POS terminals could be used even if the network or server is temporarily inaccessible or otherwise having problems. Implementing independent terminals has known advantages, and a POSITA would have understood the benefits and advantages of independent terminals and been motivated to implement independent terminals to achieve those benefits and advantages.

Furthermore, using independent terminals was a known approach for POS systems, so a POSITA would have been motivated to implement the teachings of independent terminals in the same field (e.g., the teachings of independent terminals in the Independent Terminals References, which are in the same field as Cusack) based on design incentives or market forces, including those described in the Independent Terminals References. *See* Section VI.A.7 [The Claimed Concepts Were Known, Independent Terminals References].

Additionally, a POSITA would have had a reasonable expectation of success in adding this feature to Cusack, as Cusack already discloses that it was known in the prior art to use local POS terminals. *See, e.g.*, Cusack, [0006] ("Typically, data is stored locally on each POS network or within each POS terminal device."). Cusack's disclosure of these features is in common with a system that includes independent terminals. In other words, adding the ability for POS terminals to operate offline—e.g., temporarily in case of a network or server outage—would have merely involved the use of a known technique to improve similar systems in the same way.

e. Olson alone or in combination with one or more POS GUI References

Olson alone renders obvious each of the features of the '012 patent, as shown in the attached claim chart. *See* Exhibit B-'012-8.

i. POS GUI

The ’012 patent claims recite a POS building tool that allows creating the graphical user interface of a menu with no special training. *See* ’012 patent, claim 6 (“The web-based point of sale (POS) builder system of claim 1, wherein said POS builder software is configured to provide instructions to add new POS terminals”), claim 7 (“The web-based point of sale (POS) builder system of claim 1, wherein said POS builder software is configured to provide instructions to modify existing POS terminals”), claim 10 (“The web-based point of sale (POS) builder system of claim 1, wherein a number, shape and arrangement of selection keys or buttons for said one or more POS terminals are specified from the POS builder interface”), claim 11 (“The web-based point of sale (POS) builder system of claim 10, wherein items are associated with said selection keys or buttons from the POS builder interface”). To the extent that CloudofChange contends that Olson fails to disclose this feature, a POSITA would have been aware of and would have been motivated to modify the systems and methods of Olson to incorporate this feature, or alternatively to combine various teachings from the systems and methods described in one or more of the POS GUI references with Olson into one system or method.

This feature is taught by and/or would have been obvious in view of Olson. For example, Olson relates generally to “the managing of the points of sale (POS) with Web-based back-office systems. POS transaction engine terminals are implemented on personal computer, PC hardware using PC software with a relational database. The transaction engine architecture provides cost-effective application customization. Web-based back-office software is implemented on servers. POS and BO systems are monitored and supported via the web using protocol and open software components.” Olson, Abstract. Olson also discloses a “[back-office] interface requests update from the [back-office] such as new items, price changes, employees, etc. The POS terminals communicate via HTTP protocol (hypertext transfer protocol) 23 with Back-office BO software,

which is implemented on web servers 24, which can be located anywhere in the world." Olson, [0019]. Olson further discloses "[a] client who is the store manager or owner selects a promotion type, associates it with an item, department, etc, then sets the parameters that control how that promotion works. These transaction transmissions between the POS and the BO can be encrypted to insure privacy and security." Olson, [0020]. Olson also discloses "[t]he core transaction engine interfaces with configurable modules 30, which handle the meaning & processing steps of terminal keys required. For example, one business or one store may require a different function performed by the transaction engine when the 'Total' key on the POS terminal is pressed. A convenience store may have a different way of computing the summation of taxable items than a restaurant business. These configurable modules allow the POS terminal to be very flexible for use by many different types of stores and businesses." Olson, [0021]. Furthermore, providing a POS building tool that allows creating the graphical user interface of a menu with no special training was a common feature in POS systems prior to the patent priority date and was well-known for its ability to reduce implementation and maintenance costs of POS systems, by avoiding requirements for specialized IT support (e.g., programmers) to set up or maintain the graphical user interface of a POS system menu. Accordingly, a POSITA would have been motivated to provide Olson's POS system with a POS building tool that allows creating the graphical user interface of a menu with no special training, and would have had a reasonable expectation of success in doing so.

To the extent CloudofChange contends that Olson fails to disclose or render obvious this feature, a POSITA would have been motivated to add a POS building tool that allows creating the graphical user interface of a menu with no special training to Olson's system, as disclosed in

the POS GUI references. *See* Section VI.A.3 [The Claimed Concepts Were Known, POS GUI References].

Adding a POS building tool that allows creating the graphical user interface of a menu with no special training would have realized the many benefits of POS GUIs. For example, adding a POS builder tool that allows creating the graphical user interface of a menu with no special training would improve Olson's system by making Olson's system—including Olson's POS terminals—easier to install and manage, lowering the cost of installation and maintenance, reducing or eliminating special training to manage (e.g., allowing any employee—rather than just IT—to add or update graphical user interfaces of menus or of other aspects of the POS terminals), and lowering IT costs. *See* Section VI.A.3 [The Claimed Concepts Were Known, POS GUI References].

Moreover, the ability to add a POS building tool that allows creating the graphical user interface of a menu with no special training, and the benefits of doing so, would have been predictable to a POSITA because it would have been a combination of prior art elements according to known methods to yield predictable results, because a POSITA would have known of the general availability of POS builder tools, the general availability of graphical user interfaces, and the general availability of easy-to-use tools for building graphical user interfaces, and would have understood that it would have been beneficial and desirable to add a POS building tool that allows creating the graphical user interface of a menu with no special training to a POS system, like the one disclosed in Olson.

Adding a POS building tool that allows creating the graphical user interface of a menu with no special training would have been a simple substitution of one known element (a POS system that requires programmers to manually build a user interface of a menu) for another (a

POS system with a POS building tool that allows creating the graphical user interface of a menu with no special training) to obtain predictable results (the benefits of a POS building tool that allows creating the graphical user interface of a menu with no special training).

Moreover, adding a POS building tool that allows creating the graphical user interface of a menu with no special training would have been obvious to try. When implementing a POS system, an administrator only has limited options—configuring menu interfaces by hiring a programmer or other IT professional to manually program them, or configuring graphical user interfaces of menus with a POS building tool that allows creating the graphical user interface of a menu with no special training, and thus is usable by any employee, including said manager. Each solution has its known advantages, and a POSITA would have understood the benefits and advantages of a POS building tool that allows creating the graphical user interface of a menu with no special training and been motivated to implement a POS building tool that allows creating the graphical user interface of a menu with no special training in order to achieve those benefits and advantages.

Furthermore, using a POS building tool that allows creating the graphical user interface of a menu with no special training was a known approach for POS systems, so a POSITA would have been motivated to implement the teachings of a POS building tool that allows creating the graphical user interface of a menu with no special training as described in the POS GUI References, which are in the same field as Olson, based on design incentives or market forces, including those described in the POS GUI References. *See* Section VI.A.3 [The Claimed Concepts Were Known, POS GUI References].

Additionally, a POSITA would have had a reasonable expectation of success in adding this feature to Olson. Olson's disclosure of these features is in common with a system that

includes a POS building tool that allows creating the graphical user interface of a menu with no special training. In other words, adding a POS building tool that allows creating the graphical user interface of a menu with no special training would have merely involved the use of a known technique to improve similar systems in the same way.

f. Lukkari alone or in combination with one or more POS GUI References and one or more Independent Terminals References

Lukkari alone renders obvious each of the features of the ’012 patent, as shown in the attached claim chart. *See* Exhibit B-’012-11.

i. POS GUI

The ’012 patent claims recite a POS building tool that allows creating the graphical user interface of a menu with no special training. *See* ’012 patent, claim 6 (“The web-based point of sale (POS) builder system of claim 1, wherein said POS builder software is configured to provide instructions to add new POS terminals”), claim 7 (“The web-based point of sale (POS) builder system of claim 1, wherein said POS builder software is configured to provide instructions to modify existing POS terminals”), claim 10 (“The web-based point of sale (POS) builder system of claim 1, wherein a number, shape and arrangement of selection keys or buttons for said one or more POS terminals are specified from the POS builder interface”), claim 11 (“The web-based point of sale (POS) builder system of claim 10, wherein items are associated with said selection keys or buttons from the POS builder interface”). To the extent that CloudofChange contends that Lukkari fails to disclose this feature, a POSITA would have been aware of and would have been motivated to modify the systems and methods of Lukkari to incorporate this feature, or alternatively to combine various teachings from the systems and methods described in one or more of the POS GUI references with Lukkari into one system or method.

This feature is taught by and/or would have been obvious in view of Lukkari. For example, Lukkari relates generally to "SmartRestaurant service, which allows customers to use mobile devices for ordering and paying lunches from a nearby campus restaurant beforehand. Further, SmartRestaurant provides the restaurant with means of adjusting the sales with production capacity and prior knowledge of upcoming orders." Lukkari, 1. Lukkari also discloses "SmartRestaurant's run-time view is presented in Fig. 2(a). rotuaari.net is a multi-processor host that runs the SmartRestaurant service and other server components: Apache web server, Tomcat application server and MySQL database." *Id.* at 3. Furthermore, providing a POS building tool that allows creating the graphical user interface of a menu with no special training was a common feature in POS systems prior to the patent priority date and was well-known for its ability to reduce implementation and maintenance costs of POS systems, by avoiding requirements for specialized IT support (e.g., programmers) to set up or maintain the graphical user interface of a POS system menu. Accordingly, a POSITA would have been motivated to provide Lukkari's POS system with a POS building tool that allows creating the graphical user interface of a menu with no special training, and would have had a reasonable expectation of success in doing so.

To the extent CloudofChange contends that Lukkari fails to disclose or render obvious this feature, a POSITA would have been motivated to add a POS building tool that allows creating the graphical user interface of a menu with no special training to Lukkari's system, as disclosed in the POS GUI references. *See* Section VI.A.3 [The Claimed Concepts Were Known, POS GUI References].

Adding a POS building tool that allows creating the graphical user interface of a menu with no special training would have realized the many benefits of POS GUIs. For example,

adding a POS builder tool that allows creating the graphical user interface of a menu with no special training would improve Lukkari's system by making Lukkari's system—including Lukkari's POS terminals—easier to install and manage, lowering the cost of installation and maintenance, reducing or eliminating special training to manage (e.g., allowing any employee—rather than just IT—to add or update graphical user interfaces of menus or of other aspects of the POS terminals), and lowering IT costs. *See* Section VI.A.3 [The Claimed Concepts Were Known, POS GUI References].

Moreover, the ability to add a POS building tool that allows creating the graphical user interface of a menu with no special training, and the benefits of doing so, would have been predictable to a POSITA because it would have been a combination of prior art elements according to known methods to yield predictable results, because a POSITA would have known of the general availability of POS builder tools, the general availability of graphical user interfaces, and the general availability of easy-to-use tools for building graphical user interfaces, and would have understood that it would have been beneficial and desirable to add a POS building tool that allows creating the graphical user interface of a menu with no special training to a POS system, like the one disclosed in Lukkari.

Adding a POS building tool that allows creating the graphical user interface of a menu with no special training would have been a simple substitution of one known element (a POS system that requires programmers to manually build a user interface of a menu) for another (a POS system with a POS building tool that allows creating the graphical user interface of a menu with no special training) to obtain predictable results (the benefits of a POS building tool that allows creating the graphical user interface of a menu with no special training).

Moreover, adding a POS building tool that allows creating the graphical user interface of a menu with no special training would have been obvious to try. When implementing a POS system, an administrator only has limited options—configuring menu interfaces by hiring a programmer or other IT professional to manually program them, or configuring graphical user interfaces of menus with a POS building tool that allows creating the graphical user interface of a menu with no special training, and thus is usable by any employee, including said manager. Each solution has its known advantages, and a POSITA would have understood the benefits and advantages of a POS building tool that allows creating the graphical user interface of a menu with no special training and been motivated to implement a POS building tool that allows creating the graphical user interface of a menu with no special training in order to achieve those benefits and advantages.

Furthermore, using a POS building tool that allows creating the graphical user interface of a menu with no special training was a known approach for POS systems, so a POSITA would have been motivated to implement the teachings of a POS building tool that allows creating the graphical user interface of a menu with no special training as described in the POS GUI References, which are in the same field as Lukkari, based on design incentives or market forces, including those described in the POS GUI References. *See* Section VI.A.3 [The Claimed Concepts Were Known, POS GUI References].

Additionally, a POSITA would have had a reasonable expectation of success in adding this feature to Lukkari. Lukkari's disclosure of these features is in common with a system that includes a POS building tool that allows creating the graphical user interface of a menu with no special training. In other words, adding a POS building tool that allows creating the graphical

user interface of a menu with no special training would have merely involved the use of a known technique to improve similar systems in the same way.

ii. Independent Terminals

Claim 9 of the ’012 patent recites the ability of the POS terminals to continue functioning—at least temporarily—even if they lose their connection to the server. *See* ’012 patent, claim 9 (“The web-based point of sale (POS) builder system of claim 1, wherein said one or more POS terminals, once generated, are configured to operate independently of said web server.”). To the extent that CloudofChange contends that Lukkari fails to disclose this feature, a POSITA would have been aware of and would have been motivated to modify the systems and methods of Lukkari to incorporate this feature, or alternatively to combine various teachings from the systems and methods described in one or more of the Independent Terminals references with Lukkari into one system or method.

This feature is taught by and/or would have been obvious in view of Lukkari. For example, Lukkari relates generally to “SmartRestaurant service, which allows customers to use mobile devices for ordering and paying lunches from a nearby campus restaurant beforehand. Further, SmartRestaurant provides the restaurant with means of adjusting the sales with production capacity and prior knowledge of upcoming orders.” Lukkari, 1. Lukkari also discloses “SmartRestaurant’s run-time view is presented in Fig. 2(a). rotuaari.net is a multi-processor host that runs the SmartRestaurant service and other server components: Apache web server, Tomcat application server and MySQL database.” *Id.* at 3. Furthermore, providing the ability of the POS terminals to continue functioning—at least temporarily—even if they lose their connection to the server was an established feature of POS systems prior to the patent priority date that was well-known for its ability to reduce the costs of downtime for POS systems, by ensuring that restaurants, retail locations, or other businesses implementing POS

systems would be able to rely on their chosen POS system to continue to enter transactions and/or take orders even if a network connection or server was down. Accordingly, a POSITA would have been motivated to provide Lukkari's POS system with the ability of the POS terminals to continue functioning—at least temporarily—even if they lose their connection to the server, and would have had a reasonable expectation of success in doing so.

To the extent CloudofChange contends that Lukkari fails to disclose or render obvious this feature, a POSITA would have been motivated to add the ability of the POS terminals to continue functioning—at least temporarily—even if they lose their connection to the server, as disclosed in the Independent Terminals references. *See* Section VI.A.7 [The Claimed Concepts Were Known, Independent Terminals References].

Modifying Lukkari to enable its POS terminals to continue operating independently would have realized many benefits. For example, adding independent terminals would improve Lukkari's system by making Lukkari's system—including Lukkari's POS terminals—more reliable, increasing revenue (e.g., because of reduced downtime), and making the system work faster (e.g., because menu processing is done locally instead of over a network). *See* Section VI.A.7 [The Claimed Concepts Were Known, Independent Terminals References].

Moreover, the ability to add independent terminals, and the benefits of doing so, would have been predictable to a POSITA because it would have been a combination of prior art elements according to known methods to yield predictable results, because a POSITA would have known of the ability to make POS terminals operate independently from servers (e.g., like all POS terminals pre-dating the Internet), of the general availability of independent terminals, and would have understood that it would have been beneficial and desirable to add independent terminals to a POS system, like the one disclosed in Lukkari.

Adding independent terminals would have been a simple substitution of one known element (a POS system with POS terminals that do not have the ability to operate independently of a server) for another (a POS system with POS terminals that have the ability to operate independently of a server) to obtain predictable results (the benefits of POS terminals that can continue to operate even if a connection to a server is lost).

Moreover, adding independent terminals would have been obvious to try. A POSITA implementing a POS system would have known that before the advent of computer networking and the Internet, POS terminals were independent terminals (e.g., cash register), and so POS terminals could be operated without a connection to a server. Moreover, a POSITA would have understood that most or all networks sometimes experience connectivity issues, and similarly servers sometimes experience downtime, so it would have been obvious to try implementing a failsafe option to allow POS terminals to operate independently of a server, so that the POS terminals could be used even if the network or server is temporarily inaccessible or otherwise having problems. Implementing independent terminals has known advantages, and a POSITA would have understood the benefits and advantages of independent terminals and been motivated to implement independent terminals to achieve those benefits and advantages.

Furthermore, using independent terminals was a known approach for POS systems, so a POSITA would have been motivated to implement the teachings of independent terminals in the same field (e.g., the teachings of independent terminals in the Independent Terminals References, which are in the same field as Lukkari) based on design incentives or market forces, including those described in the Independent Terminals References. *See* Section VI.A.7 [The Claimed Concepts Were Known, Independent Terminals References].

Additionally, a POSITA would have had a reasonable expectation of success in adding this feature to Lukkari. Lukkari's disclosure of these features is in common with a system that includes independent terminals. In other words, adding the ability for POS terminals to operate offline—e.g., temporarily in case of a network or server outage—would have merely involved the use of a known technique to improve similar systems in the same way.

g. Flashpoint alone or in combination with one or more Remote Web Server References and one or more Browser References

Flashpoint alone renders obvious each of the features of the '012 patent, as shown in the attached claim chart. *See* Exhibit B-'012-13.

i. Remote Web Server

The '012 patent claims recite a web server that is remote from one or more point of sale terminals. *See* '012 patent, 1[a] ("a web server including POS builder software installed thereon;"), 1[d] ("wherein said POS builder interface is configured to be utilized to access said POS builder software for programmatically creating or modifying said one or more POS terminals in real time over the communications network, wherein said POS builder software is configured to interact with said one or more POS terminals over the communications network in order for the web-based point of sale (POS) builder system to perform functions in accordance with instructions sent from the POS builder interface;"), 1[e] (wherein said POS transactions and corresponding transaction data from said one or more POS terminals are configured to be transmitted to said web server via the communications network;").

This feature is taught by and/or would have been obvious in view of Flashpoint. For example, Flashpoint relates generally to a POS system that offers "integrated software" including "mission-critical point of sale applications and back office management tools" as well as "enterprise-wide management solutions." Flashpoint,

https://web.archive.org/web/20071130010505/http://www.trimpos.net:80/ (Nov. 3, 2007). Restaurants using Flashpoint could use "corporate software" on a remote server at the corporate office to update menus on the remote server. *See* FLASHPOINT™ P.O.S. Software, Flashpoint Operations Guide, System Setup Manual at 4 ("Once the System Set Up button has been selected the following screen will appear, allowing you to complete many of the setups for Flashpoint. If you are using the corporate software, you must select the location to setup before going into system setup."). Flashpoint also discloses a local server computer. *See, e.g.*, Flashpoint, *Quick Food Service*, https://web.archive.org/web/20071031033856/http:/www.trimpos.net/sitefiles/index.html (Oct. 31, 2007) ("Use the back office station for confidential reporting and daily management needs, allowing the POS stations to be dedicated to order processing."); FLASHPOINT™ P.O.S. Software, Flashpoint Operations Guide, System Setup Manual at 48 ("Back Office Station – If set, the station will automatically display the system menu once the employee enters their id. Use this option when the station is sitting in the office and only used to change prices, run reports etc."). Remote servers were well known at the time, as was conducting business over the web. Therefore, this feature would have also been obvious in view of Flashpoint to move a local server to a remote location and shift the system to be web-based.

To the extent CloudofChange contends that Flashpoint fails to disclose or render obvious this feature, a POSITA would have been motivated to modify Flashpoint's system to make the server remote and web-based, as disclosed in the Remote Web Server References. *See* Section VI.A.4 [The Claimed Concepts Were Known, Remote Web Server / Remote Web Server].

Making Flashpoint's server remote and web-based would have realized many benefits. For example, making the server remote would allow a single manager to manage multiple

restaurants, reducing hardware costs and simplifying management of multiple restaurants, ensure uniformity across the multiple restaurants, improve the ability to manage from anywhere, and improve the ability to order from anywhere. *See* Section VI.A.4 [The Claimed Concepts Were Known, Remote Web Server / Remote Web Server]. In addition, because web-servers were well-known long before the Asserted Patents, making the remote server web-based would have allowed Flashpoint's system to use standardized hardware and software, or training by using HTML standard protocol, and lower the cost of POS terminals. Making Flashpoint's server a remote web server would also allow a single server to manage multiple restaurants, reducing hardware costs and simplifying management of multiple restaurants, ensure uniformity across the multiple restaurants, improve the ability to manage from anywhere, and improve the ability to order from anywhere.

Moreover, the ability to add a remote web server, and the benefits of doing so, would have been predictable to a POSITA because it would have been a combination of prior art elements according to known methods to yield predictable results, because a POSITA would have known of the general availability of remote web servers and would have understood that it would have been beneficial and desirable to add a remote web server to a networked computer system, such as a POS system, like the one disclosed in Flashpoint.

Modifying Flashpoint to use a remote web server rather than a local web server would have been a simple substitution of one known element (a local server that is maintained by the individual owner) for another (a remote web server) to obtain predictable results (the benefits of using a remote web server).

Moreover, modifying Flashpoint to use a remote web server would have been obvious to try. When implementing a server, an administrator only has limited options—making the server

local or remote, making the server web-based or non-web-based. Each solution has its known advantages and disadvantages, and a POSITA would have understood the benefits and advantages of a remote web server and been motivated to implement a remote web server in order to achieve those benefits and advantages.

Furthermore, using a remote web server was a known approach for computing hardware and software in POS systems, and particularly remote server hardware and software, so a POSITA would have been motivated to implement a remote web server, as taught in the Remote Web References in the same field, based on design incentives or market forces, including those described in the Remote Web Server References. *See* Section VI.A.4 [The Claimed Concepts Were Known, Remote Web Server / Remote Web Server].

Additionally, a POSITA would have had a reasonable expectation of success in adding this feature to Flashpoint, as Flashpoint already discloses a corporate software that could manage servers remotely. In other words, modifying Flashpoint to use a remote web server would have merely involved the use of a known technique to improve similar systems in the same way.

ii. Browser

The ’012 patent claims recite access to point of sale screens with browser-based web access. *See* claim 8 (“The web-based point of sale (POS) builder system of claim 1, wherein said one or more POS terminals are configured to be accessible by a web browser.”).

This feature is taught by and/or would have been obvious in view of Flashpoint. For example, Flashpoint relates generally to a POS system that offers “integrated software” including “mission-critical point of sale applications and back office management tools” as well as “enterprise-wide management solutions.” Flashpoint, https://web.archive.org/web/20071130010505/http://www.trimpos.net:80/ (Nov. 3, 2007). Flashpoint discloses using the web for online ordering. *See* FLASHPOINT™ P.O.S. Software,

Flashpoint Operations Guide, System Setup Manual at 84 ("You can use the web button to setup your location for use with online ordering. For detailed information, see the Setting up your location for online ordering manual."). It would have been obvious to use a browser when ordering on the web.

To the extent CloudofChange contends that Flashpoint fails to disclose or render obvious this feature, a POSITA would have been motivated to modify Flashpoint's system to use a browser, as disclosed in the Browser References. *See* Section VI.A.6 [The Claimed Concepts Were Known, Access to Point of Sale Screens / Browser].

