UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE PATENT TRIAL AND APPEAL BOARD

CLOVER NETWORK, LLC

Petitioner,

v.

CLOUDOFCHANGE, LLC

Patent Owner.

Case IPR2023-00287

U.S. Patent No. 11,226,793

DECLARATION OF DR. TODD MOWRY IN SUPPORT OF PETITION FOR *INTER PARTES* REVIEW

TABLE OF CONTENTS

Page

I. INTRODUCTION 1
II. PROFESSIONAL BACKGROUND 1
III. RELEVANT LEGAL STANDARDS 5
IV. LEVEL OF SKILL IN THE ART 8
V. TECHNOLOGY BACKGROUND 9
VI. THE '793 PATENT 14
A. Overview of the '793 patent 14
B. Prosecution History of the '793 patent 15
VII. LISTING OF CLAIMS 19
VIII. CLAIM CONSTRUCTION 28
A. "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…" (all claims) 28
B. "display interfaces" (all claims) 29
C. "the input interface element" (claim 23) 29
D. "the web server"/"the at least one web server" (claim 38) 30
E. "creating or modifying functionality of the one or POS terminals" (claim 44) 30
IX. OVERVIEW OF THE PRIOR ART 31
A. Overview of Woycik (EX1004) 31
B. Overview of Tengler (EX1005) 32
X. THE PRIOR ART GROUNDS RELIED UPON HEREIN ARE DISTINCT FROM THOSE CONSIDERED DURING EXAMINATION 33

XI. GROUND 1: WOYCIK IN VIEW OF THE KNOWLEDGE OF A PERSON OF ORDINARY SKILL IN THE ART RENDERS CLAIMS 1-4, 7-28, AND 31-44 OBVIOUS 34
A. Claim 1 34
1. 1[pre] 34
2. 1[a] 35
3. 1[c] 38
4. 1[d] 45
5. 1[e] 46
6. 1[f] 54
B. Claims 2 59
C. Claim 3 60
E. Claim 7 63
G. Claim 10 67
H. Claim 12 69
I. Claim 13 71
J. Claim 14 74
K. Claim 15 75
L. Claim 16 76
M. Claim 17 78
N. Claims 18 79
O. Claim 19 80
P. Claim 21 81
Q. Claim 22 82
R. Claim 23 83
S. Claim 24 86
T. Claims 25 87
U. Claim 27 90

1. 27[pre] 91
2. 27[a] 91
3. 27[c] 92
4. 27[d] 94
5. 27[e] 94
6. 27[f] 95
7. 27[g] 96

V. Claims 28, 31-38, 40-41 96
W. Claim 39 98
X. Claim 42 98
Y. Claim 43 99
Z. Claim 44 100

XII. GROUND 2: TENGLER IN VIEW OF THE KNOWLEDGE OF A PERSON OF ORDINARY SKILL IN THE ART RENDERS CLAIMS 1-4, 7-28, AND 31-44 OBVIOUS 101

A. Claim 1 101
1. 1[pre] 101
2. 1[a] 103
3. 1[c] 109
4. 1[d] 115
5. 1[e] 120
6. 1[f] 124

B. Claims 2 132
C. Claim 3 133
D. Claims 4 135
E. Claim 7 137
F. Claim 8 138
G. Claim 10 140

H. Claim 12 143
I. Claim 13 144
J. Claim 14 145
K. Claim 15 146
L. Claim 16 148
M. Claim 17 150
N. Claims 18 151
O. Claim 19 153
P. Claim 21 155
Q. Claim 22 156
R. Claim 23 156
S. Claim 24 159
T. Claims 25 160
U. Claim 27 162
1. 27[pre] 162
3. 27[c] 163
4. 27[d] 165
5. 27[e] 165
6. 27[f] 166
7. 27[g] 166
V. Claims 28, 31-38, 40-41 167
W. Claim 39 169
X. Claim 42 169
Y. Claim 43 170
Z. Claim 44 171
XIII. CONCLUSION 172
XIV. AVAILABILITY FOR CROSS-EXAMINATION 172
XV. RIGHT TO SUPPLEMENT 173

XVI. ACKNOWLEDGMENT 173

I. INTRODUCTION

1. My name is Todd Mowry.

2. I have been retained by Clover Network, LLC ("Clover") to provide opinions in this proceeding related to U.S. Patent No. 11,226,793 ("'793 patent").

II. PROFESSIONAL BACKGROUND

3. I am a Professor in the Department of Computer Science at Carnegie Mellon University. I also have a courtesy appointment in the Department of Electrical and Computer Engineering. I have served on the faculty of Carnegie Mellon University for 25 years starting in 1997 through the present (2022).

4. As a faculty member, I have taught and continue to teach courses and directed research in computer systems and software, operating systems, distributed and network systems, object-oriented programming and design, and mobile computing.

5. I also served on the faculty of the University of Toronto for four years between 1993 and 1997, in the Department of Electrical and Computer Engineering and with a courtesy appointment in the Department of Computer Science.

6. Prior to that appointment, I served as a Graduate Research Assistant in the Department of Electrical Engineering at Stanford University for four years between 1989 and 1993.

7. I received a B.S. degree in Electrical Engineering with Highest Distinction from the University of Virginia in May 1988. I received an M.S. in Electrical Engineering from Stanford University in June 1989 and a Ph.D. in Electrical Engineering from Stanford University in March 1994.

8. I have worked in the computer industry in various capacities. I was a part-time Computer Architect and then Computer Architecture Consultant at Silicon Graphics, Inc., in Mountain View, California (formerly MIPS Computer Systems in Sunnyvale, California) from 1989 to 1993 and 1993 to 1996, respectively.

9. I was a Visiting Scientist at IBM in Toronto from 1996 to 2004. During that same time period (1996 to 2004), I was also a Member of the Technical Advisory Board of SandCraft, Inc. in Santa Clara, California. I was the Director of the Intel Research Pittsburgh Lab at Intel Corporation in Pittsburgh, Pennsylvania from 2004 to 2007.

10. I have authored 19 journal articles and 60 conference papers. I am also an inventor on 5 patents. I am the recipient of several honors and awards: the

Arthur Samuel Thesis Award (awarded by the Stanford Computer Science department to the top two Ph.D. theses in a given year), several IBM Faculty Development Awards (1996, 1997, 1998, 2000, 2001, 2002, and 2003), several Best Paper Awards (the Second Symposium on Operating Systems Design and Implementation in 1996; the 20th International Conference on Data Engineering (ICDE) in 2004), the Alfred P. Sloan Research Fellowship (awarded to researchers in recognition of distinguished performance and a unique potential to make substantial contributions to their field), the Most Thought-Provoking Idea Award in 2004 (awarded by the Architectural Support for Programming Languages and Operating Systems (ASPLOS)), the TR100 Award in 1999 (awarded by MIT's Technology Review magazine to the top 100 most promising young innovators in science and technology), and I became an ACM Fellow in 2016.

11. I am a member of the Institute of Electrical and Electronics Engineers (IEEE) and the Association of Computing Machinery (ACM).

12. I was the Editor-in-Chief of ACM Transactions on Computer Systems, the premier journal for computer systems research, from 2013-2018. I was an Associate Editor for the journal prior to that, since 2001.

13. I was the Program Chair of the International Conference on Architectural Support for Programming Languages and Operating Systems

(ASPLOS) in 2010. I was the Co-Program Chair of the International Conference on Parallel Architectures and Compilation Techniques (PACT) in 2001.

14. I have been on the programming committee in various years for ASPLOS, the International Symposium on Computer Architecture (ISCA), the International Symposium on Microarchitectures, and the Workshop on Architectural and System Support for Improving Format.

15. Overall, I have over 25 years of experience in the field of computer science, including relevant experience with web-based distributed software applications, programming environments, distributed systems, database management systems, and user interfaces.

16. A copy of my *curriculum vitae,* including references to the publications I authored, is attached to my Report as Appendix A.

17. In light of the foregoing, I consider myself to be an expert in the field of computer science, and I believe that I am qualified to provide an opinion as to what a person of ordinary skill in the art would have understood, known, or concluded regarding the subject matter of the ’793 patent at the time of its alleged invention.

18. I am being compensated at my normal consulting rate of $650 an hour for my work. My compensation is not dependent on the outcome of this IPR

proceeding or related litigation, nor is my compensation tied to any positions I take in this Declaration, and therefore it does not affect the substance of my statements in this Declaration.

19. I have no financial interest in Petitioner. I have no financial interest in the '793 patent.

III. RELEVANT LEGAL STANDARDS

20. I am not an attorney. For purposes of this declaration, I have been informed about and understand certain aspects of the law that are relevant to my analysis and opinions.

21. I have been informed and understand that a patent claim can be considered to have been obvious to a person of ordinary skill in the art at the time of the alleged invention. This means that, even if all the requirements of a claim are not found in a single prior art reference, the claim is not patentable if the differences between the subject matter in the prior art and the subject matter in the claim would have been obvious to a person of ordinary skill in the art at the time of the alleged invention.

22. I have been informed and understand that a determination of whether a claim would have been obvious should be based upon several factors, including, among others:

- the level of ordinary skill in the art at the time the application was filed;
- the scope and content of the prior art; and
- what differences, if any, existed between the claimed invention and the prior art.

23. I have been informed and understand that the teachings of two or more references may be combined if such a combination would have been obvious to one having ordinary skill in the art.

24. In determining whether a combination based on multiple references would have been obvious, it is appropriate to consider, among other factors:

- whether the teachings of the prior art references disclose known concepts combined in familiar ways, and when combined, would yield predictable results;
- whether a person of ordinary skill in the art could implement a predictable variation, and would see the benefit of doing so;
- whether the claimed elements represent one of a limited number of known design choices, and would have a reasonable expectation of success by those skilled in the art;

- whether a person of ordinary skill would have recognized a reason to combine known elements in the manner described in the claim;
- whether there is some teaching or suggestion in the prior art to make the modification or combination of elements claimed in the patent; and
- whether the claimed invention applies a known technique that had been used to improve a similar device or method in a similar way.

25. I have been informed and understand that one of ordinary skill in the art has ordinary creativity and is not an automaton.

26. I have been informed and understand that in considering obviousness, it is important not to determine obviousness using the benefit of hindsight derived from the patent being considered.

27. I have been informed and understand that for a patent's claims to receive the benefit of priority of an earlier-filed application, every application in the priority chain leading to the application in question must support every limitation of the claim. This means that every parent application must describe every limitation of every claim and that it is not enough for the parent application to simply render obvious the claims of the child patent. I have been informed and understand that if a child patent includes claims that are greater in scope than the

subject matter that the inventor chose to disclose to the public in the written description of the parent application, the parent application does not provide written description support for the claims of the child patent.

28. I have been informed and understand that whether an application describes a patent's claim limitations is known as the "written description requirement." I have been informed and understand that to satisfy the written description requirement, the patent's specification must describe the claimed invention in sufficient detail such that a person of ordinary skill in the art could reasonably conclude that the inventor had possession of the claimed invention at the time of filing.

IV. LEVEL OF SKILL IN THE ART

29. It is my understanding that when interpreting the claims of the '793 patent, I must do so based on the perspective of one of ordinary skill in the art at the relevant priority date. I understand that the earliest priority date for the '793 patent is February 5, 2008 ("the Critical Date"). For the purpose of this proceeding, I will consider the Critical Date to be the relevant date to ascertain the perspective of one of ordinary skill in the art.

30. In my opinion, a person of ordinary skill in the art at the Critical Date would have been someone with a working knowledge of designing, developing and

deploying web-based software and systems. A person of ordinary skill in the art would have a Bachelor of Science in computer science or a related field, and approximately two years of professional experience or equivalent study in the design and development of web-based software and systems, including web-based POS systems. Additional graduate education could substitute for professional experience, or significant experience in the field could substitute for formal education.

V. TECHNOLOGY BACKGROUND

31. Web-based POS systems were known in the art. For example, Manno discloses a "web-based point of sale (WPOS) system." EX1007, Abstract. Woycik discloses a web-based administrative tool for building and editing user interfaces of self-service POS kiosks. EX1004, Abstract. Olson discloses a web-based back-office system for managing POS terminal devices. EX1006, Abstract. Cusack discloses a "[s]ystem and method for facilitating real-time, web based point of sale operations." EX1009, Abstract. Tomlinson discloses retail store tills coupled to a remote transaction server over the internet. EX1010, [57]. McNally discloses a method that "facilitates database equilibrium and synchronization with wired, wireless and Web-based systems, user-friendly and efficient generation of computerized menus and reservations for restaurants and other applications . . .

information management and communication with host computer, digital input device or remote pager via standard hardwired connection, the internet, a wireless link or the like." EX1021, Abstract. Mueller discloses "a 'Web server,'" "POS server 105," that communicates with terminal devices via the Internet. EX1008, [0036], [0033], [0057]. Tengler discloses "restaurant management software" that allows "a manager to access a management database of a quick-serve restaurant location remotely through a web interface." EX1005, [0049], [0022].

32. POS builder software was also known in the art. For example, Manno discloses the WPOS system allows a manager "to create or alter web pages on the store web sites . . . or to view any other commercial data for either store location in real time." EX1007, [0051]. Woycik discloses "an administrative tool" that provides "a menu editor that enables the administrator to create and edit the interactive menu screens provided by the self-order application at the self-service client terminals." EX1004, [0015], [0016]. Cohen discloses a "configuration builder suitable for use in configuring point of sale systems." EX1020, 1:34-35. McNally discloses an "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." EX1021, 3:16-20. Wagner discloses "[a] builder,

tester, and runtime integration system and method for a graphical touch user interface, such as a point-of-sale (POS) touch screen interface system." EX1022, Abstract. Tengler discloses "user interface designer 614 [that] allows management to edit the user interface of the register and self-service applications and also saves the specifications in the database 602" on main server 464. EX1005, [0103], [0116]. I also understand that Patent Owner stated during a prior litigation that it "does not dispute that the prior art discloses POS builder software." EX1012, 7 n.3.

33. The background of the '793 patent references prior art Eisenbrandt, which discloses "a user configurable interface" that permits users to "configure display screens either at a point of sale location or at home with, e.g., a personal computer." EX1001, 1:54-58 (citing Eisenbrandt). The '793 patent also references prior art Cohen, which discloses "a configuration builder" for configuring POS systems. *Id.*, 1:63-67 (citing Cohen).

34. Web-based access to POS builder systems was also known in the art. For example, Manno discloses "[a]t 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." EX1007, [0051]. Woycik discloses the Internet "can also be used by the administrators to access the administrative tools remotely using a standard web browser." EX1004, [0121]. Mueller discloses "a 'Web server,'" "POS server 105 … that may be accessed via the Web and allows communication with the POS server 105 in a manner known in the art." EX1008, [0036]. Tengler discloses that "[m]anagers access information about the restaurant 2 using the web server 620 and interfaces 192 and 196," and that "managers can … modify menus and change prices using interfaces 192 and 196." EX1005, [0109], [0073]. Likewise, POS builder interfaces allowing programmatic creation and editing of user interfaces, including POS screens and web pages, were also known. For example, Woycik discloses POS builder screens and their operation in Figs. 11-13 and 20-22 and related discussions in the specification. EX1004, Figs. 11-13, 20-22, [0041]-[0043], [0050]-[0052]. Tengler discloses POS builder screens and describes their operation in related discussions in the specification. *E.g.*, EX1005, Fig. 7A, [0039]. Bernardo discloses more generalized builder screens that could be used to build POS screens and describes the same. *E.g.*, EX1026, Figs. 4-29, 4:3-59.

35. Using the Internet for communications between a POS server and POS terminals was known. For example, Woycik discloses that "central server 22" is configured to communicate with kiosks 16, 30, and/or 36 over a network

comprising "Internet 18." EX1004, [0071]. Woycik also discloses the Internet can "be used by the administrators to access the administrative tools remotely using a standard web browser." EX1005, [0121]. Mueller discloses "[c]ommunication between … [POS terminals] 120 and the POS server 105 may be … over the Internet through a Web site maintained by POS server 105 on a remote server." EX1008, [0034]. Manno discloses "[a] network or web-based point of sale (WPOS) system employs a store server and point-of-sale client computers that are connected … over the Internet to the server." EX1004, Abstract. Wetmore discloses "[t]ypically, the system 100 includes … one or more product information servers 104, and one or more point-of-sale terminals 106" that "communicate with each other via … the Internet or some other wide area network." EX1015, [0011]. Thompson discloses POS terminals connecting to a server over the Internet. EX1013, Fig. 10. And Olson discloses the "[t]he POS terminals and web servers communicate via commonly used, standardized internet protocols." EX1006, [0012]. Configuring POS terminals with information from a POS builder interface such that the terminals display a series of screens during a transaction based on information about a current or previous transaction and/or based on customer-specific information (e.g., past orders, customer ID, customer picture, loyalty program information) was also known in the art, as explained in detail below. For

example, Woycik discloses storing previous orders so "that returning customers can quickly order [and further customize] items ordered in the past without having to rebuild the order each time they use a kiosk." EX1004, [0140]. "For this purpose, the system tracks customer orders and stores them on the server." EX1004, [0089]. Tengler discloses "captur[ing] an image of the customer at the kiosk when the order is placed" and displaying "[t]he contents of the order … with the electronic image" of that customer. EX1005, [0019], [0018].

VI. THE '793 PATENT

A. Overview of the '793 patent

36. The '793 patent is titled "Web-Based Point of Sale Builder," and generally relates to POS systems:

> This invention relates to a system and a method for building a point of sale (POS) system to manage business operations. The business operations range from a single branch to a large chain of stores or branches.
>
> More particularly this invention relates to an online, web-based point of sale builder method, which can assist non-expert business operators in assembling a point of sale system to manage their businesses.