Using Flashpoint's system with a browser would have realized many benefits. For example, using a browser would allow a business to use standard and commoditized software applications, instead of special software or hardware. In addition, browsers enable the use of "thin clients" that do not need to run applications (other than the browser) locally because the processing is largely done at the remote server. Such thin clients would have reduced requirements for memory and processing power, as compared to clients that need to run the applications locally. Browser access to point-of-sale screens would also allow customers to place orders from their own phones, or remotely. *See* Section VI.A.6 [The Claimed Concepts Were Known, Access to Point of Sale Screens / Browser]. In addition, because browsers were well-known long before the Asserted Patents, using a browser would have allowed Flashpoint's system to use standardized hardware and software, or training by using HTML standard protocol, and lower the cost of POS terminals.

Moreover, the ability to use a browser, and the benefits of doing so, would have been predictable to a POSITA because it would have been a combination of prior art elements according to known methods to yield predictable results, because a POSITA would have known

of the general availability of browsers and would have understood that it would have been beneficial and desirable to use browsers to a networked computer system, such as a POS system, like the one disclosed in Flashpoint.

Modifying Flashpoint to use a browser rather than a non-browser program would have been a simple substitution of one known element (a non-browser program) for another (a browser) to obtain predictable results (the benefits of using a browser).

Moreover, modifying Flashpoint to use a browser would have been obvious to try. When implementing a server, an administrator only has limited options—using a non-browser program or a browser. Each solution has its known advantages and disadvantages, and a POSITA would have understood the benefits and advantages of a browser and been motivated to implement a remote web server in order to achieve those benefits and advantages.

Furthermore, using a browser was a known approach for running hardware and software in POS systems, so a POSITA would have been motivated to implement a browser, as taught in the Browser References in the same field, based on design incentives or market forces, including those described in the Browser References. *See* Section VI.A.6 [The Claimed Concepts Were Known, Access to Point of Sale Screens / Browser].

Additionally, a POSITA would have had a reasonable expectation of success in adding this feature to Flashpoint, as Flashpoint already discloses ordering on the web. In other words, modifying Flashpoint to use a browser would have merely involved the use of a known technique to improve similar systems in the same way.

h. Halo alone or in combination with one or more Independent Terminals References

Halo alone renders obvious each of the features of the ’012 patent, as shown in the attached claim chart. *See* Exhibit B-’012-14.

i. Independent Terminals

Claim 9 of the ’012 patent recites the ability of the POS terminals to continue functioning—at least temporarily—even if they lose their connection to the server. *See* ’012 patent, claim 9 (“The web-based point of sale (POS) builder system of claim 1, wherein said one or more POS terminals, once generated, are configured to operate independently of said web server.”). To the extent that CloudofChange contends that Halo fails to disclose this feature, a POSITA would have been aware of and would have been motivated to modify the systems and methods of Halo to incorporate this feature, or alternatively to combine various teachings from the systems and methods described in one or more of the Independent Terminals references with Halo into one system or method.

This feature is taught by and/or would have been obvious in view of Halo. For example, Halo relates generally to a POS system that “only requires a POS terminal connected via Internet to a central computer server.” Ahmed H. Medina, *An Analysis of Market Development Strategy of a Point-of-Sale Solutions Provider’s Market Research Database* (Summer 2007) (M.B.A. thesis, Simon Fraser University), 3. Halo also discloses “Halo does not require any computer server at the restaurant location to host the POS software; instead, the touch screen terminal connects via the public Internet to a central application server located at Vivonet’s office. In case of a problem with the Internet connection, the Halo terminal at the customer site can switch to offline mode and process POS transactions locally until the connection is restored.” *Id.* Halo additionally discloses: “2. What happens if the Internet goes down; do I lose my POS system? No, the Halo Off Line Manager (OLM) keeps store operations running in the event of an Internet outage. The OLM buffers transactions and automatically resynchronizes with the hosted data warehouse once the Internet connection is re-established. In offline mode all POS functions are available as well as POS-based reports that will allow you to perform your end of day and cash

out servers and cashiers." *FAQs*, HALO, https://web.archive.org/web/20061120125911/http://www.myhalo.com/section.asp?catid=274, at 1. Furthermore, providing the ability of the POS terminals to continue functioning—at least temporarily—even if they lose their connection to the server was an established feature of POS systems prior to the patent priority date that was well-known for its ability to reduce the costs of downtime for POS systems, by ensuring that restaurants, retail locations, or other businesses implementing POS systems would be able to rely on their chosen POS system to continue to enter transactions and/or take orders even if a network connection or server was down. Accordingly, a POSITA would have been motivated to provide Halo's POS system with the ability of the POS terminals to continue functioning—at least temporarily—even if they lose their connection to the server, and would have had a reasonable expectation of success in doing so.

To the extent CloudofChange contends that Halo fails to disclose or render obvious this feature, a POSITA would have been motivated to add the ability of the POS terminals to continue functioning—at least temporarily—even if they lose their connection to the server, as disclosed in the Independent Terminals references. *See* Section VI.A.7 [The Claimed Concepts Were Known, Independent Terminals References].

Modifying Halo to enable its POS terminals to continue operating independently would have realized many benefits. For example, adding independent terminals would improve Halo's system by making Halo's system—including Halo's POS terminals—more reliable, increasing revenue (e.g., because of reduced downtime), and making the system work faster (e.g., because menu processing is done locally instead of over a network). *See* Section VI.A.7 [The Claimed Concepts Were Known, Independent Terminals References].

Moreover, the ability to add independent terminals, and the benefits of doing so, would have been predictable to a POSITA because it would have been a combination of prior art elements according to known methods to yield predictable results, because a POSITA would have known of the ability to make POS terminals operate independently from servers (e.g., like all POS terminals pre-dating the Internet), of the general availability of independent terminals, and would have understood that it would have been beneficial and desirable to add independent terminals to a POS system, like the one disclosed in Halo.

Adding independent terminals would have been a simple substitution of one known element (a POS system with POS terminals that do not have the ability to operate independently of a server) for another (a POS system with POS terminals that have the ability to operate independently of a server) to obtain predictable results (the benefits of POS terminals that can continue to operate even if a connection to a server is lost).

Moreover, adding independent terminals would have been obvious to try. A POSITA implementing a POS system would have known that before the advent of computer networking and the Internet, POS terminals were independent terminals (e.g., cash register), and so POS terminals could be operated without a connection to a server. Moreover, a POSITA would have understood that most or all networks sometimes experience connectivity issues, and similarly servers sometimes experience downtime, so it would have been obvious to try implementing a failsafe option to allow POS terminals to operate independently of a server, so that the POS terminals could be used even if the network or server is temporarily inaccessible or otherwise having problems. Implementing independent terminals has known advantages, and a POSITA would have understood the benefits and advantages of independent terminals and been motivated to implement independent terminals to achieve those benefits and advantages.

Furthermore, using independent terminals was a known approach for POS systems, so a POSITA would have been motivated to implement the teachings of independent terminals in the same field (e.g., the teachings of independent terminals in the Independent Terminals References, which are in the same field as Halo) based on design incentives or market forces, including those described in the Independent Terminals References. *See* Section VI.A.7 [The Claimed Concepts Were Known, Independent Terminals References].

Additionally, a POSITA would have had a reasonable expectation of success in adding this feature to Halo. Halo's disclosure of these features is in common with a system that includes independent terminals. In other words, adding the ability for POS terminals to operate offline—e.g., temporarily in case of a network or server outage—would have merely involved the use of a known technique to improve similar systems in the same way.

i. NextPOS alone or in combination with one or more Remote Web Server References and one or more Browser References and one or more Independent Terminals References

NextPOS alone renders obvious each of the features of the '012 patent, as shown in the attached claim chart. *See* Exhibit B-'012-15.

i. Remote Web Server

The '012 patent claims recite a web server that is remote from one or more point of sale terminals. *See* '012 patent, 1[a] ("a web server including POS builder software installed thereon;"), 1[d] ("wherein said POS builder interface is configured to be utilized to access said POS builder software for programmatically creating or modifying said one or more POS terminals in real time over the communications network, wherein said POS builder software is configured to interact with said one or more POS terminals over the communications network in order for the web-based point of sale (POS) builder system to perform functions in accordance with instructions sent from the POS builder interface;"), 1[e] (wherein said POS transactions and

corresponding transaction data from said one or more POS terminals are configured to be transmitted to said web server via the communications network;").

This feature is taught by and/or would have been obvious in view of NextPOS. For example, NextPOS relates generally to a "POS and management software" with features such as table service, hostess & paging, quick service, CRM, inventory management, labor control, payroll preparation, [and] reports." NTPOS, *Looking for a complete restaurant point of sale solution?*, https://web.archive.org/web/20060114023119/http://www.ntpos.com:80/ (Jan. 14, 2006). NextPOS further discloses that "the server computer in our POS network is responsible for holding the database and allowing other systems to connect to it. Along with these tasks, the server can also be used to run programs that complement Aldelo For Restaurants." Aldelo For Restaurants™ User Manual at 20. Remote servers were well known at the time, as was conducting business over the web. Furthermore, NextPOS already employed remote capabilities, such as allowing users "to create, modify and send orders to the kitchen remotely via handheld PDA devices." NTPOS, *Looking for a complete restaurant point of sale solution?*, https://web.archive.org/web/20060114023119/http://www.ntpos.com:80/ (Jan. 14, 2006). Therefore, this feature would have been obvious in view of NextPOS to move a local server to a remote location and shift the system to be web-based.

To the extent CloudofChange contends that NextPOS fails to disclose or render obvious this feature, a POSITA would have been motivated to modify NextPOS's system to make the server remote and web-based, as disclosed in the Remote Web Server References. *See* Section VI.A.4 [The Claimed Concepts Were Known, Remote Web Server / Remote Web Server].

Making NextPOS's server remote and web-based would have realized many benefits. For example, making the server remote would allow a single manager to manage multiple

restaurants, reducing hardware costs and simplifying management of multiple restaurants, ensure uniformity across the multiple restaurants, improve the ability to manage from anywhere, and improve the ability to order from anywhere. *See* Section VI.A.4 [The Claimed Concepts Were Known, Remote Web Server / Remote Web Server]. In addition, because web-servers were well-known long before the Asserted Patents, making the remote server web-based would have allowed NextPOS's system to use standardized hardware and software, or training by using HTML standard protocol, and lower the cost of POS terminals. Making NextPOS's server a remote web server would also allow a single server to manage multiple restaurants, reducing hardware costs and simplifying management of multiple restaurants, ensure uniformity across the multiple restaurants, improve the ability to manage from anywhere, and improve the ability to order from anywhere.

Moreover, the ability to add a remote web server, and the benefits of doing so, would have been predictable to a POSITA because it would have been a combination of prior art elements according to known methods to yield predictable results, because a POSITA would have known of the general availability of remote web servers and would have understood that it would have been beneficial and desirable to add a remote web server to a networked computer system, such as a POS system, like the one disclosed in NextPOS.

Modifying NextPOS to use a remote web server rather than a local web server would have been a simple substitution of one known element (a local server that is maintained by the individual owner) for another (a remote web server) to obtain predictable results (the benefits of using a remote web server).

Moreover, modifying NextPOS to use a remote web server would have been obvious to try. When implementing a server, an administrator only has limited options—making the server

local or remote, making the server web-based or non-web-based. Each solution has its known advantages and disadvantages, and a POSITA would have understood the benefits and advantages of a remote web server and been motivated to implement a remote web server in order to achieve those benefits and advantages.

Furthermore, using a remote web server was a known approach for computing hardware and software in POS systems, and particularly remote server hardware and software, so a POSITA would have been motivated to implement a remote web server, as taught in the Remote Web References in the same field, based on design incentives or market forces, including those described in the Remote Web Server References. *See* Section VI.A.4 [The Claimed Concepts Were Known, Remote Web Server / Remote Web Server].

Additionally, a POSITA would have had a reasonable expectation of success in adding this feature to NextPOS, as NextPOS already discloses a local server and the ability to control management tasks remotely. In other words, modifying NextPOS to use a remote web server would have merely involved the use of a known technique to improve similar systems in the same way.

ii. Browser

The ’012 patent claims recite access to point of sale screens with browser-based web access. *See* claim 8 (“The web-based point of sale (POS) builder system of claim 1, wherein said one or more POS terminals are configured to be accessible by a web browser.”).

This feature is taught by and/or would have been obvious in view of NextPOS. For example, NextPOS relates generally to “POS and management software” with features such as table service, hostess & paging, quick service, CRM, inventory management, labor control, payroll preparation, [and] reports.” NTPOS, *Looking for a complete restaurant point of sale solution?*, https://web.archive.org/web/20060114023119/http://www.ntpos.com:80/ (Jan. 14,

2006). NextPOS further discloses a "server computer" that "is responsible for holding the database and allowing other systems to connect to it" and "run[s] programs." Aldelo For Restaurants™ User Manual at 20. NextPOS also discloses that a "common system requirement is an internet connection on the server computer." *Id.* It would have been obvious to use a browser instead of desktop software, as both were known concepts of how to run a POS system.

To the extent CloudofChange contends that NextPOS fails to disclose or render obvious this feature, a POSITA would have been motivated to modify NextPOS's system to use a browser, as disclosed in the Browser References. *See* Section VI.A.6 [The Claimed Concepts Were Known, Access to Point of Sale Screens / Browser].

Using NextPOS's system with a browser would have realized many benefits. For example, using a browser would allow a business to use standard and commoditized software applications, instead of special software or hardware. In addition, browsers enable the use of "thin clients" that do not need to run applications (other than the browser) locally because the processing is largely done at the remote server. Such thin clients would have reduced requirements for memory and processing power, as compared to clients that need to run the applications locally. Browser access to point-of-sale screens would also allow customers to place orders from their own phones, or remotely. *See* Section VI.A.6 [The Claimed Concepts Were Known, Access to Point of Sale Screens / Browser]. In addition, because browsers were well-known long before the Asserted Patents, using a browser would have allowed NextPOS's system to use standardized hardware and software, or training by using HTML standard protocol, and lower the cost of POS terminals.

Moreover, the ability to use a browser, and the benefits of doing so, would have been predictable to a POSITA because it would have been a combination of prior art elements

according to known methods to yield predictable results, because a POSITA would have known of the general availability of browsers and would have understood that it would have been beneficial and desirable to use browsers to a networked computer system, such as a POS system, like the one disclosed in NextPOS.

Modifying NextPOS to use a browser rather than a non-browser program would have been a simple substitution of one known element (a non-browser program) for another (a browser) to obtain predictable results (the benefits of using a browser).

Moreover, modifying NextPOS to use a browser would have been obvious to try. When implementing a server, an administrator only has limited options—using a non-browser program or a browser. Each solution has its known advantages and disadvantages, and a POSITA would have understood the benefits and advantages of a browser and been motivated to implement a remote web server in order to achieve those benefits and advantages.

Furthermore, using a browser was a known approach for running hardware and software in POS systems, so a POSITA would have been motivated to implement a browser, as taught in the Browser References in the same field, based on design incentives or market forces, including those described in the Browser References. *See* Section VI.A.6 [The Claimed Concepts Were Known, Access to Point of Sale Screens / Browser].

Additionally, a POSITA would have had a reasonable expectation of success in adding this feature to NextPOS, as NextPOS already discloses a program that runs via connection to the internet. In other words, modifying NextPOS to use a browser would have merely involved the use of a known technique to improve similar systems in the same way.

iii. Independent Terminals

Claim 9 of the ’012 patent recites the ability of the POS terminals to continue functioning—at least temporarily—even if they lose their connection to the server. *See* ’012

patent, claim 9 ("The web-based point of sale (POS) builder system of claim 1, wherein said one or more POS terminals, once generated, are configured to operate independently of said web server."). To the extent that CloudofChange contends that NextPOS fails to disclose this feature, a POSITA would have been aware of and would have been motivated to modify the systems and methods of NextPOS to incorporate this feature, or alternatively to combine various teachings from the systems and methods described in one or more of the Independent Terminals references with NextPOS into one system or method.

This feature is taught by and/or would have been obvious in view of NextPOS. For example, NextPOS relates generally to "POS and management software" with features such as table service, hostess & paging, quick service, CRM, inventory management, labor control, payroll preparation, [and] reports." NTPOS, *Looking for a complete restaurant point of sale solution?*, https://web.archive.org/web/20060114023119/http://www.ntpos.com:80/ (Jan. 14, 2006). NextPOS also discloses "one [server computer] that is responsible for maintaining the software database," and that "the server holds the database that all the other terminals connect to when performing transactions." Aldelo, Aldelo for Restaurants User Manual, http://www.cardtransac.com/downloads/POSManual.pdf (1997-2007), at 3. Furthermore, providing the ability of the POS terminals to continue functioning—at least temporarily—even if they lose their connection to the server was an established feature of POS systems prior to the patent priority date that was well-known for its ability to reduce the costs of downtime for POS systems, by ensuring that restaurants, retail locations, or other businesses implementing POS systems would be able to rely on their chosen POS system to continue to enter transactions and/or take orders even if a network connection or server was down. Accordingly, a POSITA would have been motivated to provide NextPOS's POS system with the ability of the POS

terminals to continue functioning—at least temporarily—even if they lose their connection to the server, and would have had a reasonable expectation of success in doing so.

To the extent CloudofChange contends that NextPOS fails to disclose or render obvious this feature, a POSITA would have been motivated to add the ability of the POS terminals to continue functioning—at least temporarily—even if they lose their connection to the server, as disclosed in the Independent Terminals references. *See* Section VI.A.7 [The Claimed Concepts Were Known, Independent Terminals References].

Modifying NextPOS to enable its POS terminals to continue operating independently would have realized many benefits. For example, adding independent terminals would improve NextPOS's system by making NextPOS's system—including NextPOS's POS terminals—more reliable, increasing revenue (e.g., because of reduced downtime), and making the system work faster (e.g., because menu processing is done locally instead of over a network). *See* Section VI.A.7 [The Claimed Concepts Were Known, Independent Terminals References].

Moreover, the ability to add independent terminals, and the benefits of doing so, would have been predictable to a POSITA because it would have been a combination of prior art elements according to known methods to yield predictable results, because a POSITA would have known of the ability to make POS terminals operate independently from servers (e.g., like all POS terminals pre-dating the Internet), of the general availability of independent terminals, and would have understood that it would have been beneficial and desirable to add independent terminals to a POS system, like the one disclosed in NextPOS.

Adding independent terminals would have been a simple substitution of one known element (a POS system with POS terminals that do not have the ability to operate independently of a server) for another (a POS system with POS terminals that have the ability to operate

independently of a server) to obtain predictable results (the benefits of POS terminals that can continue to operate even if a connection to a server is lost).

Moreover, adding independent terminals would have been obvious to try. A POSITA implementing a POS system would have known that before the advent of computer networking and the Internet, POS terminals were independent terminals (e.g., cash register), and so POS terminals could be operated without a connection to a server. Moreover, a POSITA would have understood that most or all networks sometimes experience connectivity issues, and similarly servers sometimes experience downtime, so it would have been obvious to try implementing a failsafe option to allow POS terminals to operate independently of a server, so that the POS terminals could be used even if the network or server is temporarily inaccessible or otherwise having problems. Implementing independent terminals has known advantages, and a POSITA would have understood the benefits and advantages of independent terminals and been motivated to implement independent terminals to achieve those benefits and advantages.

Furthermore, using independent terminals was a known approach for POS systems, so a POSITA would have been motivated to implement the teachings of independent terminals in the same field (e.g., the teachings of independent terminals in the Independent Terminals References, which are in the same field as NextPOS) based on design incentives or market forces, including those described in the Independent Terminals References. *See* Section VI.A.7 [The Claimed Concepts Were Known, Independent Terminals References].

Additionally, a POSITA would have had a reasonable expectation of success in adding this feature to NextPOS. NextPOS's disclosure of these features is in common with a system that includes independent terminals. In other words, adding the ability for POS terminals to

operate offline—e.g., temporarily in case of a network or server outage—would have merely involved the use of a known technique to improve similar systems in the same way.

j. Point of Success alone or in combination with one or more Remote Web Server References and one or more Browser References and one or more Independent Terminals References

Point of Success alone renders obvious each of the features of the ’012 patent, as shown in the attached claim chart. *See* Exhibit B-’012-17.

i. Remote Web Server

The ’012 patent claims recite a web server that is remote from one or more point of sale terminals. *See* ’012 patent, 1[a] (“a web server including POS builder software installed thereon;”), 1[d] (“wherein said POS builder interface is configured to be utilized to access said POS builder software for programmatically creating or modifying said one or more POS terminals in real time over the communications network, wherein said POS builder software is configured to interact with said one or more POS terminals over the communications network in order for the web-based point of sale (POS) builder system to perform functions in accordance with instructions sent from the POS builder interface;”), 1[e] (wherein said POS transactions and corresponding transaction data from said one or more POS terminals are configured to be transmitted to said web server via the communications network;”).

This feature is taught by and/or would have been obvious in view of Point of Success. For example, Point of Success relates generally to POS software that allows managers to “[c]ustomize ordering menus” and “tracks customers, previous order history, and stores a restaurant's entire product list and ordering menu.” Point of Success, *Restaurant Software for POS Systems*, https://web.archive.org/web/20060104044810/http:/www.pointofsuccess.com/restaurant-pos-

software.htm (Jan. 4, 2006). Point of Success further discloses "the Advantage Database Server for managing and protecting data." pointofsuccess, *Restaurant POS Hardware Secrets Revealed*, https://web.archive.org/web/20051222101337/http:/www.pointofsuccess.com/systemrequirements.htm (Dec. 22, 2005). Point of Success further discloses: "The database software requires a computer running Windows 2000 or Windows XP (Windows XP is recommended for easier network setup). The computer running the database software must be on at all times when Point of Success is used. The database server computer can be up day and night or started before using Point of Success. It is best to run the database software on a separate server computer, but the database software can also be loaded on the manager's workstation or other computer on the network." *Id.* Remote servers were well known at the time, as was conducting business over the web. Therefore, this feature would have also been obvious in view of Point of Success to move a local server to a remote location and shift the system to be web-based.

To the extent CloudofChange contends that Point of Success fails to disclose or render obvious this feature, a POSITA would have been motivated to modify Point of Success's system to make the server remote and web-based, as disclosed in the Remote Web Server References. *See* Section VI.A.4 [The Claimed Concepts Were Known, Remote Web Server / Remote Web Server].

Making Point of Success's server remote and web-based would have realized many benefits. For example, making the server remote would allow a single manager to manage multiple restaurants, reducing hardware costs and simplifying management of multiple restaurants, ensure uniformity across the multiple restaurants, improve the ability to manage from anywhere, and improve the ability to order from anywhere. *See* Section VI.A.4 [The Claimed Concepts Were Known, Remote Web Server / Remote Web Server]. In addition,

because web-servers were well-known long before the Asserted Patents, making the remote server web-based would have allowed Point of Success's system to use standardized hardware and software, or training by using HTML standard protocol, and lower the cost of POS terminals. Making Point of Success's server a remote web server would also allow a single server to manage multiple restaurants, reducing hardware costs and simplifying management of multiple restaurants, ensure uniformity across the multiple restaurants, improve the ability to manage from anywhere, and improve the ability to order from anywhere.

Moreover, the ability to add a remote web server, and the benefits of doing so, would have been predictable to a POSITA because it would have been a combination of prior art elements according to known methods to yield predictable results, because a POSITA would have known of the general availability of remote web servers and would have understood that it would have been beneficial and desirable to add a remote web server to a networked computer system, such as a POS system, like the one disclosed in Point of Success.

Modifying Point of Success to use a remote web server rather than a local web server would have been a simple substitution of one known element (a local server that is maintained by the individual owner) for another (a remote web server) to obtain predictable results (the benefits of using a remote web server).

Moreover, modifying Point of Success to use a remote web server would have been obvious to try. When implementing a server, an administrator only has limited options—making the server local or remote, making the server web-based or non-web-based. Each solution has its known advantages and disadvantages, and a POSITA would have understood the benefits and advantages of a remote web server and been motivated to implement a remote web server in order to achieve those benefits and advantages.