EX1001, 1:22-29.

37. The background of the '793 patent describes the "[c]urrent practice in the field of assembling point of sale systems includes manually coding front-of-screen information" with the help of "a business expert" and "a programmer or

data expert." *Id.*, 1:33-39. The manual process required "specially trained people," was "prone to mistakes," and was "time-consuming." *Id.*, 1:47-49.

38. The '793 patent purports to improve the process of creating or modifying POS screens displayed on POS terminals by using a POS builder interface communicating with a networked server over the Internet. *Id*., 2:16-33, 5:43-6:11, Claim 1.

B. Prosecution History of the '793 patent

39. I have reviewed the prosecution history of the '793 patent. The following is a brief summary of events, none of which impact my analysis. But I reserve the right to consider the full prosecution history and cited prior art references should Patent Owner raise a claim construction argument or otherwise rely on them in response to the petition.

40. The application for the '793 patent was filed November 30, 2017. claiming priority to a February 5, 2008 application (and two other applications) and sharing the same specification as those earlier applications. EX1001, (22), (63) (continuation of earlier applications). The '793 patent application included claims 3–42. EX1003, 589-595. The claims generally covered software to run on a server connected to POS terminals wherein the server received information over a network from a POS builder interface, created or modified POS screens based on

that information, and configured the POS terminals over the network with the created/modified POS screens. *See id.*. Applicant also filed a preliminary amendment refocusing the claims on a server (not software to run on the server). *See, e.g.*, *id.* at 516.

41. The examiner originally rejected all claims for "nonstatutory double patenting," explaining that at least one of the claims "is not patentably distinct from the reference claim(s) because the examined application is either anticipated by, or would have been obvious over, the reference claim(s)." *Id.* at 495. In response, the applicant filed a terminal disclaimer and another terminal disclaimer to address a defect in the first. *Id.* at 488–490, 464–474. I have been told a terminal disclaimer in this context is an agreement that the claims will only be enforceable while they are commonly assigned (e.g., commonly owned) with the reference claim(s) during the first-issued patent's term. This prevents separate lawsuits by separate patent owners over very similar patent claims and limits the term of the later-issued patent.

42. The examiner relied on the following named references in subsequent rejections.

- Wagner in view of Sullivan and McCue (McCue is listed but not cited in a rejection). *Id.* at 402-10 (July 9, 2020 Action).

- In response, Applicant initiated an examiner interview to discuss claim amendments, but no agreement was reached. *Id.* at 400. Applicant amended pending claims to add essentially what are now limitations 1[d] and 1[f]. *Id.*, pp. 380-95 (October 6, 2020 Response).
- A Notice of Allowance issued in November 2020 with those limitations identified in the Reasons for Allowance. *Id.* at 363-70.
- Applicant submitted a pre-issuance Request for Continued Examination (RCE) including litigation claim charts for Woycik and Tengler for the '640 and '012 patents. *Id.* at 155-61.

43. The examiner twice rejected most claims as anticipated by Mueller, and rejected what are now claims 5-6 and 29-30 related to capturing videos as obvious over Mueller and Tengler. *Id.* at 134-45 (January 7, 2021 Action),[1] 93-103

[1] The Office erroneously identified Woycik as a reference cited at EX1003, 145, but Woycik was never discussed in any Office Action or used in a rejection of any claim.

(July 22, 2021 Final). I understand that this petition does not challenge claims 5-6 or 29-30; thus, Tengler was not discussed relative to, or applied against, any challenged claim.

44. In response, Applicant amended the pending claims to add the "further information"-related limitations. *Id.* at 109-27 (April 7, 2021 Response). Applicant argued that Mueller did not "disclose the receipt of 'POS transaction' information from POS terminals." *Id.* at 124. Applicant later argued that Mueller does not disclose "sending of POS transaction data [from the POS terminal to the server] that comprises any 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'" as recited by the amended claims. *Id.* at 85 (October 15, 2021 Response/CPA). After Applicant filed a supplemental response focusing on the "further information"-related limitations (*id.* at 38-57), the examiner allowed the claims, identifying limitations 1[e]-1[g] in the Reasons for Allowance. *Id.* at 19-27.

45. As explained below, PO relied on limitations regarding configuring POS screens based on "further information" during prosecution. The corresponding disclosure in the specification appears to relate only to showing

"secondary screens" based on the selections made on prior screens (information about items/options selected), i.e., using "touch keys for pretzels 11 and for drinks 12" on a first screen "would typically bring up secondary screens displaying specific product keys for ordering different types of pretzels and drinks." EX1001, 2:54-58. The discussion of "loyalty points," "promotions," and "discounts" treats them as interrelated and does not appear to disclose creating/modifying a POS screen based on such information. *Id.*, 4:49-54.

46. The '793 patent issued January 18, 2022. EX1001, (45).

VII. LISTING OF CLAIMS

Claim	Limitation
1	1 [pre] A web-based point of sale (POS) builder system comprising: 1[a] at least one server configured to: 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[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; 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; and 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; 1[g] wherein said further information regarding the one or more POS transactions relate to one or more transactions by corresponding customers respectively associated with at least one of said one or more POS terminals.
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.
3	The web-based point of sale (POS) builder system of claim 1, wherein the one or more POS terminals comprise a plurality of POS terminals in a plurality of locations.
4	The web-based point of sale (POS) builder system of claim 1, wherein the one or more items comprise at least one of: one or more items for sale, one or more promotions, or one or more loyalty points programs.
7	The web-based point of sale (POS) builder system of claim 1, wherein the information regarding one or more POS transactions comprises one or more of the employee clock information, the customer add/update information, the item add/update information, or the promotion information.
8	The web-based point of sale (POS) builder system of claim 1, wherein the information regarding one or more POS transactions are viewable via the POS builder interface.

9	The web-based point of sale (POS) builder system of claim 1, wherein the one or more display interfaces comprise one or more buttons or keys.
10	The web-based point of sale (POS) builder system of claim 1, wherein the received information comprises information indicative of at least one of a number, shape, or arrangement of the one or more display interfaces.
11	The web-based point of sale (POS) builder system of claim 1, wherein the POS builder interface is accessible via a web browser.
12	The web-based point of sale (POS) builder system of claim 1, wherein the one or more display interfaces are accessible on the POS builder interface.
13	The web-based point of sale (POS) builder system of claim 1, wherein the at least one server is further configured to: receive, over the network from the POS builder interface, second information regarding a modification to at least one of the one or more POS screens; and update the at least one of the one or more POS screens on the one or more POS terminals based on the second information.
14	The web-based point of sale (POS) builder system of claim 1, wherein the at least one server is further configured to store information regarding the one or more POS screens.
15	The web-based point of sale (POS) builder system of claim 1, wherein the at least one server is located remotely from the one or more POS terminals.
16	The web-based point of sale (POS) builder system of claim 1, wherein the at least one server is further configured to receive the information for creating or modifying the one or more POS screens and create or modify the one or more POS screens in real time while the one or more POS terminals are in use performing one or more POS transactions.

17	The web-based point of sale (POS) builder system of claim 1, wherein the one or more POS terminals use the one or more POS screens after completing a pending POS transaction.
18	The web-based point of sale (POS) builder system of claim 1, wherein the at least one server is further configured to maintain information regarding POS screens for separate sets of POS terminals separately.
19	The web-based point of sale (POS) builder system of claim 1, wherein instructions to the POS builder interface for programmatic creation and modification of the POS terminals are not formatted in programming code.
20	The web-based point of sale (POS) builder system of claim 1, wherein the at least one server is further configured to maintain information regarding the one or more POS screens.
21	The web-based point of sale (POS) builder system of claim 1, wherein the received information comprises one or more attributes of the one or more items.
22	The web-based point of sale (POS) builder system of claim 1, wherein the POS terminals are configured to perform transactions independently of a connection with the network.
23	The web-based point of sale (POS) builder system of claim 1, wherein the input interface element comprises a data interface for inputting at least some of said further information.
24	The web-based point of sale (POS) builder system of claim 1, wherein to configure the one or more POS terminals comprises dynamically configuring the one or more POS terminals specific for the corresponding customer based on the one or more transactions by the corresponding customer.
25	The web-based point of sale (POS) builder system of claim 1, wherein the POS builder interface is configured to create or modify at least one of: a position or operation of a first display interface of the one or more

	display interfaces, wherein the first display interface comprises an input interface element.
26	The web-based point of sale (POS) builder system of claim 25, wherein the input interface element comprises a touch screen input interface element.
27	27[pre] A web-based point of sale (POS) builder system comprising: 27[a] at least one POS terminal configured to: display one or more POS screens; 27[b] communicate with at least one server over a network comprising the Internet; 27[c] receive, over the network from the at least one server, 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; 27[d] perform one or more transactions with respect to the one or more items; and 27[e] transmit, 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; and 27[f] create or modify based on the received information or further information the one or more POS screens; 27[g] 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; and

	27[h] wherein said one or more POS transactions relate to one or more transactions by corresponding customers respectively associated with at least one of said one or more POS terminals.
28	The web-based point of sale (POS) builder system of claim 27, wherein the at least one POS terminal comprises a plurality of POS terminals in a plurality of locations.
31	The web-based point of sale (POS) builder system of claim 27, wherein the one or more display interfaces comprise one or more buttons or keys.
32	The web-based point of sale (POS) builder system of claim 27, wherein the received information comprises information indicative of at least one of a number, shape, or arrangement of the one or more display interfaces.
33	The web-based point of sale (POS) builder system of claim 27, wherein the at least one server is located remotely from the at least one POS terminal.
34	The web-based point of sale (POS) builder system of claim 27, wherein the at least one POS terminal is further configured to perform the one or more transactions in real time while the one or more POS screens are created or modified.
35	The web-based point of sale (POS) builder system of claim 34, wherein the at least one POS terminal uses the created or modified one or more POS screens after completing a pending POS transaction.
36	The web-based point of sale (POS) builder system of claim 27, wherein the received information comprises one or more attributes of the one or more items.
37	The web-based point of sale (POS) builder system of claim 27, wherein the one or more items comprise at least one of: one or more items for sale, one or more promotions, or one or more loyalty points programs.
38	The web-based point of sale (POS) builder system of claim 27, wherein the at least one POS terminal is further configured to perform the one or

	more transactions without a connection to the web server and transmit the information regarding the one or more transactions to the at least one web server after performing the one or more transactions and when the connection to the web server is established.
39	The web-based point of sale (POS) builder system of claim 38, wherein the at least one POS terminal is further configured to store locally the information regarding the one or more transactions.
40	The web-based point of sale (POS) builder system of claim 27, wherein a position or operation of a first display interface of the one or more display interfaces is created or modified, wherein the first display interface comprises an input interface element.
41	The web-based point of sale (POS) builder system of claim 40, wherein the input interface element comprises a touch screen input interface element.
42	42[pre] A method of implementing a web-based point of sale (POS) builder system, the method comprising: 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[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; 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; and 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;

	42[e] 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; and 42[f] wherein said further information regarding the one or more POS transactions relate to one or more transactions by corresponding customers respectively associated with at least one of said one or more POS terminals.
43	43[pre] A method of implementing a web-based point of sale (POS) builder system, the method comprising: 43[a] displaying one or more POS screens; 43[b] communicating with at least one server over a network comprising the Internet; 43[c] receiving, over the network from the at least one server, 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; 43[d] performing one or more transactions with respect to the one or more items; 43[e] transmitting, to the at least one server, over the network, further information regarding one or more transactions corresponding to the one or more items; and 43[f] creating or modifying based on the received information the one or more POS screens; 43[g] wherein the further information regarding the one or more 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; and 43[h] wherein said further information regarding the one or more POS transactions relate to one or more transactions by corresponding customers respectively associated with at least one of said one or more POS terminals.
44	44[pre] A web-based point of sale (POS) builder system comprising: 44[a] at least one server configured to: 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[c] receive, over the network from a POS builder interface, information for creating or modifying functionality of the one or more POS terminals; 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; and 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; 44[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; and 44[g] wherein said further information regarding the one or more POS transactions relate to one or more transactions by corresponding

	customers respectively associated with at least one of said one or more POS terminals.

VIII. CLAIM CONSTRUCTION

47. It is my understanding that the claims of the ’793 patent should be given their ordinary and customary meaning for the purposes of this proceeding, and that claim terms should be given the meaning that the terms would have to a person of ordinary skill in the art at the time of the invention. This is the meaning that I have applied to the claims in my analysis.

48. I also understand that several terms of the patents to which the ’793 claims priority were previously construed by a district court in a prior lawsuit filed by Patent Owner. EX1024, EX1025, EX1029. Most terms were given their plain and ordinary meaning. I have reviewed the Final Claim Construction Order and Supplemental Claim Construction Order in that lawsuit, but these orders do not impact my analysis in this declaration.

A. “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…" (all claims)[2]

49. This term recites two lists of alternatives and is met if any of "the information," "the further information," **or** "a combination thereof" includes at least one of the enumerated information types. This is a natural reading of this language in the context of a software-based system. A POSITA would have understood that "a combination" of "the information" and "the further information" includes a combination of all or some of each category of information (e.g., if information from each category is used to create/modify a POS screen).

B. "display interfaces" (all claims)

50. A "display interface" is an area of a POS screen, e.g., for displaying content and/or inputting information. In the context of POS systems, display interfaces would at least include buttons or keys for display on POS screens and for receiving user selection input on those POS screens. EX1001, Claim 9.

C. "the input interface element" (claim 23)

51. The term "input interface element" appears in claims 23, 25, and 26, however claim 23 recites "the input interface element" without antecedent basis.

[2] Claim 43 replaces "POS transactions" with "transactions" but is treated the same here.

For purposes of this proceeding, I assume claim 23 includes "the display interface comprises an input interface element" limitation of claim 25, and address claim 23 under that reading.

D. "the web server"/"the at least one web server" (claim 38)

52. Claim 38 depends from claim 27 and recites "the web server" and "the at least one web server," neither of which has antecedent basis. I assume here these terms refer to "at least one server" recited in claim 27.

E. "creating or modifying functionality of the one or POS terminals" (claim 44)

53. System claim 44 recites similar limitations to system claim 1, but more broadly claims "creating or modifying *functionality of the one or POS terminals*" (44[c]) rather than "creating or modifying *the one or more POS screens*" (1[c]) (emphasis added). A POSITA would have understood the latter is a subset of POS terminal functionality because modifying POS screens (e.g., modifying which buttons appear on the screen, their layout, and which items are associated with them) modifies how the POS terminal functions (e.g., which screens are displayed and which items/options are added to the cart when particular buttons/keys are selected). Limitation 44[f] in view of 44[c] supports this understanding by reciting, "***information*** for creating or modifying ***functionality of***" POS terminals (44[c]) followed by "***the information*** used for creating or

modifying ***the one or more POS screens***" (44[f])—the latter of which lacks antecedent basis in claim 44 and suggests creating/modifying functionality includes at least creating/modifying POS screens.

54. I reserve the right to respond to constructions offered by Patent Owner or adopted by the Board.

IX. OVERVIEW OF THE PRIOR ART

A. Overview of Woycik (EX1004)

55. Woycik was filed May 1, 2007 and published November 15, 2007, both prior to the '793 patent's priority date of February 5, 2008. EX1005, (22). (43). Woycik is prior art under at least §§ 102(a) and (e).

56. Woycik discloses a web-based administrative tool for building and editing user interfaces displayed on POS screens of self-service POS kiosks. EX1004, Abstract, [0071]-[0073], Fig. 1. The administrative tool is installed on a central server and accessible by "web access." *Id.*, [0075]. The administrative tool and its user interfaces enable "the administrator to perform various administrative functions such as configuring kiosks [and] creating and editing menus and available food items." *Id.*, [0073]. Kiosks are configured with information from the administrative tool's user interface to create and modify POS screens with touchscreen buttons/keys displayed during a transaction based on a manager's

menu/item/design/layout instructions and based on further information about POS transactions received by kiosks and transmitted to the central server (including order histories, items added/updated, and other customer- and order-specific information), e.g., to display a series of hierarchical/sequential screens based on customer additions/updates/selections and to allow returning customers to "quickly order items ordered in the past without having to rebuild the order each time they use a kiosk." [0089], [0091], [0140]-[0142], Fig. 40.

B. Overview of Tengler (EX1005)

57. Tengler published March 3, 2005, more than one year before the '793 patent's priority date of February 5, 2008. EX1005, (43). Tengler is prior art under at least § 102(b).

58. Tengler discloses an order processing system for merchants. EX1005, Abstract. Restaurant Management Software is installed on a web server, and POS terminal devices run register and self-service applications and display POS screens with touchscreen buttons/keys. *Id*., [0095]. Tengler's system allows "a manager to access a management database of a quick-serve restaurant location remotely through a web interface." *Id.*, [0022]. The manager's web interface provides access to "[a] user interface designer 614 [that] allows management to edit the user interface of the register and self-service applications" running on POS terminals.

Id., [0103]. POS terminals/kiosks are configured with information from the manager's web interface to create and modify hierarchical/sequential POS screens displayed based on a manager's menu/item/design selections and based on further information about POS transactions at terminals/kiosks that is transmitted to the web server (including items added/updated, price and tax, loyalty/promotion information, and images of the customer). *Id.*, [0017]-[0018], [0076], Figs. 9-10.