Furthermore, using a remote web server was a known approach for computing hardware and software in POS systems, and particularly remote server hardware and software, so a POSITA would have been motivated to implement a remote web server, as taught in the Remote Web References in the same field, based on design incentives or market forces, including those described in the Remote Web Server References. *See* Section VI.A.4 [The Claimed Concepts Were Known, Remote Web Server / Remote Web Server].

Additionally, a POSITA would have had a reasonable expectation of success in adding this feature to Point of Success, as Point of Success already discloses a server that uses internet connection. In other words, modifying Point of Success to use a remote web server would have merely involved the use of a known technique to improve similar systems in the same way.

ii. Browser

The ’012 patent claims recite access to point of sale screens with browser-based web access. *See* claim 8 (“The web-based point of sale (POS) builder system of claim 1, wherein said one or more POS terminals are configured to be accessible by a web browser.”).

This feature is taught by and/or would have been obvious in view of Point of Success. For example, Point of Success relates generally to POS software that Restaurant Software that allows managers to “[c]ustomize ordering menus” and “tracks customers, previous order history, and stores a restaurant's entire product list and ordering menu.” Point of Success, *Restaurant Software for POS Systems,* https://web.archive.org/web/20060104044810/http:/www.pointofsuccess.com/restaurant-pos-software.htm (Jan. 4, 2006). Furthermore, Point of Success discloses that “Microsoft Internet Explorer version 4 or above must be installed” on the manager’s computer. pointofsuccess, *Restaurant POS Hardware Secrets Revealed,* https://web.archive.org/web/20051222101337/http:/www.pointofsuccess.com/systemrequirement

s.htm (Dec. 22, 2005). It thus would have been obvious to use a browser instead of a software program to run the POS system.

To the extent CloudofChange contends that Point of Success fails to disclose or render obvious this feature, a POSITA would have been motivated to modify Point of Success's system to use a browser, as disclosed in the Browser References. *See* Section VI.A.6 [The Claimed Concepts Were Known, Access to Point of Sale Screens / Browser].

Using Point of Success's system with a browser would have realized many benefits. For example, using a browser would allow a business to use standard and commoditized software applications, instead of special software or hardware. In addition, browsers enable the use of "thin clients" that do not need to run applications (other than the browser) locally because the processing is largely done at the remote server. Such thin clients would have reduced requirements for memory and processing power, as compared to clients that need to run the applications locally. Browser access to point-of-sale screens would also allow customers to place orders from their own phones, or remotely. *See* Section VI.A.6 [The Claimed Concepts Were Known, Access to Point of Sale Screens / Browser]. In addition, because browsers were well-known long before the Asserted Patents, using a browser would have allowed Point of Success's system to use standardized hardware and software, or training by using HTML standard protocol, and lower the cost of POS terminals.

Moreover, the ability to use a browser, and the benefits of doing so, would have been predictable to a POSITA because it would have been a combination of prior art elements according to known methods to yield predictable results, because a POSITA would have known of the general availability of browsers and would have understood that it would have been

beneficial and desirable to use browsers to a networked computer system, such as a POS system, like the one disclosed in Point of Success.

Modifying Point of Success to use a browser rather than a non-browser program would have been a simple substitution of one known element (a non-browser program) for another (a browser) to obtain predictable results (the benefits of using a browser).

Moreover, modifying Point of Success to use a browser would have been obvious to try. When implementing a server, an administrator only has limited options—using a non-browser program or a browser. Each solution has its known advantages and disadvantages, and a POSITA would have understood the benefits and advantages of a browser and been motivated to implement a remote web server in order to achieve those benefits and advantages.

Furthermore, using a browser was a known approach for running hardware and software in POS systems, so a POSITA would have been motivated to implement a browser, as taught in the Browser References in the same field, based on design incentives or market forces, including those described in the Browser References. *See* Section VI.A.6 [The Claimed Concepts Were Known, Access to Point of Sale Screens / Browser].

Additionally, a POSITA would have had a reasonable expectation of success in adding this feature to Point of Success, as Point of Success already discloses POS software and having a browser on the computer system. In other words, modifying Point of Success to use a browser would have merely involved the use of a known technique to improve similar systems in the same way.

iii. Independent Terminals

Claim 9 of the ’012 patent recites the ability of the POS terminals to continue functioning—at least temporarily—even if they lose their connection to the server. *See* ’012 patent, claim 9 (“The web-based point of sale (POS) builder system of claim 1, wherein said one

or more POS terminals, once generated, are configured to operate independently of said web server."). To the extent that CloudofChange contends that Point of Success fails to disclose this feature, a POSITA would have been aware of and would have been motivated to modify the systems and methods of Point of Success to incorporate this feature, or alternatively to combine various teachings from the systems and methods described in one or more of the Independent Terminals references with Point of Success into one system or method.

This feature is taught by and/or would have been obvious in view of Point of Success. For example, Point of Success relates generally to POS software that Restaurant Software that allows managers to "[c]ustomize ordering menus" and "tracks customers, previous order history, and stores a restaurant's entire product list and ordering menu." Point of Success, *Restaurant Software for POS Systems*, https://web.archive.org/web/20060104044810/http:/www.pointofsuccess.com/restaurant-pos-software.htm (Jan. 4, 2006). Point of Success also discloses "the Advantage Database Server for managing and protecting data." pointofsuccess, *Restaurant POS Hardware Secrets Revealed*, https://web.archive.org/web/20051222101337/http:/www.pointofsuccess.com/systemrequirements.htm (Dec. 22, 2005). Point of Success further discloses: "The database software requires a computer running Windows 2000 or Windows XP (Windows XP is recommended for easier network setup). The computer running the database software must be on at all times when Point of Success is used. The database server computer can be up day and night or started before using Point of Success. It is best to run the database software on a separate server computer, but the database software can also be loaded on the manager's workstation or other computer on the network." *Id.* Furthermore, providing the ability of the POS terminals to continue functioning—at least temporarily—even if they lose their connection to the server was an established feature

of POS systems prior to the patent priority date that was well-known for its ability to reduce the costs of downtime for POS systems, by ensuring that restaurants, retail locations, or other businesses implementing POS systems would be able to rely on their chosen POS system to continue to enter transactions and/or take orders even if a network connection or server was down. Accordingly, a POSITA would have been motivated to provide Point of Success's POS system with the ability of the POS terminals to continue functioning—at least temporarily—even if they lose their connection to the server, and would have had a reasonable expectation of success in doing so.

To the extent CloudofChange contends that Point of Success fails to disclose or render obvious this feature, a POSITA would have been motivated to add the ability of the POS terminals to continue functioning—at least temporarily—even if they lose their connection to the server, as disclosed in the Independent Terminals references. *See* Section VI.A.7 [The Claimed Concepts Were Known, Independent Terminals References].

Modifying Point of Success to enable its POS terminals to continue operating independently would have realized many benefits. For example, adding independent terminals would improve Point of Success's system by making Point of Success's system—including Point of Success's POS terminals—more reliable, increasing revenue (e.g., because of reduced downtime), and making the system work faster (e.g., because menu processing is done locally instead of over a network). *See* Section VI.A.7 [The Claimed Concepts Were Known, Independent Terminals References].

Moreover, the ability to add independent terminals, and the benefits of doing so, would have been predictable to a POSITA because it would have been a combination of prior art elements according to known methods to yield predictable results, because a POSITA would

have known of the ability to make POS terminals operate independently from servers (e.g., like all POS terminals pre-dating the Internet), of the general availability of independent terminals, and would have understood that it would have been beneficial and desirable to add independent terminals to a POS system, like the one disclosed in Point of Success.

Adding independent terminals would have been a simple substitution of one known element (a POS system with POS terminals that do not have the ability to operate independently of a server) for another (a POS system with POS terminals that have the ability to operate independently of a server) to obtain predictable results (the benefits of POS terminals that can continue to operate even if a connection to a server is lost).

Moreover, adding independent terminals would have been obvious to try. A POSITA implementing a POS system would have known that before the advent of computer networking and the Internet, POS terminals were independent terminals (e.g., cash register), and so POS terminals could be operated without a connection to a server. Moreover, a POSITA would have understood that most or all networks sometimes experience connectivity issues, and similarly servers sometimes experience downtime, so it would have been obvious to try implementing a failsafe option to allow POS terminals to operate independently of a server, so that the POS terminals could be used even if the network or server is temporarily inaccessible or otherwise having problems. Implementing independent terminals has known advantages, and a POSITA would have understood the benefits and advantages of independent terminals and been motivated to implement independent terminals to achieve those benefits and advantages.

Furthermore, using independent terminals was a known approach for POS systems, so a POSITA would have been motivated to implement the teachings of independent terminals in the same field (e.g., the teachings of independent terminals in the Independent Terminals

References, which are in the same field as Point of Success) based on design incentives or market forces, including those described in the Independent Terminals References. *See* Section VI.A.7 [The Claimed Concepts Were Known, Independent Terminals References].

Additionally, a POSITA would have had a reasonable expectation of success in adding this feature to Point of Success. Point of Success's disclosure of these features is in common with a system that includes independent terminals. In other words, adding the ability for POS terminals to operate offline—e.g., temporarily in case of a network or server outage—would have merely involved the use of a known technique to improve similar systems in the same way.

k. InfoGenesis alone or in combination with one or more POS GUI References

InfoGenesis alone renders obvious each of the features of the '012 patent, as shown in the attached claim chart. *See* Exhibit B-'012-19.

i. POS GUI

The '012 patent claims recite a POS building tool that allows creating the graphical user interface of a menu with no special training. *See* '012 patent, claim 6 ("The web-based point of sale (POS) builder system of claim 1, wherein said POS builder software is configured to provide instructions to add new POS terminals"), claim 7 ("The web-based point of sale (POS) builder system of claim 1, wherein said POS builder software is configured to provide instructions to modify existing POS terminals"), claim 10 ("The web-based point of sale (POS) builder system of claim 1, wherein a number, shape and arrangement of selection keys or buttons for said one or more POS terminals are specified from the POS builder interface"), claim 11 ("The web-based point of sale (POS) builder system of claim 10, wherein items are associated with said selection keys or buttons from the POS builder interface"). To the extent that CloudofChange contends that InfoGenesis fails to disclose this feature, a POSITA would have been aware of and would

have been motivated to modify the systems and methods of InfoGenesis to incorporate this feature, or alternatively to combine various teachings from the systems and methods described in one or more of the POS GUI references with InfoGenesis into one system or method.

This feature is taught by and/or would have been obvious in view of InfoGenesis. For example, InfoGenesis relates generally to "point-of-sale software that combines powerful reporting and configuration capabilities in the back office with an easy-to-use, touch screen Windows Terminal Application (WTA) that makes better service for frontline staff second nature." InfoGenesis, Point-of-Sale, https://web.archive.org/web/20070416203641/http://www.infogenesis.com/restaurants/pointofsalepos.aspx (Mar. 11, 2006), at 1. InfoGenesis also discloses "If we need to add an item or change a price, we can do it on the fly . . . When the company introduces database changes, such as new menu items, Infogenesis automatically builds the updates into the system." Hospitality Technology, Point of Service: Now and Later of Restaurant POS, https://hospitalitytech.com/point-service-now-and-later-restaurant-pos (April 9, 2007), at 2. "InfoGenesis POS allows non-technical staff to tailor system features to expanding business needs without costly, time-consuming custom programming. All POS management is done in the familiar Windows environment. . . . The 'point and click' interface, drop-down lists, and drill down icons make learning InfoGenesis POS a breeze." InfoGenesis, Point-of-Sale, https://web.archive.org/web/20051210182507/http://www.infogenesis.com:80/products/revelation.aspx (Dec. 10, 2005), at 1-2. Furthermore, providing a POS building tool that allows creating the graphical user interface of a menu with no special training was a common feature in POS systems prior to the patent priority date and was well-known for its ability to reduce implementation and maintenance costs of POS systems, by avoiding requirements for specialized

IT support (e.g., programmers) to set up or maintain the graphical user interface of a POS system menu. Accordingly, a POSITA would have been motivated to provide InfoGenesis's POS system with a POS building tool that allows creating the graphical user interface of a menu with no special training, and would have had a reasonable expectation of success in doing so.

To the extent CloudofChange contends that InfoGenesis fails to disclose or render obvious this feature, a POSITA would have been motivated to add a POS building tool that allows creating the graphical user interface of a menu with no special training to InfoGenesis's system, as disclosed in the POS GUI references. *See* Section VI.A.3 [The Claimed Concepts Were Known, POS GUI References].

Adding a POS building tool that allows creating the graphical user interface of a menu with no special training would have realized the many benefits of POS GUIs. For example, adding a POS builder tool that allows creating the graphical user interface of a menu with no special training would improve InfoGenesis's system by making InfoGenesis's system—including InfoGenesis's POS terminals—easier to install and manage, lowering the cost of installation and maintenance, reducing or eliminating special training to manage (e.g., allowing any employee—rather than just IT—to add or update graphical user interfaces of menus or of other aspects of the POS terminals), and lowering IT costs. *See* Section VI.A.3 [The Claimed Concepts Were Known, POS GUI References].

Moreover, the ability to add a POS building tool that allows creating the graphical user interface of a menu with no special training, and the benefits of doing so, would have been predictable to a POSITA because it would have been a combination of prior art elements according to known methods to yield predictable results, because a POSITA would have known of the general availability of POS builder tools, the general availability of graphical user

interfaces, and the general availability of easy-to-use tools for building graphical user interfaces, and would have understood that it would have been beneficial and desirable to add a POS building tool that allows creating the graphical user interface of a menu with no special training to a POS system, like the one disclosed in InfoGenesis.

Adding a POS building tool that allows creating the graphical user interface of a menu with no special training would have been a simple substitution of one known element (a POS system that requires programmers to manually build a user interface of a menu) for another (a POS system with a POS building tool that allows creating the graphical user interface of a menu with no special training) to obtain predictable results (the benefits of a POS building tool that allows creating the graphical user interface of a menu with no special training).

Moreover, adding a POS building tool that allows creating the graphical user interface of a menu with no special training would have been obvious to try. When implementing a POS system, an administrator only has limited options—configuring menu interfaces by hiring a programmer or other IT professional to manually program them, or configuring graphical user interfaces of menus with a POS building tool that allows creating the graphical user interface of a menu with no special training, and thus is usable by any employee, including said manager. Each solution has its known advantages, and a POSITA would have understood the benefits and advantages of a POS building tool that allows creating the graphical user interface of a menu with no special training and been motivated to implement a POS building tool that allows creating the graphical user interface of a menu with no special training in order to achieve those benefits and advantages.

Furthermore, using a POS building tool that allows creating the graphical user interface of a menu with no special training was a known approach for POS systems, so a POSITA would

have been motivated to implement the teachings of a POS building tool that allows creating the graphical user interface of a menu with no special training as described in the POS GUI References, which are in the same field as InfoGenesis, based on design incentives or market forces, including those described in the POS GUI References. *See* Section VI.A.3 [The Claimed Concepts Were Known, POS GUI References].

Additionally, a POSITA would have had a reasonable expectation of success in adding this feature to InfoGenesis. InfoGenesis's disclosure of these features is in common with a system that includes a POS building tool that allows creating the graphical user interface of a menu with no special training. In other words, adding a POS building tool that allows creating the graphical user interface of a menu with no special training would have merely involved the use of a known technique to improve similar systems in the same way.

l. Squirrel alone or in combination with one or more POS GUI References and one or more Remote Web Server References and one or more Browser References and one or more Independent Terminals References

Squirrel alone renders obvious each of the features of the '012 patent, as shown in the attached claim chart. *See* Exhibit B-'012-21.

i. POS GUI

The '012 patent claims recite a POS building tool that allows creating the graphical user interface of a menu with no special training. *See* '012 patent, claim 6 ("The web-based point of sale (POS) builder system of claim 1, wherein said POS builder software is configured to provide instructions to add new POS terminals"), claim 7 ("The web-based point of sale (POS) builder system of claim 1, wherein said POS builder software is configured to provide instructions to modify existing POS terminals"), claim 10 ("The web-based point of sale (POS) builder system of claim 1, wherein a number, shape and arrangement of selection keys or buttons for said one or

more POS terminals are specified from the POS builder interface"), claim 11 ("The web-based point of sale (POS) builder system of claim 10, wherein items are associated with said selection keys or buttons from the POS builder interface"). To the extent that CloudofChange contends that Squirrel fails to disclose this feature, a POSITA would have been aware of and would have been motivated to modify the systems and methods of Squirrel to incorporate this feature, or alternatively to combine various teachings from the systems and methods described in one or more of the POS GUI references with Squirrel into one system or method.

This feature is taught by and/or would have been obvious in view of Squirrel. For example, Squirrel relates generally to a "Restaurant Point of Sale and Management system." Squirrelsystems, *SquirrelOne Solutions*, https://web.archive.org/web/20051201101248/http://www.squirrelsystems.com:80/solutions/index.html (Dec. 1, 2005). Squirrel also discloses "SquirrelOne's open architecture, built on a Microsoft SQL relational database, provides on-demand access to real time information. Make menu and price changes instantly, at your convenience. Review and analyze live data from any location. Know what's happening in your operations now, not tomorrow." *Id.* Squirrel additionally discloses, "Corporate POS Manager facilitates the centralization of menu management, freeing restaurant managers from the burden of making menu changes. New information can be sent to locations at any time and can be automatically incorporated into the local database, whenever it's convenient. . . . You can easily make changes to the menu database at the corporate center, and download them to the restaurants which can in turn perform the up-dates at any time. Or, changes can be pre-scheduled to update automatically (unattended if you prefer)." Squirrelsystems: Corporate POS, https://web.archive.org/web/20060327174026/http://www.squirrelsystems.com/solutions/corppo

s.html (May 27, 2006). Furthermore, providing a POS building tool that allows creating the graphical user interface of a menu with no special training was a common feature in POS systems prior to the patent priority date and was well-known for its ability to reduce implementation and maintenance costs of POS systems, by avoiding requirements for specialized IT support (e.g., programmers) to set up or maintain the graphical user interface of a POS system menu. Accordingly, a POSITA would have been motivated to provide Squirrel's POS system with a POS building tool that allows creating the graphical user interface of a menu with no special training, and would have had a reasonable expectation of success in doing so.

To the extent CloudofChange contends that Squirrel fails to disclose or render obvious this feature, a POSITA would have been motivated to add a POS building tool that allows creating the graphical user interface of a menu with no special training to Squirrel's system, as disclosed in the POS GUI references. *See* Section VI.A.3 [The Claimed Concepts Were Known, POS GUI References].

Adding a POS building tool that allows creating the graphical user interface of a menu with no special training would have realized the many benefits of POS GUIs. For example, adding a POS builder tool that allows creating the graphical user interface of a menu with no special training would improve Squirrel's system by making Squirrel's system—including Squirrel's POS terminals—easier to install and manage, lowering the cost of installation and maintenance, reducing or eliminating special training to manage (e.g., allowing any employee—rather than just IT—to add or update graphical user interfaces of menus or of other aspects of the POS terminals), and lowering IT costs. *See* Section VI.A.3 [The Claimed Concepts Were Known, POS GUI References].

Moreover, the ability to add a POS building tool that allows creating the graphical user interface of a menu with no special training, and the benefits of doing so, would have been predictable to a POSITA because it would have been a combination of prior art elements according to known methods to yield predictable results, because a POSITA would have known of the general availability of POS builder tools, the general availability of graphical user interfaces, and the general availability of easy-to-use tools for building graphical user interfaces, and would have understood that it would have been beneficial and desirable to add a POS building tool that allows creating the graphical user interface of a menu with no special training to a POS system, like the one disclosed in Squirrel.

Adding a POS building tool that allows creating the graphical user interface of a menu with no special training would have been a simple substitution of one known element (a POS system that requires programmers to manually build a user interface of a menu) for another (a POS system with a POS building tool that allows creating the graphical user interface of a menu with no special training) to obtain predictable results (the benefits of a POS building tool that allows creating the graphical user interface of a menu with no special training).

Moreover, adding a POS building tool that allows creating the graphical user interface of a menu with no special training would have been obvious to try. When implementing a POS system, an administrator only has limited options—configuring menu interfaces by hiring a programmer or other IT professional to manually program them, or configuring graphical user interfaces of menus with a POS building tool that allows creating the graphical user interface of a menu with no special training, and thus is usable by any employee, including said manager. Each solution has its known advantages, and a POSITA would have understood the benefits and advantages of a POS building tool that allows creating the graphical user interface of a menu

with no special training and been motivated to implement a POS building tool that allows creating the graphical user interface of a menu with no special training in order to achieve those benefits and advantages.

Furthermore, using a POS building tool that allows creating the graphical user interface of a menu with no special training was a known approach for POS systems, so a POSITA would have been motivated to implement the teachings of a POS building tool that allows creating the graphical user interface of a menu with no special training as described in the POS GUI References, which are in the same field as Squirrel, based on design incentives or market forces, including those described in the POS GUI References. *See* Section VI.A.3 [The Claimed Concepts Were Known, POS GUI References].

Additionally, a POSITA would have had a reasonable expectation of success in adding this feature to Squirrel. Squirrel's disclosure of these features is in common with a system that includes a POS building tool that allows creating the graphical user interface of a menu with no special training. In other words, adding a POS building tool that allows creating the graphical user interface of a menu with no special training would have merely involved the use of a known technique to improve similar systems in the same way.

ii. Remote Web Server

The '012 patent claims recite a web server that is remote from one or more point of sale terminals. *See* '012 patent, 1[a] ("a web server including POS builder software installed thereon;"), 1[d] ("wherein said POS builder interface is configured to be utilized to access said POS builder software for programmatically creating or modifying said one or more POS terminals in real time over the communications network, wherein said POS builder software is configured to interact with said one or more POS terminals over the communications network in order for the web-based point of sale (POS) builder system to perform functions in accordance

with instructions sent from the POS builder interface;"), 1[e] (wherein said POS transactions and corresponding transaction data from said one or more POS terminals are configured to be transmitted to said web server via the communications network;").

This feature is taught by and/or would have been obvious in view of Squirrel. For example, Squirrel relates generally to a "Restaurant Point of Sale and Management system." Squirrelsystems, *SquirrelOne Solutions*, https://web.archive.org/web/20051201101248/http://www.squirrelsystems.com:80/solutions/index.html (Dec. 1, 2005). Squirrel further discloses a "32-bit client/server application" that "is fully integrated with the latest Microsoft environment." *Id.* Squirrel also discloses that its POS Manager was "[b]uilt on the latest Microsoft SQL Server relational database" "with a familiar browser style desktop [that] is able to be accessed directly from the central PC or remotely via the internet" Squirrelsystems, *Corporate POS*, https://web.archive.org/web/20060327174026/http://www.squirrelsystems.com/solutions/corppos.html (Mar. 27, 2006). Conducting business over the web was well known at the time. Therefore, this feature would have also been obvious in view of Squirrel to shift the system to be web-based.

To the extent CloudofChange contends that Squirrel fails to disclose or render obvious this feature, a POSITA would have been motivated to modify Squirrel's system to make the server remote and web-based, as disclosed in the Remote Web Server References. *See* Section VI.A.4 [The Claimed Concepts Were Known, Remote Web Server / Remote Web Server].

Making Squirrel's server remote and web-based would have realized many benefits. For example, making the server remote would allow a single manager to manage multiple restaurants, reducing hardware costs and simplifying management of multiple restaurants, ensure

uniformity across the multiple restaurants, improve the ability to manage from anywhere, and improve the ability to order from anywhere. *See* Section VI.A.4 [The Claimed Concepts Were Known, Remote Web Server / Remote Web Server]. In addition, because web-servers were well-known long before the Asserted Patents, making the remote server web-based would have allowed Squirrel's system to use standardized hardware and software, or training by using HTML standard protocol, and lower the cost of POS terminals. Making Squirrel's server a remote web server would also allow a single server to manage multiple restaurants, reducing hardware costs and simplifying management of multiple restaurants, ensure uniformity across the multiple restaurants, improve the ability to manage from anywhere, and improve the ability to order from anywhere.

Moreover, the ability to add a remote web server, and the benefits of doing so, would have been predictable to a POSITA because it would have been a combination of prior art elements according to known methods to yield predictable results, because a POSITA would have known of the general availability of remote web servers and would have understood that it would have been beneficial and desirable to add a remote web server to a networked computer system, such as a POS system, like the one disclosed in Squirrel.

Modifying Squirrel to use a remote web server rather than a local web server would have been a simple substitution of one known element (a local server that is maintained by the individual owner) for another (a remote web server) to obtain predictable results (the benefits of using a remote web server).

Moreover, modifying Squirrel to use a remote web server would have been obvious to try. When implementing a server, an administrator only has limited options—making the server local or remote, making the server web-based or non-web-based. Each solution has its known

advantages and disadvantages, and a POSITA would have understood the benefits and advantages of a remote web server and been motivated to implement a remote web server in order to achieve those benefits and advantages.

Furthermore, using a remote web server was a known approach for computing hardware and software in POS systems, and particularly remote server hardware and software, so a POSITA would have been motivated to implement a remote web server, as taught in the Remote Web References in the same field, based on design incentives or market forces, including those described in the Remote Web Server References. *See* Section VI.A.4 [The Claimed Concepts Were Known, Remote Web Server / Remote Web Server].

Additionally, a POSITA would have had a reasonable expectation of success in adding this feature to Squirrel, as Squirrel already discloses a browser like application that could be accessed remotely. In other words, modifying Squirrel to use a remote web server would have merely involved the use of a known technique to improve similar systems in the same way.

iii. Browser

The ’012 patent claims recite access to point of sale screens with browser-based web access. *See* claim 8 (“The web-based point of sale (POS) builder system of claim 1, wherein said one or more POS terminals are configured to be accessible by a web browser.”).

This feature is taught by and/or would have been obvious in view of Squirrel. For example, Squirrel relates generally to a “Restaurant Point of Sale and Management system.” Squirrelsystems, *SquirrelOne Solutions*, https://web.archive.org/web/20051201101248/http://www.squirrelsystems.com:80/solutions/index.html (Dec. 1, 2005). Squirrel further discloses that its POS Manager was “[b]uilt on the latest Microsoft SQL Server relational database” “with a familiar browser style desktop [that] is able to be accessed directly from the central PC or remotely via the internet.” Squirrelsystems,

Corporate POS,

https://web.archive.org/web/20060327174026/http://www.squirrelsystems.com/solutions/corppos.html (Mar. 27, 2006). It thus would have been obvious to use a browser to run the POS Manager. It would have been obvious to use a browser when ordering on the web.

To the extent CloudofChange contends that Squirrel fails to disclose or render obvious this feature, a POSITA would have been motivated to modify Squirrel's system to use a browser, as disclosed in the Browser References. *See* Section VI.A.6 [The Claimed Concepts Were Known, Access to Point of Sale Screens / Browser].

Using Squirrel's system with a browser would have realized many benefits. For example, using a browser would allow a business to use standard and commoditized software applications, instead of special software or hardware. In addition, browsers enable the use of "thin clients" that do not need to run applications (other than the browser) locally because the processing is largely done at the remote server. Such thin clients would have reduced requirements for memory and processing power, as compared to clients that need to run the applications locally. Browser access to point-of-sale screens would also allow customers to place orders from their own phones, or remotely. *See* Section VI.A.6 [The Claimed Concepts Were Known, Access to Point of Sale Screens / Browser]. In addition, because browsers were well-known long before the Asserted Patents, using a browser would have allowed Squirrel's system to use standardized hardware and software, or training by using HTML standard protocol, and lower the cost of POS terminals.

Moreover, the ability to use a browser, and the benefits of doing so, would have been predictable to a POSITA because it would have been a combination of prior art elements according to known methods to yield predictable results, because a POSITA would have known

of the general availability of browsers and would have understood that it would have been beneficial and desirable to use browsers to a networked computer system, such as a POS system, like the one disclosed in Squirrel.

Modifying Squirrel to use a browser rather than a non-browser program would have been a simple substitution of one known element (a non-browser program) for another (a browser) to obtain predictable results (the benefits of using a browser).

Moreover, modifying Squirrel to use a browser would have been obvious to try. When implementing a server, an administrator only has limited options—using a non-browser program or a browser. Each solution has its known advantages and disadvantages, and a POSITA would have understood the benefits and advantages of a browser and been motivated to implement a remote web server in order to achieve those benefits and advantages.

Furthermore, using a browser was a known approach for running hardware and software in POS systems, so a POSITA would have been motivated to implement a browser, as taught in the Browser References in the same field, based on design incentives or market forces, including those described in the Browser References. *See* Section VI.A.6 [The Claimed Concepts Were Known, Access to Point of Sale Screens / Browser].

Additionally, a POSITA would have had a reasonable expectation of success in adding this feature to Squirrel, as Squirrel already discloses a browser style desktop that is able to be accessed remotely via the Internet. In other words, modifying Squirrel to use a browser would have merely involved the use of a known technique to improve similar systems in the same way.

iv. Independent Terminals

Claim 9 of the ’012 patent recites the ability of the POS terminals to continue functioning—at least temporarily—even if they lose their connection to the server. *See* ’012 patent, claim 9 (“The web-based point of sale (POS) builder system of claim 1, wherein said one

or more POS terminals, once generated, are configured to operate independently of said web server."). To the extent that CloudofChange contends that Squirrel fails to disclose this feature, a POSITA would have been aware of and would have been motivated to modify the systems and methods of Squirrel to incorporate this feature, or alternatively to combine various teachings from the systems and methods described in one or more of the Independent Terminals references with Squirrel into one system or method.

This feature is taught by and/or would have been obvious in view of Squirrel. For example, Squirrel relates generally to a "Restaurant Point of Sale and Management system." Squirrelsystems, *SquirrelOne Solutions*, https://web.archive.org/web/20051201101248/http://www.squirrelsystems.com:80/solutions/index.html (Dec. 1, 2005). Squirrel also discloses that Squirrel's system is "[b]uilt on the latest Microsoft SQL Server relational database . . . . Squirrel's POS Manager, with a familiar browser style desktop is able to be accessed directly from the central PC or remotely via the internet. It has an open architecture Ethernet network communication which is a worldwide standard." Squirrelsystems: Corporate POS, https://web.archive.org/web/20060327174026/http://www.squirrelsystems.com/solutions/corppos.html (May 27, 2006). Furthermore, providing the ability of the POS terminals to continue functioning—at least temporarily—even if they lose their connection to the server was an established feature of POS systems prior to the patent priority date that was well-known for its ability to reduce the costs of downtime for POS systems, by ensuring that restaurants, retail locations, or other businesses implementing POS systems would be able to rely on their chosen POS system to continue to enter transactions and/or take orders even if a network connection or server was down. Accordingly, a POSITA would have been motivated to provide Squirrel's

POS system with the ability of the POS terminals to continue functioning—at least temporarily—even if they lose their connection to the server, and would have had a reasonable expectation of success in doing so.

To the extent CloudofChange contends that Squirrel fails to disclose or render obvious this feature, a POSITA would have been motivated to add the ability of the POS terminals to continue functioning—at least temporarily—even if they lose their connection to the server, as disclosed in the Independent Terminals references. *See* Section VI.A.7 [The Claimed Concepts Were Known, Independent Terminals References].

Modifying Squirrel to enable its POS terminals to continue operating independently would have realized many benefits. For example, adding independent terminals would improve Squirrel's system by making Squirrel's system—including Squirrel's POS terminals—more reliable, increasing revenue (e.g., because of reduced downtime), and making the system work faster (e.g., because menu processing is done locally instead of over a network). *See* Section VI.A.7 [The Claimed Concepts Were Known, Independent Terminals References].

Moreover, the ability to add independent terminals, and the benefits of doing so, would have been predictable to a POSITA because it would have been a combination of prior art elements according to known methods to yield predictable results, because a POSITA would have known of the ability to make POS terminals operate independently from servers (e.g., like all POS terminals pre-dating the Internet), of the general availability of independent terminals, and would have understood that it would have been beneficial and desirable to add independent terminals to a POS system, like the one disclosed in Squirrel.

Adding independent terminals would have been a simple substitution of one known element (a POS system with POS terminals that do not have the ability to operate independently

of a server) for another (a POS system with POS terminals that have the ability to operate independently of a server) to obtain predictable results (the benefits of POS terminals that can continue to operate even if a connection to a server is lost).

Moreover, adding independent terminals would have been obvious to try. A POSITA implementing a POS system would have known that before the advent of computer networking and the Internet, POS terminals were independent terminals (e.g., cash register), and so POS terminals could be operated without a connection to a server. Moreover, a POSITA would have understood that most or all networks sometimes experience connectivity issues, and similarly servers sometimes experience downtime, so it would have been obvious to try implementing a failsafe option to allow POS terminals to operate independently of a server, so that the POS terminals could be used even if the network or server is temporarily inaccessible or otherwise having problems. Implementing independent terminals has known advantages, and a POSITA would have understood the benefits and advantages of independent terminals and been motivated to implement independent terminals to achieve those benefits and advantages.

Furthermore, using independent terminals was a known approach for POS systems, so a POSITA would have been motivated to implement the teachings of independent terminals in the same field (e.g., the teachings of independent terminals in the Independent Terminals References, which are in the same field as Squirrel) based on design incentives or market forces, including those described in the Independent Terminals References. *See* Section VI.A.7 [The Claimed Concepts Were Known, Independent Terminals References].

Additionally, a POSITA would have had a reasonable expectation of success in adding this feature to Squirrel. Squirrel's disclosure of these features is in common with a system that includes independent terminals. In other words, adding the ability for POS terminals to operate

offline—e.g., temporarily in case of a network or server outage—would have merely involved the use of a known technique to improve similar systems in the same way.

3. ’793 Patent

Prior art references rendering the asserted claims of the ’793 patent obvious, alone or in combination with other references, are discussed below and included in Exhibits C-’793-1 to C-’793-21. Further reasons to combine the references identified in Exhibits C-’793-1 to C-’793-21 include the nature of the problem being solved, the express, implied and inherent teachings of the prior art, the knowledge of persons of ordinary skill in the art, that such combinations would have yielded predictable results, and that such combinations would have represented known alternatives to a POSITA.

Illustrative obviousness combinations are listed below (it should be understood that each listed combination can also include the knowledge of a POSITA). The Asserted Claims of the ’793 patent are obvious over:

- Tengler alone or in combination with one or more Browser References
- Olson alone or in combination with one or more Browser References
- Flashpoint alone or in combination with one or more Remote Web Server References and one or more Browser References
- Halo alone or in combination with one or more Remote Web Server References
- Phoenix alone or in combination with one or more POS Builder References and one or more Remote Web Server References
- InfoGenesis alone or in combination with one or more Remote Web Server References and one or more Browser References

- Simphony alone or in combination with one or more POS Builder References and one or more Remote Web Server References and one or more Browser References
- Squirrel alone or in combination with one or more Remote Web Server References and one or more Browser References

a. Tengler alone or in combination with one or more Browser References

Tengler alone renders obvious each of the features of the ’793 patent, as shown in the attached claim chart. *See* Exhibit C-’793-4.

i. Browser

The ’793 patent claims recite access to point of sale screens with browser-based web access. *See* ’793 patent, claim 11 (“The web-based point of sale (POS) builder system of claim 1, wherein the POS builder interface is accessible via a web browser.”).

This feature is taught by and/or would have been obvious in view of Tengler. For example, Tengler relates generally to “an apparatus that includes stations in a quick-serve commercial establishment, the stations enabling users to enter orders, check the status of orders, and assemble orders for delivery, a network interconnecting the stations, each of the stations comprising storage for data about the orders, and software to cause copies of the data to be cached at multiple ones of the stations.” Tengler, [0011]. Tengler further discloses a “manager station” that can use a “broadband data line” to “link the manager to the Internet.” Tengler, [0073]. Tengler discloses that “managers can view information about the multiple restaurants” using interfaces “over the Internet.” *Id.* Tengler also discloses that, while “a main server computer” “runs the system server software,” “the database,” “the store-resident web server software,” “the production display software,” and “the drive-through display service,” there is a “computer” that “runs a web browser.” Tengler, [0016]. Therefore, it would have been obvious

to use a browser for the POS system. It would have been obvious to use a browser when ordering on the web.

To the extent CloudofChange contends that Tengler fails to disclose or render obvious this feature, a POSITA would have been motivated to modify Tengler's system to use a browser, as disclosed in the Browser References. *See* Section VI.A.6 [The Claimed Concepts Were Known, Access to Point of Sale Screens / Browser].

Using Tengler's system with a browser would have realized many benefits. For example, using a browser would allow a business to use standard and commoditized software applications, instead of special software or hardware. In addition, browsers enable the use of "thin clients" that do not need to run applications (other than the browser) locally because the processing is largely done at the remote server. Such thin clients would have reduced requirements for memory and processing power, as compared to clients that need to run the applications locally. Browser access to point-of-sale screens would also allow customers to place orders from their own phones, or remotely. *See* Section VI.A.6 [The Claimed Concepts Were Known, Access to Point of Sale Screens / Browser]. In addition, because browsers were well-known long before the Asserted Patents, using a browser would have allowed Tengler's system to use standardized hardware and software, or training by using HTML standard protocol, and lower the cost of POS terminals.

Moreover, the ability to use a browser, and the benefits of doing so, would have been predictable to a POSITA because it would have been a combination of prior art elements according to known methods to yield predictable results, because a POSITA would have known of the general availability of browsers and would have understood that it would have been

beneficial and desirable to use browsers to a networked computer system, such as a POS system, like the one disclosed in Tengler.

Modifying Tengler to use a browser rather than a non-browser program would have been a simple substitution of one known element (a non-browser program) for another (a browser) to obtain predictable results (the benefits of using a browser).

Moreover, modifying Tengler to use a browser would have been obvious to try. When implementing a server, an administrator only has limited options—using a non-browser program or a browser. Each solution has its known advantages and disadvantages, and a POSITA would have understood the benefits and advantages of a browser and been motivated to implement a remote web server in order to achieve those benefits and advantages.

Furthermore, using a browser was a known approach for running hardware and software in POS systems, so a POSITA would have been motivated to implement a browser, as taught in the Browser References in the same field, based on design incentives or market forces, including those described in the Browser References. *See* Section VI.A.6 [The Claimed Concepts Were Known, Access to Point of Sale Screens / Browser].

Additionally, a POSITA would have had a reasonable expectation of success in adding this feature to Tengler, as Tengler already discloses running a POS system over the internet, and computers using web browsers. In other words, modifying Tengler to use a browser would have merely involved the use of a known technique to improve similar systems in the same way.

b. Olson alone or in combination with one or more Browser References

Olson alone renders obvious each of the features of the ’793 patent, as shown in the attached claim chart. *See* Exhibit C-’793-8.

i. Browser

The ’793 patent claims recite access to point of sale screens with browser-based web access. *See* ’793 patent, claim 11 (“The web-based point of sale (POS) builder system of claim 1, wherein the POS builder interface is accessible via a web browser.”).

This feature is taught by and/or would have been obvious in view of Olson. For example, Olson relates generally to “the managing of the points of sale (POS) with Web-based back-office systems. POS transaction engine terminals are implemented on personal computer, PC hardware using PC software with a relational database. The transaction engine architecture provides cost-effective application customization. Web-based back-office software is implemented on servers. POS and BO systems are monitored and supported via the web using protocol and open software components.” Olson, Abstract. Olson further discloses a “point of sale system with a web-based back office system” with real-time status of POS terminals and back-office web servers “obtained at remote locations via personal computers with browser based web access.” Olson, claim 5. Olson further discloses “Remote and local web browsers” as “a method used by store managers, corporate administrators, or system administers to access and view a company & its store's instantaneous performance from minute to minute.” Olson, [0026]. Olson discloses that “[t]he POS, BO, and Web Browsers all communicate via the Web.” *Id.* Olson further discloses that “[t]he protocols between the POS and Web browsers and BO servers is HTTPs, XMURPC. 35. XML-RPC is a remote procedure call protocol which uses XML to encode its calls and HTTP as a transport mechanism.” *Id.* It therefore would have been obvious to use browser-based web for access to POS screens.

To the extent CloudofChange contends that Olson fails to disclose or render obvious this feature, a POSITA would have been motivated to modify Olson’s system to use a browser, as

disclosed in the Browser References. *See* Section VI.A.6 [The Claimed Concepts Were Known, Access to Point of Sale Screens / Browser].

Using Olson's system with a browser would have realized many benefits. For example, using a browser would allow a business to use standard and commoditized software applications, instead of special software or hardware. In addition, browsers enable the use of "thin clients" that do not need to run applications (other than the browser) locally because the processing is largely done at the remote server. Such thin clients would have reduced requirements for memory and processing power, as compared to clients that need to run the applications locally. Browser access to point-of-sale screens would also allow customers to place orders from their own phones, or remotely. *See* Section VI.A.6 [The Claimed Concepts Were Known, Access to Point of Sale Screens / Browser]. In addition, because browsers were well-known long before the Asserted Patents, using a browser would have allowed Olson's system to use standardized hardware and software, or training by using HTML standard protocol, and lower the cost of POS terminals.

Moreover, the ability to use a browser, and the benefits of doing so, would have been predictable to a POSITA because it would have been a combination of prior art elements according to known methods to yield predictable results, because a POSITA would have known of the general availability of browsers and would have understood that it would have been beneficial and desirable to use browsers to a networked computer system, such as a POS system, like the one disclosed in Olson.

Modifying Olson to use a browser rather than a non-browser program would have been a simple substitution of one known element (a non-browser program) for another (a browser) to obtain predictable results (the benefits of using a browser).

Moreover, modifying Olson to use a browser would have been obvious to try. When implementing a server, an administrator only has limited options—using a non-browser program or a browser. Each solution has its known advantages and disadvantages, and a POSITA would have understood the benefits and advantages of a browser and been motivated to implement a remote web server in order to achieve those benefits and advantages.

Furthermore, using a browser was a known approach for running hardware and software in POS systems, so a POSITA would have been motivated to implement a browser, as taught in the Browser References in the same field, based on design incentives or market forces, including those described in the Browser References. *See* Section VI.A.6 [The Claimed Concepts Were Known, Access to Point of Sale Screens / Browser].

Additionally, a POSITA would have had a reasonable expectation of success in adding this feature to Olson, as Olson already discloses web-based browsers for manager administration. In other words, modifying Olson to use a browser would have merely involved the use of a known technique to improve similar systems in the same way.

c. Flashpoint alone or in combination with one or more Remote Web Server References and one or more Browser References

Flashpoint alone renders obvious each of the features of the ’793 patent, as shown in the attached claim chart. *See* Exhibit C-’793-13.

i. Remote Web Server

The ’793 patent claims recite a web server that is remote from one or more point of sale terminals. *See* ’793 patent, 1[b] (“communicate with one or more POS terminals over a network comprising the Internet, wherein the one or more POS terminals are configured to display one or more POS screens;”), 1[d] (“receive, from at least one of the one or more POS terminals over the network, further information regarding one or more POS transactions corresponding to the one or

more items;"), 1[e] ("configure the one or more POS terminals with the information over the network to create or modify based on the further information regarding one or more POS transactions the one or more POS screens displayed on the one or more POS terminals;"), 1[f] ("wherein the further information regarding the one or more POS transactions, the information used for creating or modifying the one or more POS screens, or a combination thereof comprises one or more of employee clock information, customer add/update information, item add/update information, promotion information, loyalty point information, discount information, taxation information, item cost information, or inventory information;"), 42[a] ("communicating with one or more POS terminals over a network comprising the Internet, wherein the one or more POS terminals are configured to display one or more POS screens;"), 42[c] ("receiving, from at least one of the one or more POS terminals over the network, further information regarding one or more POS transactions corresponding to the one or more items;"), 42[d] ("provisioning the one or more POS terminals with the information over the network to create, or modify based on the further information the one or more POS screens displayed on the one or more POS terminals;"), 44[b] ("communicate with one or more POS terminals over a network comprising the Internet, wherein the one or more POS terminals are configured to display one or more POS screens;"), 44[d] ("receive, from at least one of the one or more POS terminals over the network, further information regarding one or more POS transactions corresponding to the one or more items;"), 44[e] ("configure the one or more POS terminals with the information over the network to create, or modify based on the further information the functionality of the one or more POS terminals;").

This feature is taught by and/or would have been obvious in view of Flashpoint. For example, Flashpoint relates generally to a POS system that offers "integrated software" including "mission-critical point of sale applications and back office management tools" as well as

"enterprise-wide management solutions." Flashpoint, https://web.archive.org/web/20071130010505/http://www.trimpos.net:80/ (Nov. 3, 2007). Restaurants using Flashpoint could use "corporate software" on a remote server at the corporate office to update menus on the remote server. *See* FLASHPOINT™ P.O.S. Software, Flashpoint Operations Guide, System Setup Manual at 4 ("Once the System Set Up button has been selected the following screen will appear, allowing you to complete many of the setups for Flashpoint. If you are using the corporate software, you must select the location to setup before going into system setup."). Flashpoint also discloses a local server computer. *See, e.g.*, Flashpoint, *Quick Food Service*, https://web.archive.org/web/20071031033856/http:/www.trimpos.net/sitefiles/index.html (Oct. 31, 2007) ("Use the back office station for confidential reporting and daily management needs, allowing the POS stations to be dedicated to order processing."); FLASHPOINT™ P.O.S. Software, Flashpoint Operations Guide, System Setup Manual at 48 ("Back Office Station – If set, the station will automatically display the system menu once the employee enters their id. Use this option when the station is sitting in the office and only used to change prices, run reports etc."). Remote servers were well known at the time, as was conducting business over the web. Therefore, this feature would have also been obvious in view of Flashpoint to move a local server to a remote location and shift the system to be web-based.

To the extent CloudofChange contends that Flashpoint fails to disclose or render obvious this feature, a POSITA would have been motivated to modify Flashpoint's system to make the server remote and web-based, as disclosed in the Remote Web Server References. *See* Section VI.A.4 [The Claimed Concepts Were Known, Remote Web Server / Remote Web Server].

Making Flashpoint's server remote and web-based would have realized many benefits. For example, making the server remote would allow a single manager to manage multiple restaurants, reducing hardware costs and simplifying management of multiple restaurants, ensure uniformity across the multiple restaurants, improve the ability to manage from anywhere, and improve the ability to order from anywhere. *See* Section VI.A.4 [The Claimed Concepts Were Known, Remote Web Server / Remote Web Server]. In addition, because web-servers were well-known long before the Asserted Patents, making the remote server web-based would have allowed Flashpoint's system to use standardized hardware and software, or training by using HTML standard protocol, and lower the cost of POS terminals. Making Flashpoint's server a remote web server would also allow a single server to manage multiple restaurants, reducing hardware costs and simplifying management of multiple restaurants, ensure uniformity across the multiple restaurants, improve the ability to manage from anywhere, and improve the ability to order from anywhere.

Moreover, the ability to add a remote web server, and the benefits of doing so, would have been predictable to a POSITA because it would have been a combination of prior art elements according to known methods to yield predictable results, because a POSITA would have known of the general availability of remote web servers and would have understood that it would have been beneficial and desirable to add a remote web server to a networked computer system, such as a POS system, like the one disclosed in Flashpoint.

Modifying Flashpoint to use a remote web server rather than a local web server would have been a simple substitution of one known element (a local server that is maintained by the individual owner) for another (a remote web server) to obtain predictable results (the benefits of using a remote web server).