X. THE PRIOR ART GROUNDS RELIED UPON HEREIN ARE DISTINCT FROM THOSE CONSIDERED DURING EXAMINATION

59. I have been asked to review the prosecution history of the '793 patent, and to compare the prior art grounds presented herein to those considered by the examiner during the examination of the '793 patent. While Woycik and Tengler are listed on the face of the '793 patent, Woycik was not discussed during prosecution or used in a rejection. The reference at EX1003,145 appears to be an error because the Examiner never discussed Woycik in an office action rejection. *See generally* EX1003.

60. Also, the Examiner only used Tengler in a very limited fashion. The Examiner only cited Tengler against video-related dependent claims (claims 5-6 and 29-30). *E.g.*, EX1003, 101 (addressing claims 8-9 and 28-29, which resulted in issued claims 5-6 and 29-30). Thus, Tengler was never discussed by the Examiner with regard to the claims challenged in this declaration.

61. Accordingly, it is my opinion that the grounds of invalidity considered herein are distinct from those considered by the examiner during examination.

XI. GROUND 1: WOYCIK IN VIEW OF THE KNOWLEDGE OF A PERSON OF ORDINARY SKILL IN THE ART RENDERS CLAIMS 1-4, 7-28, AND 31-44 OBVIOUS

A. Claim 1

1. 1[pre][3]

"A web-based point of sale (POS) builder system comprising"

62. Woycik discloses a computer-based POS system "for ordering goods and services [that] includes a plurality of self-service client terminals and a server." EX1004, Abstract; *see also id.*, [0006]-[0009], [0129]. The system includes an "administrative tool" (POS builder) providing "a menu editor that enables the administrator to create and edit the interactive menu screens" displayed on self-service terminals/kiosks. *Id.*, Abstract; *see also id.*, [0011]-[0012], [0016]-[0017], [0041], [0050]-[0051], [0079], [0081], [0097]-[0099], [0112] (item editor). The administrative tool is available via "web access" to a central server "using a

[3] I understand PO contends the preamble is not limiting. EX1017, 5. To the extent any preambles are limiting, they are disclosed in each ground.

simplified user interface … from any Internet-connected computer." *Id.*, [0075], [0080].

63. Accordingly, it is my opinion that Woycik discloses each limitation of 1[pre].

2. 1[a]

"at least one server configured to"

and 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"

64. Woycik discloses a computer-based "point of sale system for ordering an item" that "includes a plurality of self-service client terminals and a server." EX1004, [0129], Abstract; *see also id.*, [0004]-[0008]. "[C]entral server 22" is configured to communicate with kiosks 16, 30, and/or 36 over a network comprising "Internet 18." EX1004, [0071], Fig. 1:[4]

[4] Annotations/alterations to figures and text added.

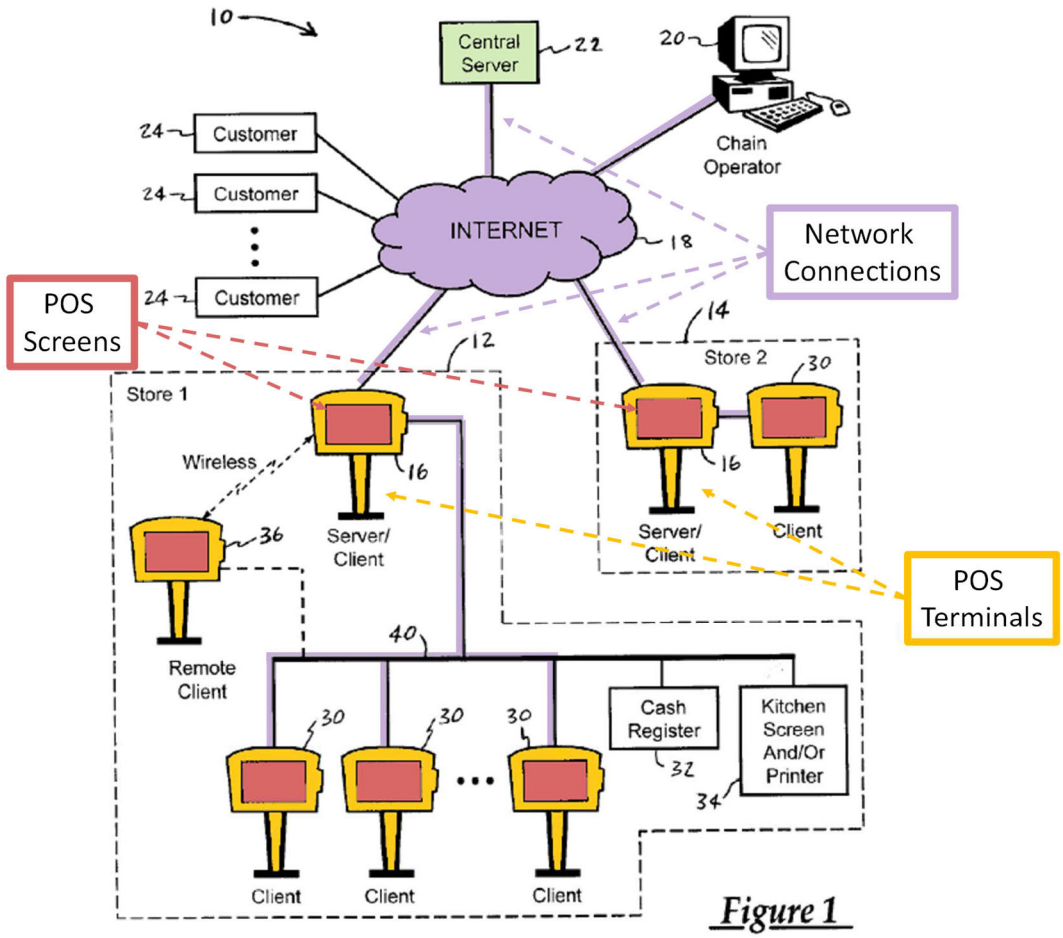


Figure 1

When discussing software, Woycik refers to central server 22 as "central server 84" and kiosk 16 as "client/server kiosk 82." *Id.*, [0078], Fig. 3:

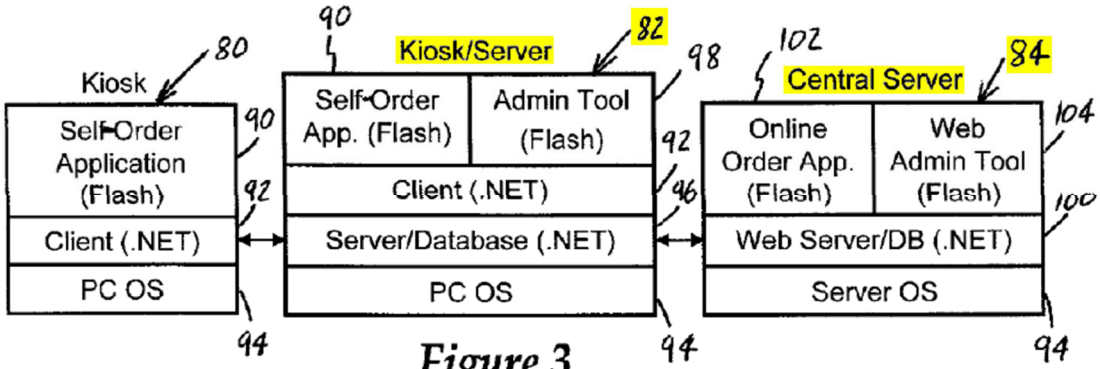


Figure 3

Woycik thus discloses kiosks 16/82 communicate with the central server over the Internet (thus disclosing 1[a] and 1[b]), while kiosks 30/36 communicate with the central server over a network comprising the Internet and network 40 (additionally disclosing 1[a] and 1[b]). *Id.*, [0074] (any "device on the network 40 can be connected to … the Internet 18"); *see also id.*, [0124], Fig. 27. A POSITA would have understood that these open-ended comprising claims contemplate a network that includes the Internet and other network nodes/elements, e.g., devices interconnected via a local area network connected to the Internet via a local server.