Moreover, modifying Flashpoint to use a remote web server would have been obvious to try. When implementing a server, an administrator only has limited options—making the server local or remote, making the server web-based or non-web-based. Each solution has its known advantages and disadvantages, and a POSITA would have understood the benefits and advantages of a remote web server and been motivated to implement a remote web server in order to achieve those benefits and advantages.

Furthermore, using a remote web server was a known approach for computing hardware and software in POS systems, and particularly remote server hardware and software, so a POSITA would have been motivated to implement a remote web server, as taught in the Remote Web References in the same field, based on design incentives or market forces, including those described in the Remote Web Server References. *See* Section VI.A.4 [The Claimed Concepts Were Known, Remote Web Server / Remote Web Server].

Additionally, a POSITA would have had a reasonable expectation of success in adding this feature to Flashpoint, as Flashpoint already discloses a corporate software that could manage servers remotely. In other words, modifying Flashpoint to use a remote web server would have merely involved the use of a known technique to improve similar systems in the same way.

ii. Browser

The ’793 patent claims recite access to point of sale screens with browser-based web access. *See* ’793 patent, claim 11 (“The web-based point of sale (POS) builder system of claim 1, wherein the POS builder interface is accessible via a web browser.”).

This feature is taught by and/or would have been obvious in view of Flashpoint. For example, Flashpoint relates generally to a POS system that offers “integrated software” including “mission-critical point of sale applications and back office management tools” as well as “enterprise-wide management solutions.” Flashpoint,

https://web.archive.org/web/20071130010505/http://www.trimpos.net:80/ (Nov. 3, 2007). Flashpoint discloses using the web for online ordering. *See* FLASHPOINT™ P.O.S. Software, Flashpoint Operations Guide, System Setup Manual at 84 ("You can use the web button to setup your location for use with online ordering. For detailed information, see the Setting up your location for online ordering manual."). It would have been obvious to use a browser when ordering on the web.

To the extent CloudofChange contends that Flashpoint fails to disclose or render obvious this feature, a POSITA would have been motivated to modify Flashpoint's system to use a browser, as disclosed in the Browser References. *See* Section VI.A.6 [The Claimed Concepts Were Known, Access to Point of Sale Screens / Browser].

Using Flashpoint's system with a browser would have realized many benefits. For example, using a browser would allow a business to use standard and commoditized software applications, instead of special software or hardware. In addition, browsers enable the use of "thin clients" that do not need to run applications (other than the browser) locally because the processing is largely done at the remote server. Such thin clients would have reduced requirements for memory and processing power, as compared to clients that need to run the applications locally. Browser access to point-of-sale screens would also allow customers to place orders from their own phones, or remotely. *See* Section VI.A.6 [The Claimed Concepts Were Known, Access to Point of Sale Screens / Browser]. In addition, because browsers were well-known long before the Asserted Patents, using a browser would have allowed Flashpoint's system to use standardized hardware and software, or training by using HTML standard protocol, and lower the cost of POS terminals.

Moreover, the ability to use a browser, and the benefits of doing so, would have been predictable to a POSITA because it would have been a combination of prior art elements according to known methods to yield predictable results, because a POSITA would have known of the general availability of browsers and would have understood that it would have been beneficial and desirable to use browsers to a networked computer system, such as a POS system, like the one disclosed in Flashpoint.

Modifying Flashpoint to use a browser rather than a non-browser program would have been a simple substitution of one known element (a non-browser program) for another (a browser) to obtain predictable results (the benefits of using a browser).

Moreover, modifying Flashpoint to use a browser would have been obvious to try. When implementing a server, an administrator only has limited options—using a non-browser program or a browser. Each solution has its known advantages and disadvantages, and a POSITA would have understood the benefits and advantages of a browser and been motivated to implement a remote web server in order to achieve those benefits and advantages.

Furthermore, using a browser was a known approach for running hardware and software in POS systems, so a POSITA would have been motivated to implement a browser, as taught in the Browser References in the same field, based on design incentives or market forces, including those described in the Browser References. *See* Section VI.A.6 [The Claimed Concepts Were Known, Access to Point of Sale Screens / Browser].

Additionally, a POSITA would have had a reasonable expectation of success in adding this feature to Flashpoint, as Flashpoint already discloses ordering on the web. In other words, modifying Flashpoint to use a browser would have merely involved the use of a known technique to improve similar systems in the same way.

d. Halo alone or in combination with one or more Remote Web Server References

Halo alone renders obvious each of the features of the ’793 patent, as shown in the attached claim chart. *See* Exhibit C-’793-14.

i. Remote Web Server

The ’793 patent claims recite a web server that is remote from one or more point of sale terminals. *See* ’793 patent, 1[b] (“communicate with one or more POS terminals over a network comprising the Internet, wherein the one or more POS terminals are configured to display one or more POS screens;”), 1[d] (“receive, from at least one of the one or more POS terminals over the network, further information regarding one or more POS transactions corresponding to the one or more items;”), 1[e] (“configure the one or more POS terminals with the information over the network to create or modify based on the further information regarding one or more POS transactions the one or more POS screens displayed on the one or more POS terminals;”), 1[f] (“wherein the further information regarding the one or more POS transactions, the information used for creating or modifying the one or more POS screens, or a combination thereof comprises one or more of employee clock information, customer add/update information, item add/update information, promotion information, loyalty point information, discount information, taxation information, item cost information, or inventory information;”), 42[a] (“communicating with one or more POS terminals over a network comprising the Internet, wherein the one or more POS terminals are configured to display one or more POS screens;”), 42[c] (“receiving, from at least one of the one or more POS terminals over the network, further information regarding one or more POS transactions corresponding to the one or more items;”), 42[d] ("provisioning the one or more POS terminals with the information over the network to create, or modify based on the further information the one or more POS screens displayed on the one or more POS terminals;”),

44[b] (“communicate with one or more POS terminals over a network comprising the Internet, wherein the one or more POS terminals are configured to display one or more POS screens;”), 44[d] ("receive, from at least one of the one or more POS terminals over the network, further information regarding one or more POS transactions corresponding to the one or more items;”), 44[e] (“configure the one or more POS terminals with the information over the network to create, or modify based on the further information the functionality of the one or more POS terminals;”).

This feature is taught by and/or would have been obvious in view of Halo. For example, Halo relates generally to a POS system that “only requires a POS terminal connected via Internet to a central computer server.” Ahmed H. Medina, *An Analysis of Market Development Strategy of a Point-of-Sale Solutions Provider’s Market Research Database* (Summer 2007) (M.B.A. thesis, Simon Fraser University), 3. Halo further discloses “Halo Enterprise Manager, which allows the restaurant to configure its POS system by setting menu process and promotions. Restaurants can operate the feature through any web browser.” *Id.* Halo further discloses “Halo's secure architecture will support Juice It Up! management and franchisees in making strategic decisions based on real-time information about different regions, menu items and store performance, while letting them view reports and change menu items from anywhere at any time using a Web browser.” *Juice It Up! Moves to Web-based POS*, NATION’S RESTAURANT NEWS (June 26, 2008), https://www.nrn.com/archive/juice-it-moves-web-based-pos, 1. Halo further discloses “Halo's secure architecture will support Juice It Up! management and franchisees in making strategic decisions based on real-time information about different regions, menu items and store performance, while letting them view reports and change menu items from anywhere at any time using a Web browser.” *Juice It Up! Moves to Web-based POS*, NATION’S RESTAURANT NEWS (June 26, 2008), https://www.nrn.com/archive/juice-it-moves-web-based-pos, 1. Halo

discloses that: "With Halo for Restaurants, you have the ability to monitor and manage a single store or your entire chain from anywhere you have access to a web browser. Change prices, add menu items, manage your touch screen terminals and drill down to the transaction level of detail in real time." *Halo For Restaurants*, HALO (Nov. 20, 2006), https://web.archive.org/web/20061120125846/http://www.myhalo.com/section.asp?catid=267, 1. Moreover, Halo discloses that "Halo also integrates multiple locations and enables merchants to compare them in real-time and change items and prices instantly." *Look to the Light: Retail POS System Can Replace ECR,* THE GREEN SHEET ONLINE EDITION, Issue 7:6:1 (June 7, 2007), www.greensheet.com/emagazine.php?article_id=136, 1. Conducting business over the web was well known. It was also well known for a POS terminal to access the web from a remote server. Therefore, this feature would have also been obvious in view of Halo in order to have a web server that is remote from one or more point of sale terminals.

To the extent CloudofChange contends that Halo fails to disclose or render obvious this feature, a POSITA would have been motivated to modify Halo's system to make the server remote and web-based, as disclosed in the Remote Web Server References. *See* Section VI.A.4 [The Claimed Concepts Were Known, Remote Web Server / Remote Web Server].

Making Halo's server remote and web-based would have realized many benefits. For example, making the server remote would allow a single manager to manage multiple restaurants, reducing hardware costs and simplifying management of multiple restaurants, ensure uniformity across the multiple restaurants, improve the ability to manage from anywhere, and improve the ability to order from anywhere. *See* Section VI.A.4 [The Claimed Concepts Were Known, Remote Web Server / Remote Web Server]. In addition, because web-servers were well-known long before the Asserted Patents, making the remote server web-based would have

allowed Halo's system to use standardized hardware and software, or training by using HTML standard protocol, and lower the cost of POS terminals. Making Halo's server a remote web server would also allow a single server to manage multiple restaurants, reducing hardware costs and simplifying management of multiple restaurants, ensure uniformity across the multiple restaurants, improve the ability to manage from anywhere, and improve the ability to order from anywhere.

Moreover, the ability to add a remote web server, and the benefits of doing so, would have been predictable to a POSITA because it would have been a combination of prior art elements according to known methods to yield predictable results, because a POSITA would have known of the general availability of remote web servers and would have understood that it would have been beneficial and desirable to add a remote web server to a networked computer system, such as a POS system, like the one disclosed in Halo.

Modifying Halo to use a remote web server rather than a local web server would have been a simple substitution of one known element (a local server that is maintained by the individual owner) for another (a remote web server) to obtain predictable results (the benefits of using a remote web server).

Moreover, modifying Halo to use a remote web server would have been obvious to try. When implementing a server, an administrator only has limited options—making the server local or remote, making the server web-based or non-web-based. Each solution has its known advantages and disadvantages, and a POSITA would have understood the benefits and advantages of a remote web server and been motivated to implement a remote web server in order to achieve those benefits and advantages.

Furthermore, using a remote web server was a known approach for computing hardware and software in POS systems, and particularly remote server hardware and software, so a POSITA would have been motivated to implement a remote web server, as taught in the Remote Web References in the same field, based on design incentives or market forces, including those described in the Remote Web Server References. *See* Section VI.A.4 [The Claimed Concepts Were Known, Remote Web Server / Remote Web Server].

Additionally, a POSITA would have had a reasonable expectation of success in adding this feature to Halo, as Halo already discloses managing a web server from a remote web browser. In other words, modifying Halo to use a remote web server would have merely involved the use of a known technique to improve similar systems in the same way.

e. Phoenix alone or in combination with one or more POS Builder References and one or more Remote Web Server References

Phoenix alone renders obvious each of the features of the '793 patent, as shown in the attached claim chart. *See* Exhibit C-'793-16.

i. POS Builder

The '793 patent claims recite providing a POS building tool that allows creating menus with no special training. *See* '793 patent, claim limitation 1[c] ("receive, over the network from a POS builder interface, information used for creating or modifying the one or more POS screens including creating or modifying one or more display interfaces for display on the one or more POS screens, the one or more display interfaces being associated with one or more items"), claim 2 ("The web-based point of sale (POS) builder system of claim 1, wherein the POS builder interface is configured to run on a computing device"), claim limitation 42[b] ("receiving, over the network from a POS builder interface, information used for creating or modifying the one or more POS screens including creating or modifying one or more display interfaces for display on

the one or more POS screens, the one or more display interfaces being associated with one or more items"), claim limitation 44[c] ("receive, over the network from a POS builder interface, information for creating or modifying functionality of the one or more POS terminals"). To the extent that CloudofChange contends that Phoenix fails to disclose this feature, a POSITA would have been aware of and would have been motivated to modify the systems and methods of Phoenix to incorporate this feature, or alternatively to combine various teachings from the systems and methods described in one or more of the POS Builder references with Phoenix into one system or method.

This feature is taught by and/or would have been obvious in view of Phoenix. For example, Phoenix relates generally to a POS system with "[c]ash control, easy order taking, integrated delivery features, full table-service capabilities, complete driver and server management, inventory, marketing, and more." FireFly, *POS Software Showcase '07*, Supplement to Hospitality Technology, at 5. Phoenix also discloses, "you can add a menu item or change a price in the middle of your dinner rush and have it take effect instantly, without having to touch any of your workstations." *FireFly Technologies Introduces Phoenix Point-of-Sale*, https://www.pizzamarketplace.com/news/firefly-technologies-introduces-phoenix-point-of-sale/ (Mar 13, 2006). Furthermore, a POS building tool that allows creating and modifying menus with no special training was a well-known feature of POS systems prior to the patent priority date, and was well-known for its ability to reduce implementation and maintenance costs of POS systems by reducing or eliminating the requirement for specialized IT support (e.g., programmers) to set up or maintain a POS system. Accordingly, a POSITA would have been motivated to provide Phoenix's POS system with a POS builder tool that allows creating and

modifying menus with no special training, and would have had a reasonable expectation of success in doing so.

To the extent CloudofChange contends that Phoenix fails to disclose or render obvious this feature, a POSITA would have been motivated to add a POS builder tool to Phoenix's system, as disclosed in the POS Builder references. *See* Section VI.A.3 [The Claimed Concepts Were Known, POS Builder References].

Adding a POS builder tool would have realized the many benefits of POS builder tools. For example, adding a POS builder tool would improve Phoenix's system by making Phoenix's system—including Phoenix's POS terminals—easier to install and manage, lowering the cost of installation and maintenance, reducing or eliminating special training to manage (e.g., allowing any employee—rather than just IT—to add or update menus or other aspects of the POS terminals), and lowering IT costs. *See* Section VI.A.3 [The Claimed Concepts Were Known, POS Builder References].

Moreover, the ability to add a POS building tool, and the benefits of doing so, would have been predictable to a POSITA because it would have been a combination of prior art elements according to known methods to yield predictable results, because a POSITA would have known of the general availability of POS builder tools, and would have understood that it would have been beneficial and desirable to add a POS builder tool to a POS system, like the one disclosed in Phoenix.

Adding a POS builder tool would have been a simple substitution of one known element (a POS system that requires programmers to manually build) for another (a POS system with a POS builder tool that anyone can use without special training) to obtain predictable results (the benefits of a POS builder tool).

Moreover, adding a POS builder tool would have been obvious to try. When implementing a POS system, an administrator only has limited options—configuring menus by hiring a programmer or other IT professional to manually program them, or configuring menus with a POS builder tool that requires no special training, and thus is usable by any employee, including said manager. Each solution has its known advantages, and a POSITA would have understood the benefits and advantages of a POS builder tool and been motivated to implement a POS builder tool in order to achieve those benefits and advantages.

Furthermore, using a POS builder tool was a known approach for POS systems, so a POSITA would have been motivated to implement the teachings of a POS builder tool in the same field (e.g., the teachings of POS builder tools in the POS Builder References, which are in the same field as Phoenix) based on design incentives or market forces, including those described in the POS Builder References. *See* Section VI.A.3 [The Claimed Concepts Were Known, POS Builder References].

Additionally, a POSITA would have had a reasonable expectation of success in adding this feature to Phoenix. Phoenix's disclosure of these features is in common with a system that includes a POS builder tool that requires no special training. In other words, adding a POS builder tool would have merely involved the use of a known technique to improve similar systems in the same way.

ii. Remote Web Server

The '793 patent claims recite a web server that is remote from one or more point of sale terminals. *See* '793 patent, 1[b] ("communicate with one or more POS terminals over a network comprising the Internet, wherein the one or more POS terminals are configured to display one or more POS screens;"), 1[d] ("receive, from at least one of the one or more POS terminals over the network, further information regarding one or more POS transactions corresponding to the one or

more items;"), 1[e] ("configure the one or more POS terminals with the information over the network to create or modify based on the further information regarding one or more POS transactions the one or more POS screens displayed on the one or more POS terminals;"), 1[f] ("wherein the further information regarding the one or more POS transactions, the information used for creating or modifying the one or more POS screens, or a combination thereof comprises one or more of employee clock information, customer add/update information, item add/update information, promotion information, loyalty point information, discount information, taxation information, item cost information, or inventory information;"), 42[a] ("communicating with one or more POS terminals over a network comprising the Internet, wherein the one or more POS terminals are configured to display one or more POS screens;"), 42[c] ("receiving, from at least one of the one or more POS terminals over the network, further information regarding one or more POS transactions corresponding to the one or more items;"), 42[d] ("provisioning the one or more POS terminals with the information over the network to create, or modify based on the further information the one or more POS screens displayed on the one or more POS terminals;"), 44[b] ("communicate with one or more POS terminals over a network comprising the Internet, wherein the one or more POS terminals are configured to display one or more POS screens;"), 44[d] ("receive, from at least one of the one or more POS terminals over the network, further information regarding one or more POS transactions corresponding to the one or more items;"), 44[e] ("configure the one or more POS terminals with the information over the network to create, or modify based on the further information the functionality of the one or more POS terminals;").

This feature is taught by and/or would have been obvious in view of Phoenix. For example, Phoenix relates generally to a POS system with "[c]ash control, easy order taking, integrated delivery features, full table-service capabilities, complete driver and server

management, inventory, marketing, and more." FireFly, *POS Software Showcase '07*, Supplement to Hospitality Technology, at 5. Phoenix discloses "Web-based technology [that] lets Phoenix Point-of-Sale users get more done in less time with easy anytime, anywhere access to their stores, without tying up an order terminal, without waiting for data updates, and without notifying onsite staff. Live access lets you update prices, run payroll, review reports, even take an order." Conducting business over the web was well known. It was also well known for a POS terminal to access the web from a remote server. Therefore, this feature would have also been obvious in view of Phoenix in order to have a web server that is remote from one or more point of sale terminals.

To the extent CloudofChange contends that Phoenix fails to disclose or render obvious this feature, a POSITA would have been motivated to modify Phoenix's system to make the server remote and web-based, as disclosed in the Remote Web Server References. *See* Section VI.A.4 [The Claimed Concepts Were Known, Remote Web Server / Remote Web Server].

Making Phoenix's server remote and web-based would have realized many benefits. For example, making the server remote would allow a single manager to manage multiple restaurants, reducing hardware costs and simplifying management of multiple restaurants, ensure uniformity across the multiple restaurants, improve the ability to manage from anywhere, and improve the ability to order from anywhere. *See* Section VI.A.4 [The Claimed Concepts Were Known, Remote Web Server / Remote Web Server]. In addition, because web-servers were well-known long before the Asserted Patents, making the remote server web-based would have allowed Phoenix's system to use standardized hardware and software, or training by using HTML standard protocol, and lower the cost of POS terminals. Making Phoenix's server a remote web server would also allow a single server to manage multiple restaurants, reducing

hardware costs and simplifying management of multiple restaurants, ensure uniformity across the multiple restaurants, improve the ability to manage from anywhere, and improve the ability to order from anywhere.

Moreover, the ability to add a remote web server, and the benefits of doing so, would have been predictable to a POSITA because it would have been a combination of prior art elements according to known methods to yield predictable results, because a POSITA would have known of the general availability of remote web servers and would have understood that it would have been beneficial and desirable to add a remote web server to a networked computer system, such as a POS system, like the one disclosed in Phoenix.

Modifying Phoenix to use a remote web server rather than a local web server would have been a simple substitution of one known element (a local server that is maintained by the individual owner) for another (a remote web server) to obtain predictable results (the benefits of using a remote web server).

Moreover, modifying Phoenix to use a remote web server would have been obvious to try. When implementing a server, an administrator only has limited options—making the server local or remote, making the server web-based or non-web-based. Each solution has its known advantages and disadvantages, and a POSITA would have understood the benefits and advantages of a remote web server and been motivated to implement a remote web server in order to achieve those benefits and advantages.

Furthermore, using a remote web server was a known approach for computing hardware and software in POS systems, and particularly remote server hardware and software, so a POSITA would have been motivated to implement a remote web server, as taught in the Remote Web References in the same field, based on design incentives or market forces, including those

described in the Remote Web Server References. *See* Section VI.A.4 [The Claimed Concepts Were Known, Remote Web Server / Remote Web Server].

Additionally, a POSITA would have had a reasonable expectation of success in adding this feature to Phoenix, as Phoenix already discloses managing a web server from a remote web browser. In other words, modifying Phoenix to use a remote web server would have merely involved the use of a known technique to improve similar systems in the same way.

f. InfoGenesis alone or in combination with one or more Remote Web Server References and one or more Browser References

InfoGenesis alone renders obvious each of the features of the '793 patent, as shown in the attached claim chart. *See* Exhibit C-'793-19.

i. Remote Web Server

The '793 patent claims recite a web server that is remote from one or more point of sale terminals. *See* '793 patent, 1[b] ("communicate with one or more POS terminals over a network comprising the Internet, wherein the one or more POS terminals are configured to display one or more POS screens;"), 1[d] ("receive, from at least one of the one or more POS terminals over the network, further information regarding one or more POS transactions corresponding to the one or more items;"), 1[e] ("configure the one or more POS terminals with the information over the network to create or modify based on the further information regarding one or more POS transactions the one or more POS screens displayed on the one or more POS terminals;"), 1[f] ("wherein the further information regarding the one or more POS transactions, the information used for creating or modifying the one or more POS screens, or a combination thereof comprises one or more of employee clock information, customer add/update information, item add/update information, promotion information, loyalty point information, discount information, taxation information, item cost information, or inventory information;"), 42[a] ("communicating with one

or more POS terminals over a network comprising the Internet, wherein the one or more POS terminals are configured to display one or more POS screens;"), 42[c] ("receiving, from at least one of the one or more POS terminals over the network, further information regarding one or more POS transactions corresponding to the one or more items;"), 42[d] ("provisioning the one or more POS terminals with the information over the network to create, or modify based on the further information the one or more POS screens displayed on the one or more POS terminals;"), 44[b] ("communicate with one or more POS terminals over a network comprising the Internet, wherein the one or more POS terminals are configured to display one or more POS screens;"), 44[d] ("receive, from at least one of the one or more POS terminals over the network, further information regarding one or more POS transactions corresponding to the one or more items;"), 44[e] ("configure the one or more POS terminals with the information over the network to create, or modify based on the further information the functionality of the one or more POS terminals;").

This feature is taught by and/or would have been obvious in view of InfoGenesis. For example, InfoGenesis relates generally to "point-of-sale software that combines powerful reporting and configuration capabilities in the back office with an easy-to-use, touch screen Windows Terminal Application (WTA) that makes better service for frontline staff second nature." InfoGenesis, Point-of-Sale, https://web.archive.org/web/20070416203641/http://www.infogenesis.com/restaurants/pointofsalepos.aspx (Mar. 11, 2006), at 1. InfoGenesis discloses that "InfoGenesis hosts Infogenesis as an application service provider (ASP) solution from a secure data center in California and maintains Internet connectivity with each of the three restaurants, processing all outlet and site transactions in a real-time environment." William Atkinson, Point of Service: Now and Later of Restaurant POS (Apr. 9, 2007). InfoGenesis further discloses that "This approach allows the restaurant to

receive the benefits of POS without having to hire a support staff or install server hardware." *Id.* InfoGenesis further discloses that: "In addition, the system integrates with those the restaurants have from other vendors." *Id.* Conducting business over the web was well known. It was also well known for a POS terminal to access the web from a remote server. Therefore, this feature would have also been obvious in view of InfoGenesis in order to have a web server that is remote from one or more point of sale terminals.

To the extent CloudofChange contends that InfoGenesis fails to disclose or render obvious this feature, a POSITA would have been motivated to modify InfoGenesis's system to make the server remote and web-based, as disclosed in the Remote Web Server References. *See* Section VI.A.4 [The Claimed Concepts Were Known, Remote Web Server / Remote Web Server].

Making InfoGenesis's server remote and web-based would have realized many benefits. For example, making the server remote would allow a single manager to manage multiple restaurants, reducing hardware costs and simplifying management of multiple restaurants, ensure uniformity across the multiple restaurants, improve the ability to manage from anywhere, and improve the ability to order from anywhere. *See* Section VI.A.4 [The Claimed Concepts Were Known, Remote Web Server / Remote Web Server]. In addition, because web-servers were well-known long before the Asserted Patents, making the remote server web-based would have allowed InfoGenesis's system to use standardized hardware and software, or training by using HTML standard protocol, and lower the cost of POS terminals. Making InfoGenesis's server a remote web server would also allow a single server to manage multiple restaurants, reducing hardware costs and simplifying management of multiple restaurants, ensure uniformity across the

multiple restaurants, improve the ability to manage from anywhere, and improve the ability to order from anywhere.

Moreover, the ability to add a remote web server, and the benefits of doing so, would have been predictable to a POSITA because it would have been a combination of prior art elements according to known methods to yield predictable results, because a POSITA would have known of the general availability of remote web servers and would have understood that it would have been beneficial and desirable to add a remote web server to a networked computer system, such as a POS system, like the one disclosed in InfoGenesis.

Modifying InfoGenesis to use a remote web server rather than a local web server would have been a simple substitution of one known element (a local server that is maintained by the individual owner) for another (a remote web server) to obtain predictable results (the benefits of using a remote web server).

Moreover, modifying InfoGenesis to use a remote web server would have been obvious to try. When implementing a server, an administrator only has limited options—making the server local or remote, making the server web-based or non-web-based. Each solution has its known advantages and disadvantages, and a POSITA would have understood the benefits and advantages of a remote web server and been motivated to implement a remote web server in order to achieve those benefits and advantages.

Furthermore, using a remote web server was a known approach for computing hardware and software in POS systems, and particularly remote server hardware and software, so a POSITA would have been motivated to implement a remote web server, as taught in the Remote Web References in the same field, based on design incentives or market forces, including those

described in the Remote Web Server References. *See* Section VI.A.4 [The Claimed Concepts Were Known, Remote Web Server / Remote Web Server].

Additionally, a POSITA would have had a reasonable expectation of success in adding this feature to InfoGenesis, as InfoGenesis already discloses managing a web server from a remote web browser. In other words, modifying InfoGenesis to use a remote web server would have merely involved the use of a known technique to improve similar systems in the same way.

ii. Browser

The '793 patent claims recite access to point of sale screens with browser-based web access. *See* '793 patent, claim 11 ("The web-based point of sale (POS) builder system of claim 1, wherein the POS builder interface is accessible via a web browser.").

This feature is taught by and/or would have been obvious in view of InfoGenesis. For example, InfoGenesis relates generally to "point-of-sale software that combines powerful reporting and configuration capabilities in the back office with an easy-to-use, touch screen Windows Terminal Application (WTA) that makes better service for frontline staff second nature." InfoGenesis, Point-of-Sale, https://web.archive.org/web/20070416203641/http://www.infogenesis.com/restaurants/pointofsalepos.aspx (Mar. 11, 2006), at 1. InfoGenesis further discloses "touch screen Windows Terminal Application (WTA)." It would have been obvious to use a browser instead of a WTA, as both were known concepts of how to run a POS system. Furthermore, InfoGenesis discloses "a Web-based solution that [restaurants] employ to track sales, labor and food costs in a real-time environment." William Atkinson, *Point of Service: Now and Later of Restaurant POS* (Apr. 9, 2007). Thus, the InfoGenesis technology already used web-based solutions, which used a browser.

To the extent CloudofChange contends that InfoGenesis fails to disclose or render obvious this feature, a POSITA would have been motivated to modify InfoGenesis's system to use a browser, as disclosed in the Browser References. *See* Section VI.A.6 [The Claimed Concepts Were Known, Access to Point of Sale Screens / Browser].

Using InfoGenesis's system with a browser would have realized many benefits. For example, using a browser would allow a business to use standard and commoditized software applications, instead of special software or hardware. In addition, browsers enable the use of "thin clients" that do not need to run applications (other than the browser) locally because the processing is largely done at the remote server. Such thin clients would have reduced requirements for memory and processing power, as compared to clients that need to run the applications locally. Browser access to point-of-sale screens would also allow customers to place orders from their own phones, or remotely. *See* Section VI.A.6 [The Claimed Concepts Were Known, Access to Point of Sale Screens / Browser]. In addition, because browsers were well-known long before the Asserted Patents, using a browser would have allowed InfoGenesis's system to use standardized hardware and software, or training by using HTML standard protocol, and lower the cost of POS terminals.

Moreover, the ability to use a browser, and the benefits of doing so, would have been predictable to a POSITA because it would have been a combination of prior art elements according to known methods to yield predictable results, because a POSITA would have known of the general availability of browsers and would have understood that it would have been beneficial and desirable to use browsers to a networked computer system, such as a POS system, like the one disclosed in InfoGenesis.

Modifying InfoGenesis to use a browser rather than a non-browser program would have been a simple substitution of one known element (a non-browser program) for another (a browser) to obtain predictable results (the benefits of using a browser).

Moreover, modifying InfoGenesis to use a browser would have been obvious to try. When implementing a server, an administrator only has limited options—using a non-browser program or a browser. Each solution has its known advantages and disadvantages, and a POSITA would have understood the benefits and advantages of a browser and been motivated to implement a remote web server in order to achieve those benefits and advantages.

Furthermore, using a browser was a known approach for running hardware and software in POS systems, so a POSITA would have been motivated to implement a browser, as taught in the Browser References in the same field, based on design incentives or market forces, including those described in the Browser References. *See* Section VI.A.6 [The Claimed Concepts Were Known, Access to Point of Sale Screens / Browser].

Additionally, a POSITA would have had a reasonable expectation of success in adding this feature to InfoGenesis, as InfoGenesis already discloses a WTA, as well as other web-based solutions. In other words, modifying InfoGenesis to use a browser would have merely involved the use of a known technique to improve similar systems in the same way.

g. Simphony alone or in combination with one or more POS Builder References and one or more Remote Web Server References and one or more Browser References

Simphony alone renders obvious each of the features of the ’793 patent, as shown in the attached claim chart. *See* Exhibit C-’793-20.

i. POS Builder

The ’793 patent claims recite providing a POS building tool that allows creating menus with no special training. *See* ’793 patent, claim limitation 1[c] (“receive, over the network from

a POS builder interface, information used for creating or modifying the one or more POS screens including creating or modifying one or more display interfaces for display on the one or more POS screens, the one or more display interfaces being associated with one or more items"), claim 2 ("The web-based point of sale (POS) builder system of claim 1, wherein the POS builder interface is configured to run on a computing device"), claim limitation 42[b] ("receiving, over the network from a POS builder interface, information used for creating or modifying the one or more POS screens including creating or modifying one or more display interfaces for display on the one or more POS screens, the one or more display interfaces being associated with one or more items"), claim limitation 44[c] ("receive, over the network from a POS builder interface, information for creating or modifying functionality of the one or more POS terminals"). To the extent that CloudofChange contends that Simphony fails to disclose this feature, a POSITA would have been aware of and would have been motivated to modify the systems and methods of Simphony to incorporate this feature, or alternatively to combine various teachings from the systems and methods described in one or more of the POS Builder references with Simphony into one system or method.

This feature is taught by and/or would have been obvious in view of Simphony. For example, Simphony relates generally to a "point-of-sale (POS) hospitality product." Micros Simphony TM, https://web.archive.org/web/20071112050945/http://www.micros.com:80/Products/Simphony/ (2007). Simphony also discloses "its ability to scale to a business with thousands of workstations spread over many properties. . . . The Enterprise Management Console allows for the management of the entire system from within a single application. Users are able to create and define the parameters of the enterprise, properties, and revenue centers from any PC that has

access to the central server, thus allowing local users to make changes to their configuration if necessary." *Id.* Furthermore, a POS building tool that allows creating and modifying menus with no special training was a well-known feature of POS systems prior to the patent priority date, and was well-known for its ability to reduce implementation and maintenance costs of POS systems by reducing or eliminating the requirement for specialized IT support (e.g., programmers) to set up or maintain a POS system. Accordingly, a POSITA would have been motivated to provide Simphony's POS system with a POS builder tool that allows creating and modifying menus with no special training, and would have had a reasonable expectation of success in doing so.

To the extent CloudofChange contends that Simphony fails to disclose or render obvious this feature, a POSITA would have been motivated to add a POS builder tool to Simphony's system, as disclosed in the POS Builder references. *See* Section VI.A.2 [The Claimed Concepts Were Known, POS Builder References].

Adding a POS builder tool would have realized the many benefits of POS builder tools. For example, adding a POS builder tool would improve Simphony's system by making Simphony's system—including Simphony's POS terminals—easier to install and manage, lowering the cost of installation and maintenance, reducing or eliminating special training to manage (e.g., allowing any employee—rather than just IT—to add or update menus or other aspects of the POS terminals), and lowering IT costs. *See* Section VI.A.2 [The Claimed Concepts Were Known, POS Builder References].

Moreover, the ability to add a POS building tool, and the benefits of doing so, would have been predictable to a POSITA because it would have been a combination of prior art elements according to known methods to yield predictable results, because a POSITA would

have known of the general availability of POS builder tools, and would have understood that it would have been beneficial and desirable to add a POS builder tool to a POS system, like the one disclosed in Simphony.

Adding a POS builder tool would have been a simple substitution of one known element (a POS system that requires programmers to manually build) for another (a POS system with a POS builder tool that anyone can use without special training) to obtain predictable results (the benefits of a POS builder tool).

Moreover, adding a POS builder tool would have been obvious to try. When implementing a POS system, an administrator only has limited options—configuring menus by hiring a programmer or other IT professional to manually program them, or configuring menus with a POS builder tool that requires no special training, and thus is usable by any employee, including said manager. Each solution has its known advantages, and a POSITA would have understood the benefits and advantages of a POS builder tool and been motivated to implement a POS builder tool in order to achieve those benefits and advantages.

Furthermore, using a POS builder tool was a known approach for POS systems, so a POSITA would have been motivated to implement the teachings of a POS builder tool in the same field (e.g., the teachings of POS builder tools in the POS Builder References, which are in the same field as Simphony) based on design incentives or market forces, including those described in the POS Builder References. *See* Section VI.A.2 [The Claimed Concepts Were Known, POS Builder References].

Additionally, a POSITA would have had a reasonable expectation of success in adding this feature to Simphony. Simphony's disclosure of these features is in common with a system that includes a POS builder tool that requires no special training. In other words, adding a POS

builder tool would have merely involved the use of a known technique to improve similar systems in the same way.

ii. Remote Web Server

The '793 patent claims recite a web server that is remote from one or more point of sale terminals. *See* '793 patent, 1[b] ("communicate with one or more POS terminals over a network comprising the Internet, wherein the one or more POS terminals are configured to display one or more POS screens;"), 1[d] ("receive, from at least one of the one or more POS terminals over the network, further information regarding one or more POS transactions corresponding to the one or more items;"), 1[e] ("configure the one or more POS terminals with the information over the network to create or modify based on the further information regarding one or more POS transactions the one or more POS screens displayed on the one or more POS terminals;"), 1[f] ("wherein the further information regarding the one or more POS transactions, the information used for creating or modifying the one or more POS screens, or a combination thereof comprises one or more of employee clock information, customer add/update information, item add/update information, promotion information, loyalty point information, discount information, taxation information, item cost information, or inventory information;"), 42[a] ("communicating with one or more POS terminals over a network comprising the Internet, wherein the one or more POS terminals are configured to display one or more POS screens;"), 42[c] ("receiving, from at least one of the one or more POS terminals over the network, further information regarding one or more POS transactions corresponding to the one or more items;"), 42[d] ("provisioning the one or more POS terminals with the information over the network to create, or modify based on the further information the one or more POS screens displayed on the one or more POS terminals;"), 44[b] ("communicate with one or more POS terminals over a network comprising the Internet, wherein the one or more POS terminals are configured to display one or more POS screens;"),

44[d] ("receive, from at least one of the one or more POS terminals over the network, further information regarding one or more POS transactions corresponding to the one or more items;"), 44[e] ("configure the one or more POS terminals with the information over the network to create, or modify based on the further information the functionality of the one or more POS terminals;").

This feature is taught by and/or would have been obvious in view of Simphony. For example, Simphony relates generally to a "point-of-sale (POS) hospitality product." Micros Simphony TM, https://web.archive.org/web/20071112050945/http://www.micros.com:80/Products/Simphony/ (2007). Simphony discloses that: "Although Simphony may be implemented at a single property, the true value can be found in its ability to scale to a business with thousands of workstations spread over many properties." Micros Simphony TM, https://web.archive.org/web/20071112050945/http://www.micros.com:80/Products/Simphony/ (2007). Simphony further discloses that "[t]his solution can be hosted using one of MICROS's four worldwide datacenters, or it can be self-hosted by a customer." *Id.* Simphony further discloses that "[t]he Enterprise Management Console allows for the management of the entire system from within a single application." *Id.* Simphony further discloses that "[us]sers are able to create and define the parameters of the enterprise, properties, and revenue centers from any PC that has access to the central server, thus allowing local users to make changes to their configuration if necessary." *Id.* Simphony further discloses that: "Simphony is MICROS's latest enterprise class hospitality POS product designed specifically for the leisure and entertainment market." Micros Simphony, https://web.archive.org/web/20060324063129/http://www.micros.com/Products/Simphony/ (2006). Simphony discloses that "Simphony is capable of being centrally-hosted using

MICROS's world-class hosting services or it can be self-hosted by a customer." *Id.* Simphony discloses that it "Simphony can handle a multi-property POS configuration consisting of thousands of workstations generating terabytes of sales data, and its web-based reporting solution minimizes the technology footprint on property." *Id.* Conducting business over the web was well known. It was also well known for a POS terminal to access the web from a remote server. Therefore, this feature would have also been obvious in view of Simphony in order to have a web server that is remote from one or more point of sale terminals.

To the extent CloudofChange contends that Simphony fails to disclose or render obvious this feature, a POSITA would have been motivated to modify Simphony's system to make the server remote and web-based, as disclosed in the Remote Web Server References. *See* Section VI.A.4 [The Claimed Concepts Were Known, Remote Web Server / Remote Web Server].

Making Simphony's server remote and web-based would have realized many benefits. For example, making the server remote would allow a single manager to manage multiple restaurants, reducing hardware costs and simplifying management of multiple restaurants, ensure uniformity across the multiple restaurants, improve the ability to manage from anywhere, and improve the ability to order from anywhere. *See* Section VI.A.4 [The Claimed Concepts Were Known, Remote Web Server / Remote Web Server]. In addition, because web-servers were well-known long before the Asserted Patents, making the remote server web-based would have allowed Simphony's system to use standardized hardware and software, or training by using HTML standard protocol, and lower the cost of POS terminals. Making Simphony's server a remote web server would also allow a single server to manage multiple restaurants, reducing hardware costs and simplifying management of multiple restaurants, ensure uniformity across the

multiple restaurants, improve the ability to manage from anywhere, and improve the ability to order from anywhere.

Moreover, the ability to add a remote web server, and the benefits of doing so, would have been predictable to a POSITA because it would have been a combination of prior art elements according to known methods to yield predictable results, because a POSITA would have known of the general availability of remote web servers and would have understood that it would have been beneficial and desirable to add a remote web server to a networked computer system, such as a POS system, like the one disclosed in Simphony.

Modifying Simphony to use a remote web server rather than a local web server would have been a simple substitution of one known element (a local server that is maintained by the individual owner) for another (a remote web server) to obtain predictable results (the benefits of using a remote web server).

Moreover, modifying Simphony to use a remote web server would have been obvious to try. When implementing a server, an administrator only has limited options—making the server local or remote, making the server web-based or non-web-based. Each solution has its known advantages and disadvantages, and a POSITA would have understood the benefits and advantages of a remote web server and been motivated to implement a remote web server in order to achieve those benefits and advantages.

Furthermore, using a remote web server was a known approach for computing hardware and software in POS systems, and particularly remote server hardware and software, so a POSITA would have been motivated to implement a remote web server, as taught in the Remote Web References in the same field, based on design incentives or market forces, including those

described in the Remote Web Server References. *See* Section VI.A.4 [The Claimed Concepts Were Known, Remote Web Server / Remote Web Server].

Additionally, a POSITA would have had a reasonable expectation of success in adding this feature to Simphony, as Simphony already discloses managing a web server from a remote web browser. In other words, modifying Simphony to use a remote web server would have merely involved the use of a known technique to improve similar systems in the same way.

iii. Browser

The '793 patent claims recite access to point of sale screens with browser-based web access. *See* '793 patent, claim 11 ("The web-based point of sale (POS) builder system of claim 1, wherein the POS builder interface is accessible via a web browser.").

This feature is taught by and/or would have been obvious in view of Simphony. For example, Simphony relates generally to a "point-of-sale (POS) hospitality product." Micros Simphony TM, https://web.archive.org/web/20071112050945/http://www.micros.com:80/Products/Simphony/ (2007). Simphony further discloses that "Simphony can handle a multi-property POS configuration consisting of thousands of workstations generating terabytes of sales data" and that it is a "web-based reporting solution." Micros Simphony, https://web.archive.org/web/20060324063129/http://www.micros.com/Products/Simphony/ (2006). It would have been obvious to use a browser for web-based solutions.

To the extent CloudofChange contends that Simphony fails to disclose or render obvious this feature, a POSITA would have been motivated to modify Simphony's system to use a browser, as disclosed in the Browser References. *See* Section VI.A.6 [The Claimed Concepts Were Known, Access to Point of Sale Screens / Browser].

Using Simphony's system with a browser would have realized many benefits. For example, using a browser would allow a business to use standard and commoditized software applications, instead of special software or hardware. In addition, browsers enable the use of "thin clients" that do not need to run applications (other than the browser) locally because the processing is largely done at the remote server. Such thin clients would have reduced requirements for memory and processing power, as compared to clients that need to run the applications locally. Browser access to point-of-sale screens would also allow customers to place orders from their own phones, or remotely. *See* Section VI.A.6 [The Claimed Concepts Were Known, Access to Point of Sale Screens / Browser]. In addition, because browsers were well-known long before the Asserted Patents, using a browser would have allowed Simphony's system to use standardized hardware and software, or training by using HTML standard protocol, and lower the cost of POS terminals.

Moreover, the ability to use a browser, and the benefits of doing so, would have been predictable to a POSITA because it would have been a combination of prior art elements according to known methods to yield predictable results, because a POSITA would have known of the general availability of browsers and would have understood that it would have been beneficial and desirable to use browsers to a networked computer system, such as a POS system, like the one disclosed in Simphony.

Modifying Simphony to use a browser rather than a non-browser program would have been a simple substitution of one known element (a non-browser program) for another (a browser) to obtain predictable results (the benefits of using a browser).

Moreover, modifying Simphony to use a browser would have been obvious to try. When implementing a server, an administrator only has limited options—using a non-browser program

or a browser. Each solution has its known advantages and disadvantages, and a POSITA would have understood the benefits and advantages of a browser and been motivated to implement a remote web server in order to achieve those benefits and advantages.

Furthermore, using a browser was a known approach for running hardware and software in POS systems, so a POSITA would have been motivated to implement a browser, as taught in the Browser References in the same field, based on design incentives or market forces, including those described in the Browser References. *See* Section VI.A.6 [The Claimed Concepts Were Known, Access to Point of Sale Screens / Browser].

Additionally, a POSITA would have had a reasonable expectation of success in adding this feature to Simphony, as Simphony already discloses web-based solutions. In other words, modifying Simphony to use a browser would have merely involved the use of a known technique to improve similar systems in the same way.

h. Squirrel alone or in combination with one or more Remote Web Server References and one or more Browser References

Squirrel alone renders obvious each of the features of the ’793 patent, as shown in the attached claim chart. *See* Exhibit C-’793-21.

i. Remote Web Server

The ’793 patent claims recite a web server that is remote from one or more point of sale terminals. *See* ’793 patent, 1[b] (“communicate with one or more POS terminals over a network comprising the Internet, wherein the one or more POS terminals are configured to display one or more POS screens;”), 1[d] (“receive, from at least one of the one or more POS terminals over the network, further information regarding one or more POS transactions corresponding to the one or more items;”), 1[e] (“configure the one or more POS terminals with the information over the network to create or modify based on the further information regarding one or more POS

transactions the one or more POS screens displayed on the one or more POS terminals;"), 1[f] ("wherein the further information regarding the one or more POS transactions, the information used for creating or modifying the one or more POS screens, or a combination thereof comprises one or more of employee clock information, customer add/update information, item add/update information, promotion information, loyalty point information, discount information, taxation information, item cost information, or inventory information;"), 42[a] ("communicating with one or more POS terminals over a network comprising the Internet, wherein the one or more POS terminals are configured to display one or more POS screens;"), 42[c] ("receiving, from at least one of the one or more POS terminals over the network, further information regarding one or more POS transactions corresponding to the one or more items;"), 42[d] ("provisioning the one or more POS terminals with the information over the network to create, or modify based on the further information the one or more POS screens displayed on the one or more POS terminals;"), 44[b] ("communicate with one or more POS terminals over a network comprising the Internet, wherein the one or more POS terminals are configured to display one or more POS screens;"), 44[d] ("receive, from at least one of the one or more POS terminals over the network, further information regarding one or more POS transactions corresponding to the one or more items;"), 44[e] ("configure the one or more POS terminals with the information over the network to create, or modify based on the further information the functionality of the one or more POS terminals;").

This feature is taught by and/or would have been obvious in view of Squirrel. For example, Squirrel relates generally to a "Restaurant Point of Sale and Management system." Squirrelsystems, *SquirrelOne Solutions*, https://web.archive.org/web/20051201101248/http://www.squirrelsystems.com:80/solutions/index.html (Dec. 1, 2005). Squirrel further discloses a "32-bit client/server application" that "is fully

integrated with the latest Microsoft environment." *Id.* Squirrel also discloses that its POS Manager was "[b]uilt on the latest Microsoft SQL Server relational database" "with a familiar browser style desktop [that] is able to be accessed directly from the central PC or remotely via the internet." Squirrelsystems, *Corporate POS*, https://web.archive.org/web/20060327174026/http://www.squirrelsystems.com/solutions/corppos.html (Mar. 27, 2006). Remote servers were well known at the time, as was conducting business over the web. Therefore, this feature would have also been obvious in view of Squirrel to move a local server to a remote location and shift the system to be web-based.

To the extent CloudofChange contends that Squirrel fails to disclose or render obvious this feature, a POSITA would have been motivated to modify Squirrel's system to make the server remote and web-based, as disclosed in the Remote Web Server References. *See* Section VI.A.4 [The Claimed Concepts Were Known, Remote Web Server / Remote Web Server].

Making Squirrel's server remote and web-based would have realized many benefits. For example, making the server remote would allow a single manager to manage multiple restaurants, reducing hardware costs and simplifying management of multiple restaurants, ensure uniformity across the multiple restaurants, improve the ability to manage from anywhere, and improve the ability to order from anywhere. *See* Section VI.A.4 [The Claimed Concepts Were Known, Remote Web Server / Remote Web Server]. In addition, because web-servers were well-known long before the Asserted Patents, making the remote server web-based would have allowed Squirrel's system to use standardized hardware and software, or training by using HTML standard protocol, and lower the cost of POS terminals. Making Squirrel's server a remote web server would also allow a single server to manage multiple restaurants, reducing hardware costs and simplifying management of multiple restaurants, ensure uniformity across the

multiple restaurants, improve the ability to manage from anywhere, and improve the ability to order from anywhere.

Moreover, the ability to add a remote web server, and the benefits of doing so, would have been predictable to a POSITA because it would have been a combination of prior art elements according to known methods to yield predictable results, because a POSITA would have known of the general availability of remote web servers and would have understood that it would have been beneficial and desirable to add a remote web server to a networked computer system, such as a POS system, like the one disclosed in Squirrel.

Modifying Squirrel to use a remote web server rather than a local web server would have been a simple substitution of one known element (a local server that is maintained by the individual owner) for another (a remote web server) to obtain predictable results (the benefits of using a remote web server).

Moreover, modifying Squirrel to use a remote web server would have been obvious to try. When implementing a server, an administrator only has limited options—making the server local or remote, making the server web-based or non-web-based. Each solution has its known advantages and disadvantages, and a POSITA would have understood the benefits and advantages of a remote web server and been motivated to implement a remote web server in order to achieve those benefits and advantages.

Furthermore, using a remote web server was a known approach for computing hardware and software in POS systems, and particularly remote server hardware and software, so a POSITA would have been motivated to implement a remote web server, as taught in the Remote Web References in the same field, based on design incentives or market forces, including those

described in the Remote Web Server References. *See* Section VI.A.4 [The Claimed Concepts Were Known, Remote Web Server / Remote Web Server].

Additionally, a POSITA would have had a reasonable expectation of success in adding this feature to Squirrel, as Squirrel already discloses a browser style desktop that is able to be accessed remotely via the Internet. In other words, modifying Squirrel to use a remote web server would have merely involved the use of a known technique to improve similar systems in the same way.

ii. Browser

The '793 patent claims recite access to point of sale screens with browser-based web access. *See* '793 patent, claim 11 ("The web-based point of sale (POS) builder system of claim 1, wherein the POS builder interface is accessible via a web browser.").

This feature is taught by and/or would have been obvious in view of Squirrel. For example, Squirrel relates generally to a "Restaurant Point of Sale and Management system." Squirrelsystems, *SquirrelOne Solutions*, https://web.archive.org/web/20051201101248/http://www.squirrelsystems.com:80/solutions/index.html (Dec. 1, 2005). Squirrel further discloses that its POS Manager was "[b]uilt on the latest Microsoft SQL Server relational database" "with a familiar browser style desktop [that] is able to be accessed directly from the central PC or remotely via the internet" Squirrelsystems, *Corporate POS*, https://web.archive.org/web/20060327174026/http://www.squirrelsystems.com/solutions/corppos.html (Mar. 27, 2006). It thus would have been obvious to use a browser to run the POS Manager.

To the extent CloudofChange contends that Squirrel fails to disclose or render obvious this feature, a POSITA would have been motivated to modify Squirrel's system to use a browser,

as disclosed in the Browser References. *See* Section VI.A.6 [The Claimed Concepts Were Known, Access to Point of Sale Screens / Browser].

Using Squirrel's system with a browser would have realized many benefits. For example, using a browser would allow a business to use standard and commoditized software applications, instead of special software or hardware. In addition, browsers enable the use of "thin clients" that do not need to run applications (other than the browser) locally because the processing is largely done at the remote server. Such thin clients would have reduced requirements for memory and processing power, as compared to clients that need to run the applications locally. Browser access to point-of-sale screens would also allow customers to place orders from their own phones, or remotely. *See* Section VI.A.6 [The Claimed Concepts Were Known, Access to Point of Sale Screens / Browser]. In addition, because browsers were well-known long before the Asserted Patents, using a browser would have allowed Squirrel's system to use standardized hardware and software, or training by using HTML standard protocol, and lower the cost of POS terminals.

Moreover, the ability to use a browser, and the benefits of doing so, would have been predictable to a POSITA because it would have been a combination of prior art elements according to known methods to yield predictable results, because a POSITA would have known of the general availability of browsers and would have understood that it would have been beneficial and desirable to use browsers to a networked computer system, such as a POS system, like the one disclosed in Squirrel.

Modifying Squirrel to use a browser rather than a non-browser program would have been a simple substitution of one known element (a non-browser program) for another (a browser) to obtain predictable results (the benefits of using a browser).

Moreover, modifying Squirrel to use a browser would have been obvious to try. When implementing a server, an administrator only has limited options—using a non-browser program or a browser. Each solution has its known advantages and disadvantages, and a POSITA would have understood the benefits and advantages of a browser and been motivated to implement a remote web server in order to achieve those benefits and advantages.

Furthermore, using a browser was a known approach for running hardware and software in POS systems, so a POSITA would have been motivated to implement a browser, as taught in the Browser References in the same field, based on design incentives or market forces, including those described in the Browser References. *See* Section VI.A.6 [The Claimed Concepts Were Known, Access to Point of Sale Screens / Browser].

Additionally, a POSITA would have had a reasonable expectation of success in adding this feature to Squirrel, as Squirrel already discloses a browser style desktop that is able to be accessed remotely via the Internet. In other words, modifying Squirrel to use a browser would have merely involved the use of a known technique to improve similar systems in the same way.

VII. SECONDARY CONSIDERATIONS

When performing an obviousness analysis, objective evidence of obvious or non-obviousness may be considered, if present. As the U.S. Supreme Court and Federal Circuit have both held, evidence of near simultaneous invention by others may support a finding of obviousness. *See, e.g., Concrete Appliances Co. v. Gomery*, 269 U.S. 177, 184-85 (1925) (commenting that claimed combination "independently made" by multiple engineers "within a comparatively short space of time" was "persuasive evidence" that the combination was "the product only of ordinary mechanical or engineering skill and not of inventive genius"); *George M. Martin Co. v. Alliance Mach. Sys. Int'l LLC*, 618 F.3d 1294, 1305 (Fed. Cir. 2010) ("Independently made, simultaneous inventions, made 'within a comparatively short space of

time,' are persuasive evidence that the claimed apparatus 'was the product only of ordinary mechanical or engineering skill'"). In this case, such evidence of near-simultaneous invention exists. Clover's identified prior art confirms near simultaneous invention by multiple individuals of a web-based point-of-sale builder with the components and operations recited in the Asserted Claims.

Objective evidence of non-obviousness may also be considered. *See KSR Int'l Co. v. Teleflex Inc.*, 550 U.S. 398, 399 (2007) (citing *Graham v. John Deere Co.*, 383 U.S. 1, 17-18 (1966)). But such evidence, even if present, cannot overcome a strong case of obviousness, like that presented here. *See Richardson-Vicks, Inc. v. Upjohn Co.*, 122 F.3d 1476, 1484 (Fed. Cir. 1997) (recognizing that even if commercial success is demonstrated, this is not necessarily dispositive of the issue of obviousness); *Newell Cos. v. Kenney Mfg. Co.*, 864 F.2d 757, 769 (Fed. Cir. 1988) ("In this case, as well, although the record shows a highly successful product, the record also establishes such a strong case of obviousness based on admissions and the teachings of the prior art, including [a reference that was not before the examiner], that the objective evidence of nonobviousness does not persuade us to reach a contrary conclusion."). CloudofChange has the burden to produce evidence of secondary considerations of nonobviousness and has not yet done so. Clover is unaware of any evidence of secondary considerations that would reflect any nonobviousness of the claimed subject matter of the Asserted Patents. For example, CloudofChange has not established that there is any copying by Clover or others, that any product has enjoyed commercial success due to the merits of the inventions claimed in the Asserted Patents, that the claimed inventions received any praise from others in the field, that there was any skepticism by experts, or any teaching away.

Additionally, given the strength of the prior art and the strong motivation and teachings of the art discussed above, secondary considerations will not render the patent nonobvious. *See Richardson-Vicks*, 122 F.3d 1476, 1483-84. Clover reserves the right to rebut any claims of secondary considerations of non-obviousness raised by CloudofChange.

VIII. ON-SALE BAR

35 U.S.C. § 102 (b) states that a person shall be entitled to a patent unless "the invention was patented or described in a printed publication in this or a foreign country or in public use or on sale in this country, more than one year prior to the date of the application for patent in the United States."

In IPR proceedings involving the Asserted Patents, the inventors Mr. Baratta and Mr. Olson filed declarations stating that in 2006 they began collaborating on potential projects related to point of sale (POS) systems, and on Jan. 23, 2007, formed the entity TYRAMM International Inc. in connection with their efforts. *See, e.g.*, IPR2022-00779, Ex. 2005, Ex. 2006.

Clover intends to investigate whether Mr. Baratta and Mr. Olson, including but not limited to with TYRAMM or some other entity, had an embodiment of the claims of the Asserted Patents that was in public use, on sale, or offered for sale prior to Feb. 5, 2008, as set forth in 35 U.S.C. § 102(b). Clover reserves the right to assert this defense as discovery progresses.

IX. CONTENTIONS UNDER 35 U.S.C. § 112

In addition to the reasons described above, the Asserted Claims are invalid under 35 U.S.C. § 112. Clover includes below the grounds on which Clover contends the Asserted Claims of the Asserted Patents are invalid for failure to meet the requirements of 35 U.S.C. § 112.

To the extent the following contentions reflect constructions of claim limitations consistent with or implicit in CloudofChange's Preliminary Infringement Contentions, no

inference is intended nor should any be drawn that Clover agrees with CloudofChange's claim constructions, and Clover expressly reserves the right to contest such claim constructions. Clover offers these based on the positions taken in CloudofChange's Preliminary Infringement Contentions. In several places, these contentions assume CloudofChange's positions including on claim construction even where those positions conflict with the intrinsic and extrinsic record. Clover addresses the effects of the flawed CloudofChange position for the purpose of completeness and not because Clover believes that CloudofChange's contentions are correct. Moreover, to the extent that the Court construes the claims differently than the implicit constructions in CloudofChange's contentions or CloudofChange changes the nature of its assertions in this case, these assertions may no longer apply. Accordingly, Clover reserves the right to supplement, amend, and/or modify these § 112 invalidity contentions as discovery progresses.

A. Legal Standard

1. Written Description and Enablement

35 U.S.C. § 112 ¶ 1 provides, in relevant part:

> The specification shall contain a written description of the invention, and of the manner and process of making and using it, in such full, clear, concise, and exact terms as to enable any person skilled in the art to which it pertains, or with which it is most nearly connected, to make and use the same … .

35 U.S.C. § 112 ¶ 1. Thus, § 112 ¶ 1 prescribes two separate requirements for the specification of a valid patent: the written description requirement, and the enablement requirement. *Ariad Pharms., Inc. v. Eli Lilly & Co.*, 598 F.3d 1336, 1344 (Fed. Cir. 2010) (en banc).

"The 'written description' requirement implements the principle that a patent must describe the technology that is sought to be patented; the requirement serves both to satisfy the inventor's obligation to disclose the technologic knowledge upon which the patent is based, and

to demonstrate that the patentee was in possession of the invention that is claimed." *Capon v. Eshhar*, 418 F.3d 1349, 1357 (Fed. Cir. 2005). To satisfy this requirement, a "description must clearly allow persons of ordinary skill in the art to recognize that [the inventor] invented what is claimed." *Ariad Pharms.*, 598 F.3d at 1351. "In other words, the test for sufficiency is whether the disclosure of the application relied upon reasonably conveys to those skilled in the art that the inventor had possession of the claimed subject matter as of the filing date." *Id.* This test "requires an objective inquiry into the four corners of the specification from the perspective of a POSITA . . . [to] show that the inventor actually invented the invention claimed." *Rivera v. Int'l Trade Comm'n*, 857 F.3d 1315, 1319 (Fed. Cir. 2017) (internal quotation omitted). "The knowledge of ordinary artisans may be used to inform what is actually in the specification, . . . but not to teach limitations that are not in the specification, even if those limitations would be rendered obvious by the disclosure in the specification." *Id.* at 1322.

"To obtain the benefit of the filing date of a parent application, the claims of the later-filed application must be supported by the written description in the parent 'in sufficient detail that one skilled in the art can clearly conclude that the inventor invented the claimed invention as of the filing date sought.'" *Anascape, Ltd. v. Nintendo of Am. Inc.*, 601 F.3d 1333, 1335 (Fed. Cir. 2010) (quoting *Lockwood v. American Airlines, Inc.*, 107 F.3d 1565, 1572 (Fed. Cir. 1997)). "[A] broad claim is invalid when the entirety of the specification clearly indicates that the invention is of a much narrower scope." *Carnegie Mellon Univ. v. Hoffmann-La Roche Inc.*, 541 F.3d 1115, 1127 (Fed. Cir. 2008) ((quoting *Cooper Cameron Corp. v. Kvaerner Oilfield Products, Inc.*, 291 F.3d 1317, 1323 (Fed. Cir. 2002))); *see also Gentry Gallery, Inc. v. Berkline Corp.*, 134 F.3d 1473, 1480 (Fed. Cir. 1998) ("[C]laims may be no broader than the supporting disclosure, and therefore that a narrow disclosure will limit claim breadth.")

"To meet the enablement requirement, the specification of a patent must teach those skilled in the art how to make and use the full scope of the claimed invention without undue experimentation." *Martek Biosciences Corp. v. Nutrinova Inc.*, 579 F.3d 1363, 1378 (Fed. Cir. 2009) (internal quotation omitted). Meeting this standard requires the specification "to teach those of ordinary skill how to make and how to use the invention as broadly as it is claimed." *In re Vaeck*, 947 F.2d 488, 496 (Fed. Cir. 1991). A court may consider the following factors when determining if a disclosure requires undue experimentation: "(1) the quantity of experimentation necessary, (2) the amount of direction or guidance presented, (3) the presence or absence of working examples, (4) the nature of the invention, (5) the state of the prior art, (6) the relative skill of those in the art, (7) the predictability or unpredictability of the art, and (8) the breadth of the claims." *ALZA Corp. v. Andrx Pharms., LLC*, 603 F.3d 935, 940 (Fed. Cir. 2010) (quoting *In re Wands*, 858 F.2d 731, 737 (Fed. Cir. 1988)).

2. Indefiniteness

35 U.S.C. § 112 ¶ 2 requires that a specification "conclude with one or more claims particularly pointing out and distinctly claiming the subject matter which the applicant regards as his invention." 35 U.S.C. § 112 ¶ 2. Under § 112 ¶ 2, "a patent is invalid for indefiniteness if its claims, read in light of the specification delineating the patent, and the prosecution history, fail to inform, with reasonable certainty, those skilled in the art about the scope of the invention." *Nautilus, Inc. v. Biosig Instruments, Inc.*, 572 U.S. 898-99, 901 (2014); *see also id*. at 899 ("[A] patent must be precise enough to afford clear notice of what is claimed, thereby apprising the public of what is still open to them." (internal quotation omitted)).

Section 112 ¶ 6 allows a patentee to express an element of a claim "as a means or step for performing a specified function . . . and such claim shall be construed to cover the corresponding structure . . . described in the specification and equivalents thereof." 35 U.S.C. § 112 ¶ 6. The

statute thus establishes a *quid pro quo* whereby a patentee may claim an element using generic "means" language for performing a function (rather than describing in the words of the claim itself the structure used), but if it does so, the patent's specification must disclose structure capable of performing that function, and the scope of the claims is limited by that disclosure. *Biomedino, LLC v. Waters Techs. Corp.*, 490 F.3d 946, 948 (Fed. Cir. 2007). If there are functions claimed pursuant to § 112 ¶ 6 that are performed by a computer or microprocessor, then there must be, for each function, a disclosed software algorithm that the computer performs to accomplish the function. *See Ergo Licensing LLC v. CareFusion 303, Inc.*, 673 F.3d 1361, 1364 (Fed. Cir. 2012); *Noah Sys., Inc. v. Intuit Inc.*, 675 F.3d 1302, 1318-19 (Fed. Cir. 2012); *Aristocrat Techs. Austr. Pty Ltd. v. Int'l Gaming Tech.*, 521 F.3d 1328, 1333 (Fed. Cir. 2008). "Simply reciting 'software' without providing some detail about the means to accomplish the function is not enough." *Finisar Corp. v. DirecTV Grp., Inc.*, 523 F.3d 1323, 1340-41 (Fed. Cir. 2008); *see also Blackboard, Inc. v. Desire2Learn, Inc.*, 574 F.3d 1371, 1384 (Fed. Cir. 2009). If the patentee does not disclose structure for a means-plus-function term, it is indefinite. *Ergo*, 673 F.3d at 1363.

B. The Asserted Claims

1. The Asserted Claims of the '640 Patent Lack Written Description and/or Enablement under 35 U.S.C. § 112, ¶ 1

The Asserted Claims of the '640 Patent fail to comply with 35 U.S.C. § 112, ¶ 1 for at least the following reasons.

a. Claim 1

i. "the subscriber's desired terminal function"

The term "terminal function" does not appear in the specification of the '640 patent. The specification uses the term "function" only once, stating "[t]here is a 'loose coupling' of the POS

to the back office (BO): the POS to BO connection is not required for the basic business functions of the POS." '640 patent, 3:47-49. This portion of the specification does not, however, mention "terminal function" at all. Moreover, this is a generic term without any inherent meaning to a POSITA. As such, this term lacks written description.

ii. "subscriber company's POS terminals access and repeatedly interact with said web server software from said vendor's remote servers, in order to perform the subscriber's desired terminal function"

The term "subscriber's desired terminal function" term lacks written description support, at least for similar reasons as discussed above.

In addition, the term "interact" does not appear in the specification of the '640 patent, much less "access and repeatedly interact." Nor does the term "remote server" appear. The word "remote" only appears six times, but none of those times are in connection with a "remote server." *See* '640 patent, 1:40 ("remote store location"), 2:12-13 ("local or remote PC workstations"), 2:14 ("[l]ocal or remote workstations"), 2:56-57 ("the programmer is located remotely from the store or business"), 3:6 ("a remotely located person"), 5:54-55 ("remote monitoring of screen designs").

Thus, the limitation "subscriber company's POS terminals access and repeatedly interact with said web server software from said vendor's remote servers, in order to perform the subscriber's desired terminal function" lacks written description.

b. Claim 3

i. "use . . . requires no special training"

The term "special training" does not appear in the specification of the '640 patent. The patent mentions "training" only once, but does not indicate anything about "special training," much less that "no special training" is required for the "builder." *See* '640 patent, 5:50-51

("There is less training required for programmers and system testers."). The term "special" appears only twice, and both times to describe the prior art. *See* '640 patent, 1:32-33 ("Currently, only specially trained people can build or change POS screens."), 2:54-56 ("Again in the prior art, a specialized programmer had to design the layout and data for these POS touch keys."). The patent's mention that "[c]urrently, only specially trained people can build or change POS screens" does not describe that the system "requires no special training," much less show that the inventors at the time of filing had possession of an invention that would "require no special training." This term lacks written description.

Nor does the '640 patent enable this term. At most, the '640 patent states that "[i]t can be used by a person of any skill level." '640 patent, 5:25-26. The '640 patent does not, however, enable a POSITA to build a system that "requires no special training." Special training is a subjective term, as what might be "special" training for one individual might be typical training for another, or require no training at all for a third. Moreover, the patent does not disclose how to build such a system or what a POSITA should or would do in order to achieve a system that "requires no special training." This term is not enabled.

c. Claim 5

i. "no specialized hardware or software is required to implement said POS builder"

The terms "specialized hardware" or "specialized software" or "specialized hardware or software" do not appear in the specification of the '640 patent. The term "special" appears only twice, and both times to describe the prior art, and specifically, trained people or programmers. *See* '640 patent, 1:32-33 ("Currently, only specially trained people can build or change POS screens."), 2:54-56 ("Again in the prior art, a specialized programmer had to design the layout and data for these POS touch keys.").

The patent mentions "hardware" only three times: once when describing the prior art, and twice when describing the POS terminals. *See* '640 patent, 1:51-53 ("(Cohen et al.) discloses a configuration builder useful in configuring software containing hardware units"), 3:40-41 ("Each POS includes personal computer hardware and software."), 3:43-44 ("Each POS normally operates with a hardware/software connection 35 to the Internet or Web."). At most, the patent states "the use of standard PC and web architecture" ('640 patent, 5:43-45) but "architecture" is not the same as "hardware or software." Nor does the single mention of "standard PC and web architecture" explain what is "standard," much less whether any particular "hardware or software" is "specialized." This term lacks written description.

Nor does the '640 patent enable this term. The patent does not enable a POSITA to build a system that requires "no specialized hardware or software . . . to implement said POS builder." The specification does not enable, for example, a POSITA to implement "the POS builder" with non-specialized hardware and software, since, by definition, programming non-specialized hardware to perform particular functions (e.g., the claimed functions of the POS builder) results in specialized hardware and/or software. *See WMS Gaming, Inc. v. Int'l Game Tech.*, 184 F.3d 1339, 1348 (Fed. Cir. 1999) ("A general purpose computer, or microprocessor, programmed to carry out an algorithm creates 'a new machine, because a general purpose computer in effect becomes a special purpose computer once it is programmed to perform particular functions pursuant to instructions from program software.'"); *see also id.* ("The instructions of the software program that carry out the algorithm electrically change the general purpose computer by creating electrical paths within the device. These electrical paths create a special purpose machine for carrying out the particular algorithm."). This term is not enabled.

d. Claim 6

i. "said POS builder is useable to deliver POS application solutions using software as a subscription (SAAS)."

The term "POS application solutions" does not appear in the specification of the '640 patent. Nor does the term "solution" appear in the specification of the '640 patent. Nor does the specification mention "software as a subscription" or "SAAS." At most, the specification mentions "Software as a Service (SMS)" ('640 patent, 5:60-63) but that single sentence in the specification does not disclose "using software as a subscription," much less "deliver[ing] POS application solutions" or doing so using the "POS builder." This term lacks written description.

Nor does the '640 patent enable this term. As mentioned above, the patent does not mention these terms, so accordingly does not describe or otherwise provide any direction or guidance for a POSITA to perform this claimed function. "POS application solutions" is not a term of art, nor does it have plain and ordinary meaning, so without any direction or guidance from the specification or claims, a POSITA would not know how to practice this term.

Moreover, at least because the patent does not enable "POS application solutions," the patent also does not enable "POS application solutions using software as a subscription (SAAS)." Similarly, the patent does not enable using the POS builder "to deliver POS application solutions using software as a subscription (SAAS)."

2. The Asserted Claims of the '012 Patent Lack Written Description and/or Enablement under 35 U.S.C. § 112, ¶ 1

The Asserted Claims of the '012 Patent fail to comply with 35 U.S.C. § 112, ¶ 1 for at least the following reasons.

a. Claim 1

i. "employee clock information, customer add/update information, item add/update information, promotion information, loyalty point information, discount

information, taxation information, item cost information, or inventory information"

The specification mentions "employee clock information, customer add/update information, item add/update information, promotion information, loyalty point information, discount information, taxation information, item cost information, or inventory information" each only once or a few instances. *See* '012 patent, 4:40-47 ("The packet 35 sent from the POS to the BO consists of transactions, **employee clock, customer add/update, item add/update**, **promotions** and more. **Promotions** are configured in the back office and associated with items or customers or departments. For example, a **promotion** may be associated with a customer to implement customer **loyalty points** or a **promotion** may be associated with a certain item for a % **discount**." (emphasis added)), 4:66-5:1 ("The store operator would be able to enter and/or modify item name, price, **cost**, group, **taxable** and **inventory**." (emphasis added)). But nowhere does the specification describe or enable how any of the claimed information might be part of "the further information regarding the one or more POS transactions, the information used for creating or modifying the one or more POS screens, or a combination thereof," as recited in claim 1. As such, this claim term lacks written description and is not enabled.

3. The Asserted Claims of the '793 Patent Lack Written Description and/or Enablement under 35 U.S.C. § 112, ¶ 1

The Asserted Claims of the '793 Patent fail to comply with 35 U.S.C. § 112, ¶ 1 for at least the following reasons.

a. Claim 1

i. "loyalty points information"

Claim 1 recites "loyalty points information." This term lacks written description. The term does not appear in the specification at all. The word "loyalty" appears in the specification only once. *See* '793 patent, 4:48-51 ("For example, a promotion may be associated with a

customer to implement customer loyalty points or a promotion may be associated with a certain item for a % discount."). But the single mention in the specification of "customer loyalty points" does not describe "loyalty points information," much less how the claimed "loyalty points information" is "the further information regarding the one or more POS transactions, the information used for creating or modifying the one or more POS screens, or a combination thereof."

b. Claim 4

i. "loyalty points programs"

Claim 4 recites a "loyalty points program." This term lacks written description. The term does not appear in the specification at all. The word "loyalty" appears in the specification only once. *See* '793 patent, 4:48-51 ("For example, a promotion may be associated with a customer to implement customer loyalty points or a promotion may be associated with a certain item for a % discount."). But the single mention in the specification of "customer loyalty points" does not describe a "loyalty points program," much less how the claimed "loyalty points program" is "the one or more items" associated with "the one or more display interfaces for display on the one or more POS screens."

4. The Asserted Claims of the '640 Patent are Indefinite under 35 U.S.C. § 112, ¶ 2

The Asserted Claims of the '640 Patent fail to comply with 35 U.S.C. § 112, ¶ 2 for at least the following reasons.

a. Claim 1

i. "said web servers"

Claim 1 refers to "said web servers," but never introduces "web servers." Thus, the term lacks antecedent basis and it is unclear what "said web servers" claim 1 refers to, so this claim term is indefinite.

ii. "the subscriber's desired terminal function"

It is unclear what "terminal function" means and, as such, this term is indefinite because, "read in light of the specification delineating the patent, and the prosecution history, [it] fail[s] to inform with reasonable certainty, those skilled in the art about the scope of the invention." *Nautilus, Inc. v. Biosig Instruments, Inc.*, 572 U.S. 898, 901 (2014).

As discussed above, the term "terminal function" does not appear in the specification and lacks written description, so the patent specification does not provide any guidance to a POSITA attempting to understand the scope of the invention.

This term is indefinite.

iii. "subscriber company's POS terminals access and repeatedly interact with said web server software from said vendor's remote servers, in order to perform the subscriber's desired terminal function"

This term is indefinite for reciting a method-of-using limitation. *See IPXL Holdings, L.L.C. v. Amazon.com, Inc.*, 430 F.3d 1377, 1384 (Fed. Cir. 2005) ("reciting both an apparatus and a method of using that apparatus renders a claim indefinite under section 112, paragraph 2"). Claim 1 recites "[a] web-based point of sale (POS) builder system," which is a system claim. The claim terms "access and repeatedly interact . . ." and "to perform the subscriber's desired terminal function," however, are method steps (e.g., access, interact, perform). "[A]s a result of the combination of two separate statutory classes of invention, a manufacturer or seller of the

claimed apparatus would not know from the claim whether it might also be liable for contributory infringement because a buyer or user of the apparatus later performs the claimed method of using the apparatus." *Id.* This term is indefinite.

b. Claim 3

i. "use . . . requires no special training"

It is unclear what "no special training" means and, as such, this term is indefinite because, "read in light of the specification delineating the patent, and the prosecution history, [it] fail[s] to inform with reasonable certainty, those skilled in the art about the scope of the invention." *Nautilus, Inc. v. Biosig Instruments, Inc.*, 572 U.S. 898, 901 (2014).

As discussed above, the term "no special training" does not appear in the specification and lacks written description, so the patent specification does not provide any guidance to a POSITA attempting to understand the scope of the invention.

Moreover, it is unclear what level of training would be sufficient for a user to use the system while not being "special training," as recited. Neither the claim nor specification give any guidance as to what constitutes "special training," so a POSITA would not know what type or level of training would or would not fall within the bounds of the claim.

This term is indefinite.

ii. "said builder"

Claim 3 refers to "said builder," but neither claim 3 nor claim 1, from which claim 3 depends, introduces "a builder." Thus, the term lacks antecedent basis and it is unclear what "said" builder claim 3 refers to, so this claim term is indefinite.

c. Claim 5

i. "no specialized hardware or software is required to implement said POS builder"

It is unclear what "specialized hardware or software" means and, as such, this term is indefinite because, "read in light of the specification delineating the patent, and the prosecution history, [it] fail[s] to inform with reasonable certainty, those skilled in the art about the scope of the invention." *Nautilus, Inc. v. Biosig Instruments, Inc.*, 572 U.S. 898, 901 (2014).

As discussed above, the terms "specialized hardware" or "specialized software" or "specialized hardware or software" do not appear in the specification and lack written description, so the patent specification does not provide any guidance to a POSITA attempting to understand the scope of the invention.

Moreover, it is unclear what specialization of hardware and/or software would be sufficient to achieve the "POS builder" while still not being "specialized hardware or software," since, by definition, programming non-specialized hardware to perform particular functions (e.g., the claimed functions of the POS builder) results in specialized hardware and/or software. *See WMS Gaming, Inc. v. Int'l Game Tech.*, 184 F.3d 1339, 1348 (Fed. Cir. 1999) ("A general purpose computer, or microprocessor, programmed to carry out an algorithm creates 'a new machine, because a general purpose computer in effect becomes a special purpose computer once it is programmed to perform particular functions pursuant to instructions from program software.'"); *see also id.* ("The instructions of the software program that carry out the algorithm electrically change the general purpose computer by creating electrical paths within the device. These electrical paths create a special purpose machine for carrying out the particular algorithm.").

This term is indefinite.

ii. "said POS builder"

Claim 5 refers to "said POS builder," but neither claim 5 nor claim 1, from which claim 5 depends, introduces "a POS builder." Thus, the term lacks antecedent basis and it is unclear what "said" POS builder claim 5 refers to, so this claim term is indefinite.

d. Claim 6

i. "said POS builder"

This term is indefinite for similar reasons as discussed in connection with claim 5, including that the term lacks antecedent basis.

ii. "POS application solutions"

It is unclear what "POS application solutions" means and, as such, this term is indefinite because, "read in light of the specification delineating the patent, and the prosecution history, [it] fail[s] to inform with reasonable certainty, those skilled in the art about the scope of the invention." *Nautilus, Inc. v. Biosig Instruments, Inc.*, 572 U.S. 898, 901 (2014).

As discussed above, this term does not appear in the specification and lacks written description, so the patent specification does not provide any guidance to a POSITA attempting to understand the scope of the invention.

This term is indefinite.

e. Claim 8

i. "said POS builder"

This term is indefinite for similar reasons as discussed in connection with claim 5, including that the term lacks antecedent basis.

f. Claim 9

i. "said POS builder"

This term is indefinite for similar reasons as discussed in connection with claim 5, including that the term lacks antecedent basis.

g. Claim 10

i. "said POS builder"

This term is indefinite for similar reasons as discussed in connection with claim 5, including that the term lacks antecedent basis.

h. Claim 11

i. "said POS builder"

This term is indefinite for similar reasons as discussed in connection with claim 5, including that the term lacks antecedent basis.

ii. "said keys or buttons"

Claim 11 refers to "said keys or buttons," but neither claim 11 nor claim 1, from which claim 11 depends, introduces "keys or buttons." Thus, the term lacks antecedent basis and it is unclear what "keys or buttons" claim 11 refers to, so this claim term is indefinite.

i. Claim 12

i. "said POS builder"

This term is indefinite for similar reasons as discussed in connection with claim 5, including that the term lacks antecedent basis.

5. The Asserted Claims of the ’012 Patent are Indefinite under 35 U.S.C. § 112, ¶ 2

The Asserted Claims of the ’012 Patent fail to comply with 35 U.S.C. § 112, ¶ 2 for at least the following reasons.

a. Claim 1

i. “POS terminals generated by said POS builder software”

To the extent “POS terminals” are understood to be or include hardware, it is unclear how hardware is “generated by” software.

ii. “one or more terminal devices”

To the extent “POS terminals” are understood to be software, and the “one or more terminal devices” are understood to be hardware, it is unclear why a POS “terminal” is software but a “terminal” device is hardware. *See PODS, Inc. v. Porta Stor, Inc.*, 484 F.3d 1359, 1366 (Fed. Cir. 2007) (“We apply a ‘presumption that the same terms appearing in different portions of the claims should be given the same meaning unless it is clear from the specification and prosecution history that the terms have different meanings at different portions of the claims.’”).

iii. “corresponding transaction data”

Claim 1 of the ’012 patent introduces “corresponding transaction data” three times. *See* ’012 patent, claim 1 (“collect corresponding transaction data”; “said POS transactions and corresponding transaction data”; “correlated with corresponding transaction data”). It is unclear whether the second and third instances introducing “corresponding transaction data” are intended to refer back to the first introduction of “corresponding transaction data,” or to introduce new “corresponding transaction data.” As such, this claim term is indefinite.

b. Claim 2

i. "instruction . . . for programmatic creation and modification . . . are not formatted in programming code"

This claim term is indefinite because it "fail[s] to inform with reasonable certainty, those skilled in the art about the scope of the invention." *Nautilus, Inc. v. Biosig Instruments, Inc.*, 572 U.S. 898, 901 (2014).

The specification mentions "code" only once ('012 patent, 1:35-37), and does not mention "programming code" at all, so the term "programming code" is undefined in the specification. Even if it is argued that "programming code" means "programming instructions" (or instructions "for programmatic creation and modification"), it is unclear how "instruction[s] . . . for programmatic creation" (or instructions for programming) are not formatted as instructions for programming. In other words, the claim appears to require that instructions for programming are not "formatted as" instructions for programming, which is nonsensical.

Moreover, the claim does not inform with reasonable certainty the scope of the invention because "programming code" and "formatted in" are undefined. For example, even if "programming code" is understood to mean, e.g., source code, source code comes in many languages and forms, including, e.g., assembly, bits, 1s and 0s, or high/low electrical signals. Relatedly, all input to a computer is eventually translated to an instruction for processing by a processor, e.g., in the form of bits, 1s and 0s, or high/low electrical signals, so it is unclear at what point, per the claim, "instructions to the POS builder interface" must "not [be] formatted in programming code." If the "instructions to the POS builder interface" must *never* be "formatted in programming code," then it is unclear when this claim limitation could ever be met, because, as discussed, all input to a computer is eventually translated to instructions (code) for programming.

c. Claim 4

i. "a given terminal device"

Claim 4 refers to "a given terminal device." Claim 1, from which claim 4 depends, introduces "one or more POS terminals" and "one or more terminal devices." It is unclear, however, whether claim 4's recitation refers to "a given terminal device" of "the one or more POS terminals," of the "one or more terminal devices," or of neither.

As such, this term is indefinite because, "read in light of the specification delineating the patent, and the prosecution history, [it] fail[s] to inform with reasonable certainty, those skilled in the art about the scope of the invention." *Nautilus, Inc. v. Biosig Instruments, Inc.*, 572 U.S. 898, 901 (2014).

ii. "implemented without specialized hardware or software"

This claim term is indefinite at least for similar reasons as discussed in connection with '640 patent, claim 5.

6. The Asserted Claims of the '793 Patent are Indefinite under 35 U.S.C. § 112, ¶ 2

The Asserted Claims of the '793 Patent fail to comply with 35 U.S.C. § 112, ¶ 2 for at least the following reasons.

a. Claims 1, 42, 44

i. "at least one of said/the one or more POS terminals"

Claims 1, 42, and 44 each recite "communicat[e/ing] with one or more POS terminals." These claims also recite "receiv[e/ing], from at least one of the one or more POS terminals." These claims also recite "corresponding customers respectively associated with at least one of said one or more POS terminals." In other words, each of independent claims 42 and 44 introduce "at least one of [the/said] one or more POS terminals" twice. As such, it is unclear

whether "associated with at least one of said one or more POS terminals" refers to the same "at least one of the one or more POS terminals" as the "further information" was "receive[d]" from, or a different "at least one of the one or more POS terminals" than the "further information" was "receive[d] from." This term is indefinite.

b. Claim 3

i. "a plurality of POS terminals in a plurality of locations"

To the extent the claimed "plurality of POS terminals" refers to a virtual terminal (e.g., software terminal) and not a hardware terminal, this term is indefinite. It is unclear how a virtual terminal can be "in a plurality of locations" (e.g., a plurality of terminals each individually executing across different servers, collectively executing across different servers, a plurality of terminals each individually executing across different workstations, collectively executing across different workstations, etc.).

Moreover, this term is indefinite because it is unclear whether the term is met by two locations (e.g., a plurality of locations) each having one POS terminal, or if the term requires two locations each having two POS terminals (e.g., "a plurality of POS terminals" at each of "a plurality of locations").

As such, this term is indefinite because, "read in light of the specification delineating the patent, and the prosecution history, [it] fail[s] to inform with reasonable certainty, those skilled in the art about the scope of the invention." *Nautilus, Inc. v. Biosig Instruments, Inc.*, 572 U.S. 898, 901 (2014).

* * *

Additionally, Clover reserves the right to present additional § 112 defenses if CloudofChange proffers a construction for certain terms and phrases that is broader than, or inconsistent with, the court's claim constructions or any construction that would be supportable by the disclosure set forth in the specification.

Accordingly, Clover reserves the right to supplement, amend, and/or modify these § 112 invalidity contentions as discovery progresses.

X. CONTENTIONS UNDER 35 U.S.C. § 101

Clover includes below the grounds on which Clover contends the asserted claims of the Asserted Patents are unpatentable for lack of eligible subject matter under 35 U.S.C. § 101.

In *Alice Corp. Pty. v. CLS Bank Int'l*, the Supreme Court set forth a two-part test to analyze patent eligibility under 35 U.S.C. § 101: (1) whether the claims are directed to an abstract idea or other patent-ineligible concept; and (2) if so, whether the claims add an "inventive concept" sufficient to ensure that the patent "amounts to significantly more" than the abstract idea. 573 U.S. 208, 217-18 (2014). First, the claims must be "considered in their entirety to ascertain whether their character as a whole is directed to excluded subject matter." *Internet Patents Corp. v. Active Network, Inc.*, 790 F.3d 1343, 1346 (Fed. Cir. 2015). For claims directed at computer technology, "the first step in the *Alice* inquiry . . . asks whether the focus of the claims is on the specific asserted improvement in computer capabilities . . . or, instead, a process that qualifies as an 'abstract idea' for which computers are invoked merely as a tool." *Enfish, LLC v. Microsoft Corp.*, 822 F.3d 1327, 1335-36 (Fed. Cir. 2016). This first step of the patent-eligibility inquiry has been "described . . . as looking at the 'focus' of the claims," or "their 'character as a whole.'" *Electric Power Grp., LLC v. Alstom S.A.*, 830 F.3d 1350, 1353 (Fed. Cir. 2016). "[A]nalyzing information … by mathematical algorithms" is a "mental

process[ ] within the abstract-idea category." *Id.* at 554; *accord SAP America, Inc. v. InvestPic, LLC*, 898 F.3d 1161, 1167 (Fed. Cir. 2018). Thus, "process[es] that start[ ] with data, add[ ] an algorithm, and end[ ] with a new form of data" are "directed to an abstract idea." *RecogniCorp, LLC v. Nintendo Co.*, 855 F.3d 1322, 1327 (Fed. Cir. 2017). Indeed, the Federal Circuit has specifically held abstract claims that recited generating "file content identifiers for data files … using a mathematical algorithm" as part of a method for "filtering e-mails that have unwanted content." *Intellectual Ventures I LLC v. Symantec Corp.*, 838 F.3d 1307, 1313 (Fed. Cir. 2016).

Next, if a claim is directed to an abstract idea, the elements of the claims must be examined individually and as an ordered combination to determine whether there are "additional elements" that "transform the nature of the claim into a patent-eligible application." *Alice*, 573 U.S. at 217 (quoting *Mayo Collaborative Servs. v. Prometheus Lab'ys, Inc.*, 566 U.S. 66, 72 (2012)). This step requires a search for an "inventive concept" "sufficient to ensure that the patent in practice amounts to significantly more than a patent upon the [ineligible concept] itself." *Id.* at 217-18. Implementing an abstract idea through conventional computer components performing conventional tasks does not transform the abstract idea into an inventive concept. *Id.* at 222. For computerized components to save a claim, they must involve more than performance of "well-understood, routine, conventional activit[ies] previously known to the industry." *Id.* at 225; *see also In re TLI Commc'ns LLC Patent Litig.*, 823 F.3d 607, 614 (Fed. Cir. 2016) ("generic computer components [are] insufficient to add an inventive concept to an otherwise abstract idea."); *Intellectual Ventures I*, 838 F.3d at 1320. It is not enough to "limit[] the use of an abstract idea to a particular technological environment." *Alice*, 573 U.S. at 223.

All of the asserted claims are unpatentable under 35 U.S.C. § 101 because they fail the two-step test for eligibility set forth in *Alice*.

A. The Asserted Claims Are Abstract Ideas Under *Alice* Step One

At base, the asserted claims are directed to the abstract idea of a tool for building or editing a POS terminal. The idea of building or editing a POS terminal is an "abstract idea" ineligible for patent protection because building a POS system to manage business operations is an age-old practice that existed well before the advent of computers, the Internet, and the Web. For example, long before computers, humans would use pen and paper to record transactions (such as restaurant orders), and use mechanical cash registers to record transactions for bookkeeping and general business management. Then, before the Internet, restaurants used data machines with graphical interfaces, e.g., electronic cash registers, as POS systems to take orders and record transactions. Then, after the development of the Internet, and long before the asserted patents were filed, POS systems that managed business operations over the Internet became commonplace, whereby waitstaff could customize orders using a mobile tablet with a touchscreen. The asserted patents merely describe "Web-based" elements for the basic functions of a POS system and thus do not transform the claimed abstract idea into eligible subject matter.

The Federal Circuit has clearly stated that "add[ing] networking capabilities to existing" systems "to facilitate various business transactions" does not render a claim patent eligible. *ChargePoint, Inc. v. SemaConnect, Inc.*, 920 F.3d 759, 768-770 (Fed. Cir. 2019). Further, the Federal Circuit explained that the "focus of the claim is on the abstract idea of network communication for device interaction." *Id.* at 770. The Federal Circuit held that patents claiming the use of control devices, transceivers, and servers to enable or disable electric vehicle charging stations were directed to "the abstract idea of communication over a network for interacting with a device, applied to the context of electric vehicle charging stations." *Id.* at 768. Like the ineligible claims at issue in *ChargePoint*, the claims here do nothing more than apply

the "abstract idea of communication over a network for interacting with a device" to the context of existing POS terminals. *Id.*

That some asserted claims recite some tangible components does not make them any less abstract, especially where, as here, the technology being used is generic and the claims fail to recite "specific asserted improvements in computer capabilities." *Enfish*, 822 F.3d at 1337. For example, claim limitations directed to "remote servers" does not make the asserted claims any less abstract. The Federal Circuit has repeatedly held that claims that recite nothing more than remotely controlling access to resources, such as stored data, are directed to an abstract idea. *E.g., Intellectual Ventures I LLC v. Erie Indem. Co.*, 850 F.3d 1315, 1330 (Fed. Cir. 2017) ("[r]emotely accessing and retrieving user-specified information is an age-old practice that existed well before the advent of computers and the Internet"); *Ericsson Inc. v. TCL Commc'n Tech. Holdings Ltd.*, 955 F.3d 1317, 1326 (Fed. Cir. 2020) ("That bare abstract idea, controlling access to resources by receiving a request and determining if the request for access should be granted, is at the core of claim 1. Neither of the remaining limitations alter our conclusion that claim 1 is directed to the abstract idea of controlling access to resources.").

The asserted claims also provide no details regarding "how to engineer or program" the system, and thus are "result-centric." *Interval Licensing LLC v. AOL, Inc.*, 896 F.3d 1335, 1345 (Fed. Cir. 2018). Moreover, "the prohibition against patenting abstract ideas 'cannot be circumvented by attempting to limit the use of the formula to a particular technological environment.'" *ChargePoint*, 920 F.3d at 771 (citing *Bilski*, 561 U.S. at 610-11). The abstract ideas of communication with existing systems over the Web are not rendered patent eligible simply by constraining those ideas to use with a particular type of equipment, such as POS terminals. As such, it is clear that the mere concept of web-based tool for building or editing a

POS terminal is itself abstract and cannot without more save the claims from being deemed patent ineligible.

The asserted dependent claims are also directed to tools for building or editing a POS terminal to manage business operations via the Web. Thus, all of the asserted claims are directed to an abstract idea.

B. The Asserted Claims, Alone or In Combination, Do Not Include An Inventive Concept Sufficient to Transform the Claimed Abstract Idea Into a Patent-Eligible Application Under *Alice* Step Two

The asserted claims, whether their elements are considered individually or as an ordered combination, fail to satisfy the second step of the *Alice* test. The asserted claims use known, Web-based concepts, and there is no inventive aspect to how these known components and concepts are applied to the simple, abstract idea of a tool for building or editing a POS terminal to provide any further improvements to these known computer components and concepts. For example, the asserted claims of the ’640 patent describe “an internet connection,” “a web server,” “PC workstations,” and “a network.” The asserted claims of the ’012 patent describe “a web server,” and “a communications network.” The asserted claims of the ’793 patent describe “a server” and “a network.” These are all known, generic computer and network components. The asserted claims do not describe any novel or innovative ways for creating or improving on these well-known conventional components. Instead, the asserted claims merely arrange these generic components to carry out the abstract idea of building or editing a POS terminal to manage business operations via the Web.

The asserted claims do not set forth an inventive concept that would transform their subject matter into something more than the abstract idea. Having servers in a remote location is an old and well-known concept as demonstrated by the references provided in Section VI.A.4. As further evidence from Section VI, the asserted claims of the asserted patents recite only

generic computer and network components ("a server," "a web server," "PC workstations," "a network," and "a communications network"), and thus add nothing inventive to the claims. This is insufficient to transform the claimed abstract idea into patent-eligible subject matter. *See, e.g.*, *Alice*, 573 U.S. at 225-26 (holding that "implement[ing] the abstract idea . . . on a generic computer" was not sufficient "to transform an abstract idea into a patent-eligible invention"); *Intellectual Ventures I*, 850 F.3d at 1341-42 (holding that "using generic computer components and conventional computer data processing activities" was not sufficient to find an "inventive concept"); *TLI Commnc'ns*, 823 F.3d at 614 ("[G]eneric computer components [are] insufficient to add an inventive concept to an otherwise abstract idea."). The dependent claims merely narrow the abstract idea, and thus also add nothing inventive. *See Content Extraction & Transmission LLC v. Wells Fargo Bank, N.A.*, 776 F.3d 1343, 1348-49 (Fed. Cir. 2014) (dependent claims of narrower scope and using conventional techniques are non-inventive).

Thus, the asserted claims are unpatentable under 35 U.S.C. § 101.

/s/ J. Stephen Ravel
J. Stephen Ravel
Texas State Bar No. 16584975
Kelly Ransom
Texas State Bar No. 24109427
KELLY HART & HALLMAN LLP
303 Colorado, Suite 2000
Austin, Texas 78701
Tel: (512) 495-6429
steve.ravel@kellyhart.com
kelly.ransom@kellyhart.com

Caitlyn Hubbard
Texas State Bar No. 24097853
KELLY HART & HALLMAN LLP
201 Main Street, Suite 2500
Fort Worth, Texas 76102
Tel: (817) 332-2500
caitlyn.hubbard@kellyhart.com

Attorneys for Clover Network, LLC

CERTIFICATE OF SERVICE

I hereby certify that all counsel of record are being served with a copy of the foregoing document via email on December 16, 2022.

/s/ J. Stephen Ravel