65. Kiosks operate as "self-service terminals" (POS terminals) that display "interactive menu screens having buttons" (POS screens). EX1004, Abstract, [0014]-[0016], [0077], [0082], [0100], Figs. 4-9:

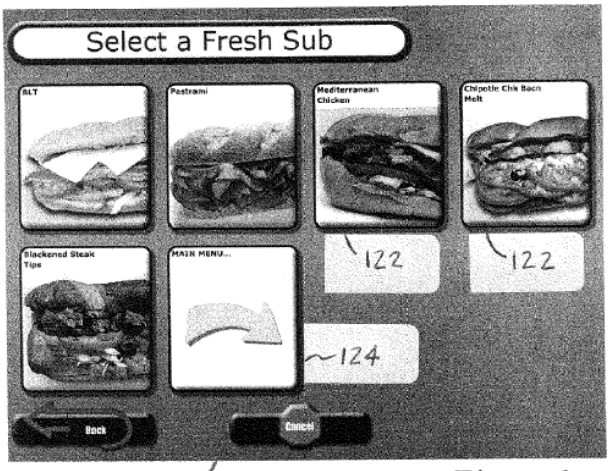


Figure 4

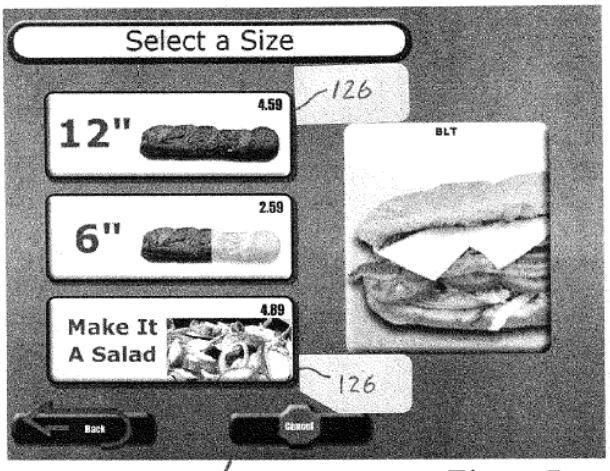


Figure 5

Specifically, kiosks/terminals display "a set of interactive screens that guide the customer through the process of placing an order." *Id.*, [0082]; *see also id.*, [0073], Fig. 3.

66. In sum, Woycik discloses at least one server (central server 22/84) configured to communicate with one or more POS terminals (kiosks 16/82 and/or kiosks 30/36) over a network comprising the Internet (Internet 18 alone or in combination with network 40), wherein the one or more POS terminals are configured to display one or more POS screens (interactive menu screens depicting buttons/keys, e.g., Figs. 4-9). Accordingly, it is my opinion that Woycik discloses each limitation of 1[a] and 1[b].

3. 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"

67. Woycik discloses "using a standard web browser" and Internet 18, chain operator 20 has web access to an "administrative tool loaded on the central server 22"/84. EX1004, [0075]; *see also id.*, [0017], [0071], [0080]-[0081], [0121]-[0122], Figs. 1, 3. The "administrative tool" includes a "**user interface application** [POS builder] 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." *Id.*, [0073], [0079]. "Thus, the store owner or chain operator can carry out administration of the system **using a simplified user interface**" (POS builder interface) "from any Internet-connected computer." *Id.*, [0080], [0121].

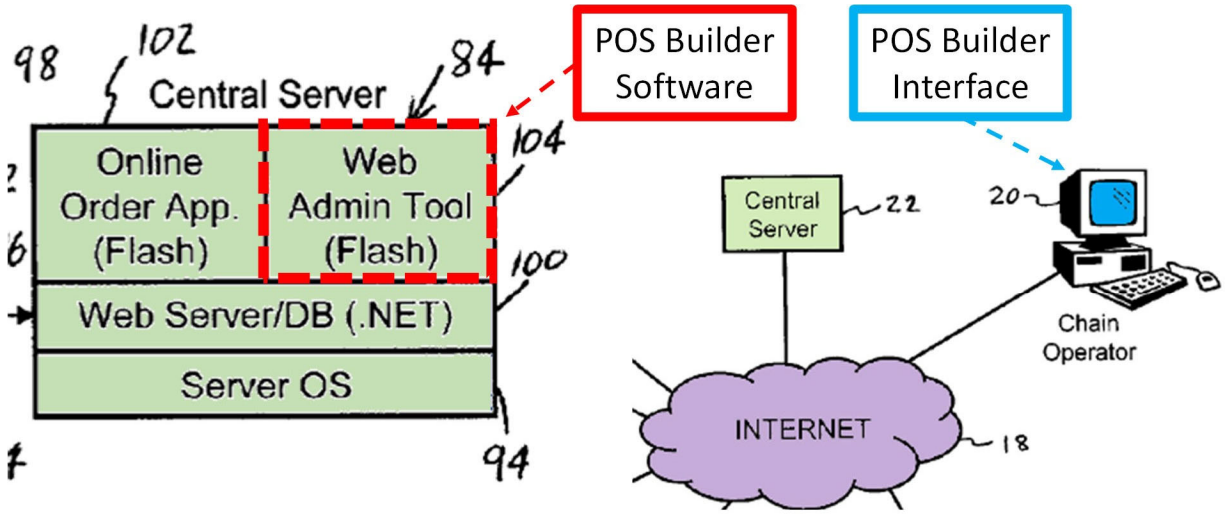


Fig. 3 (excerpted) **Fig. 1 (excerpted)**

68. The main menu interface to the web-based "administrative tool" displays options including "system settings 218, menu editor 220, [and] item editor 222." EX1004, [0097], Fig. 10 (excerpted):

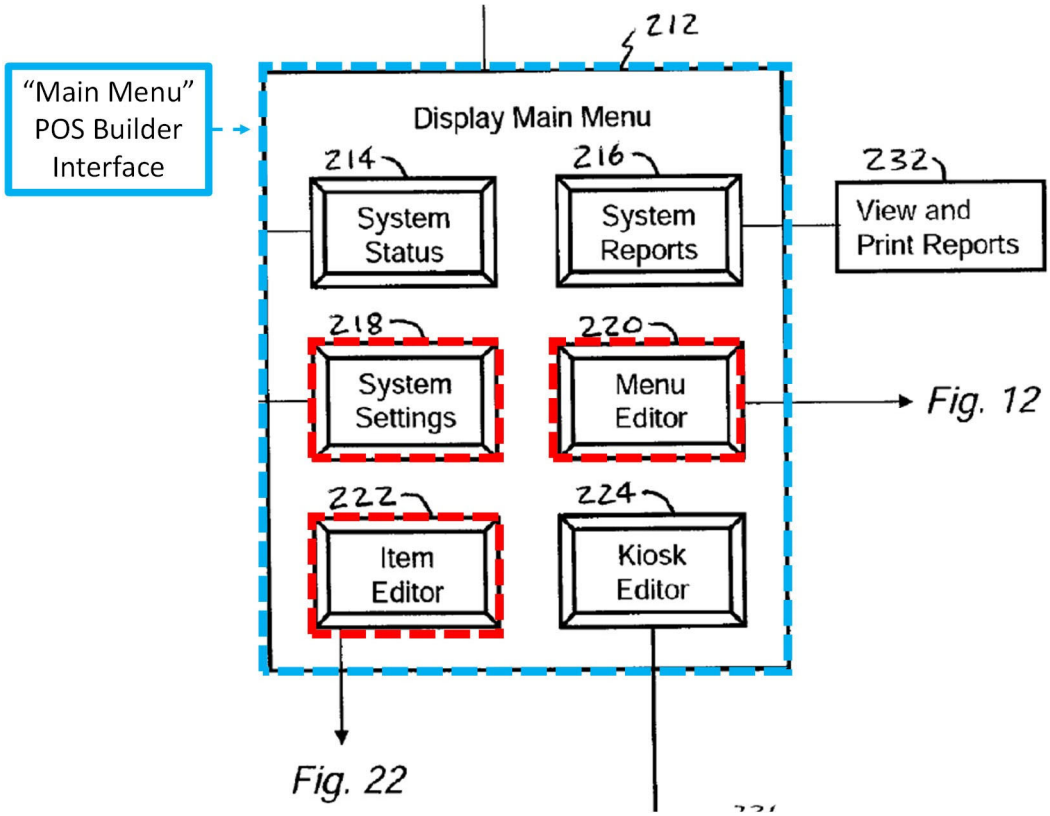


Fig. 12

Fig. 22

See also id., [0132], Fig. 34:

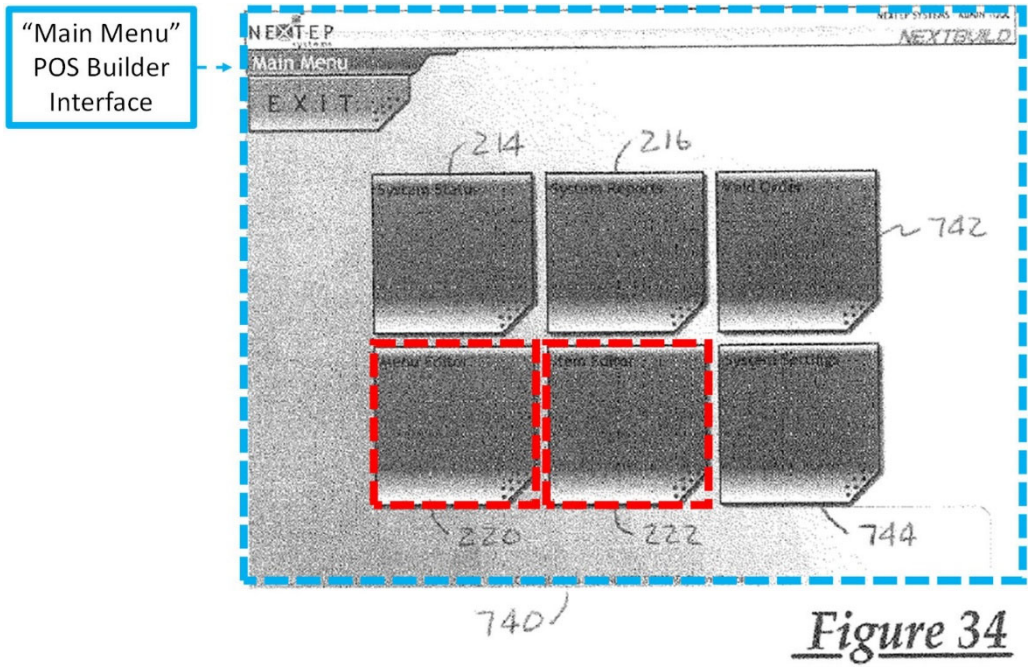


Figure 34

"System settings 218 enable the administrator … to … enter sales tax percentages [and] enter customer ids." *Id.*, [0097], Fig. 10. The "menu editor [220]" "enables the administrator to create and edit the interactive menu screens provided by the self-order application" "at the self-service client terminals" in stores, including the "hierarchy of menus to guide the customers through the order process." *Id.*, [0013], [0016], [0109]; *see also id.*, [0019]-[0020], [0073], [0079], [0098]-[0099], Figs. 11-13:

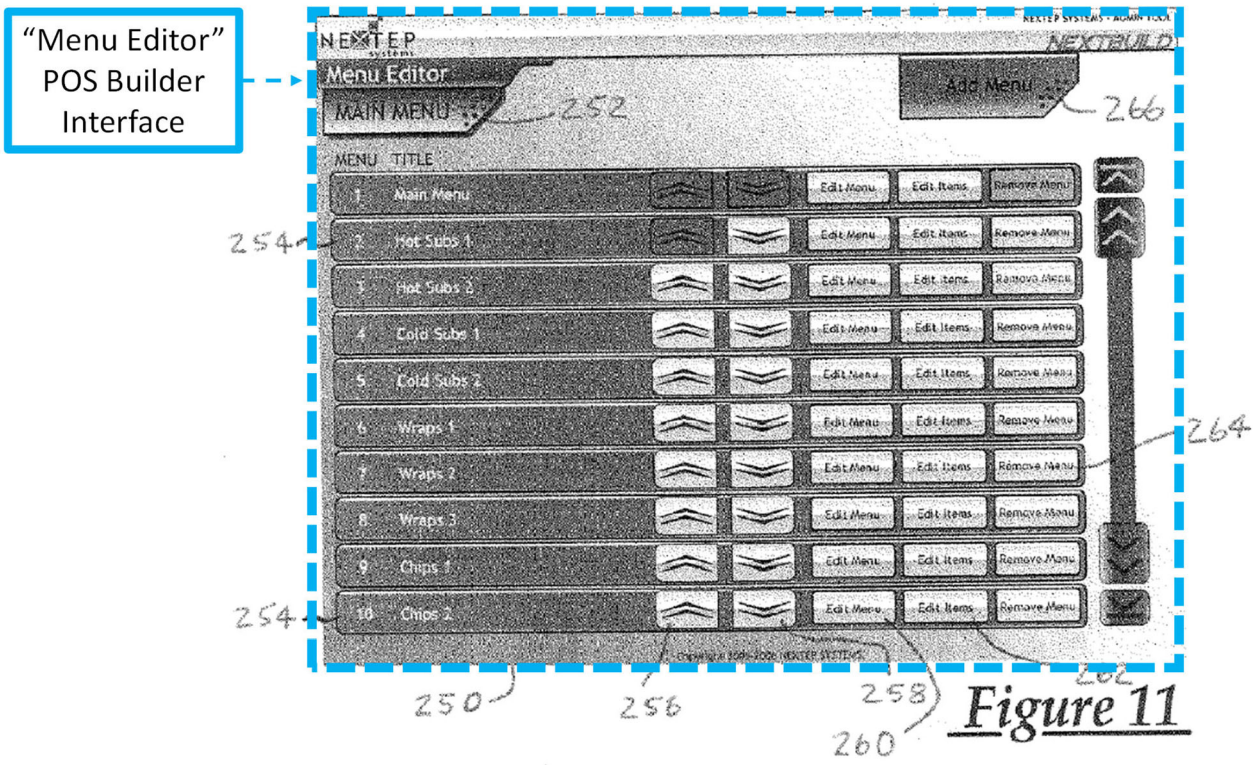


Figure 11

"The individual items or products available for sale are edited in the item editor" where the administrator can "configure the various food items and options." *Id.*, [0112]; *see also id.*, [0110]-[0113], Figs. 20-22:

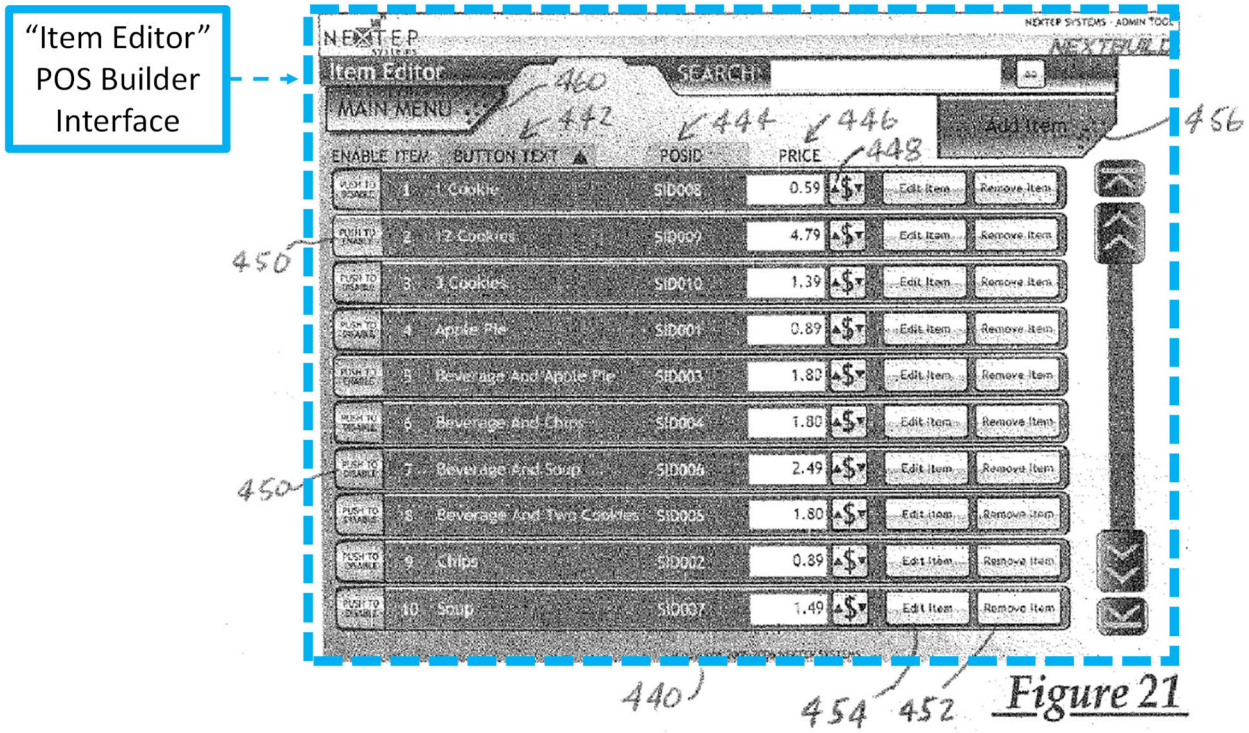


Figure 21

69. "The primary components of each menu screen are **buttons** that allow the customer to make selections among available items and options." *Id.*, [0100]. Buttons appearing on Woycik's POS screens are display interfaces. *See supra* § VIII.B (construing "display interfaces"). The administrative tool and its interface allow the manager/administrator to define the layout (number, shape, and arrangement) of buttons on POS screens, either individually or using a template. *Id.*, [0098], [0100], [0112]; *see also id.* [0019]-[0020],[0079], [0082], [0101]-[0107], Figs. 4-9:

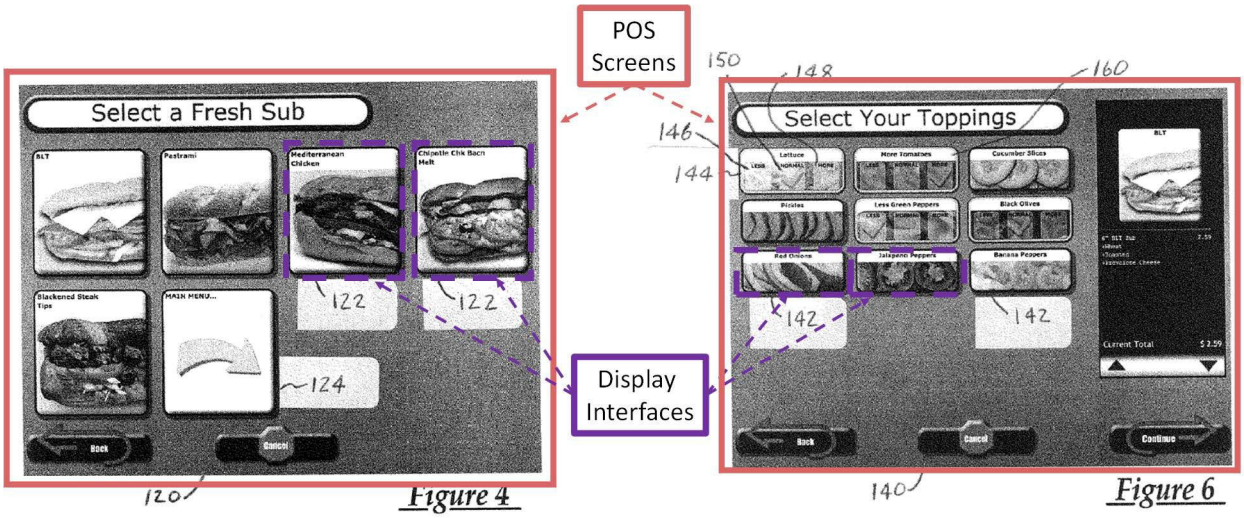


Figure 4 *Figure 6*

The manager/administrator can "edit the individual buttons and **associate them with food items and options** and associated button text, graphics, voiceovers, and button behaviors." *Id.*, [0108], Fig. 12:

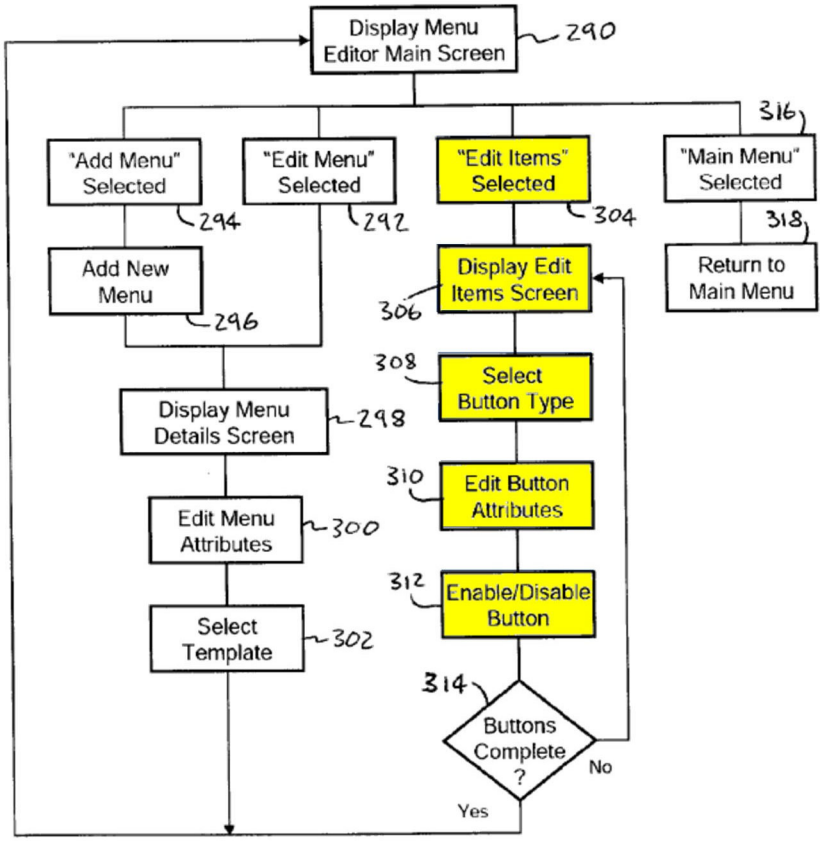


Figure 12

The server stores configuration changes received from the administrative tool user interface and can automatically send configuration updates to in-store kiosks 16/82. EX1004, [0115], [0121]-[0122].

70. In sum, Woycik discloses information received by central server 22/84 from the administrative tool user interface (POS builder interface) over the network including Internet 18 is used for creating/modifying interactive menu screens (POS screens) displayed on POS terminals, including information used for creating/modifying buttons/keys (display interfaces) associated with items on said

screens. Accordingly, it is my opinion that Woycik discloses each limitation of 1[c].

4. 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"

and 1[g]

"wherein said further information regarding the one or more POS transactions relate to one or more transactions by corresponding customers respectively associated with at least one of said one or more POS terminals"

71. Woycik discloses central server 22/84 receives further information over the network (including Internet 18) regarding POS transactions by corresponding customers ordering items in the transaction from kiosks 16/82 (POS terminals), e.g., customer information, order and item details, payment (including tax information), and loyalty/discount/promotion information. EX1004, [0028], [0030], [0089]-[0092].

72. Central server 22/84 retains information regarding "previous orders by the customer" received from kiosks 16/82, e.g., items the customer added/updated in those orders. *Id.*, [0089], [0091]. "Customer orders or other information that is to be retained to enhance the customer experience" is received from the POS terminals/kiosks 16/82 and "stored at the central server," e.g., in a "table or

database," and can be provided "to the central server in real-time throughout the day." *Id*., [0030]; *see also id.*, [0089], [0128], [0141].

73. Woycik discloses and suggests the further information relates to POS transactions by corresponding customers respectively associated with kiosks 16/82 because customers use respective POS terminals to place orders, and the database at the central server stores information about customers and their respective orders to facilitate customer recognition and "provide an enhanced customer experience." *Id.*, [0028]-[0030], Figs. 39-40.

74. Accordingly, it is my opinion that Woycik discloses and suggests each limitation of 1[d] and 1[g].

5. 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"

75. As explained for limitation 1[c], Woycik's central server receives information (input by the manager/administrator) over Internet 18 from the administrative tool's user interface (POS builder interface) used to create or modify interactive menus (POS screens) displayed on kiosks 16/82 (POS terminals). EX1004, [0071], [0073], [0075]. Central server 22/84 "automatically"

transmits configuration information and updates (e.g., menus/items and edits/additions) to kiosk 16/82 over Internet 18, and kiosks 16/82 display POS screens using the new or updated configuration information. *Id.*, [0122]; *see also id.*, [0115], [0076]. Kiosks use such information and "graphical user interface application 90" to display "a set of interactive screens that guide the customer through the process of placing an order." *Id.*, [0078], [0082], Fig. 3.

76. When selecting items/options, "pricing information is updated live on the display … as items are added to or removed from the order." EX1004, [0090].

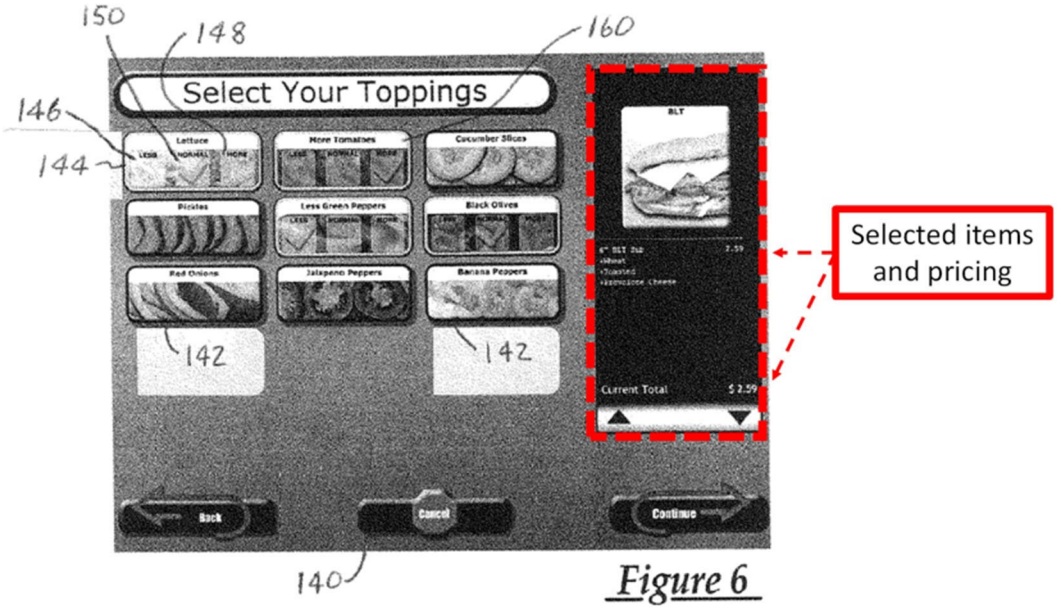


Figure 6

Customers can change toppings and amounts using buttons, e.g., "to eliminate the cheese and add tomatoes … in which case the changes are registered by the system, the American Cheese overlay buttons are eliminated, and new overlay

buttons are displayed on the tomato button" with different text displayed on the buttons and updated price information. *Id.*, [0088], Figs. 7-9:

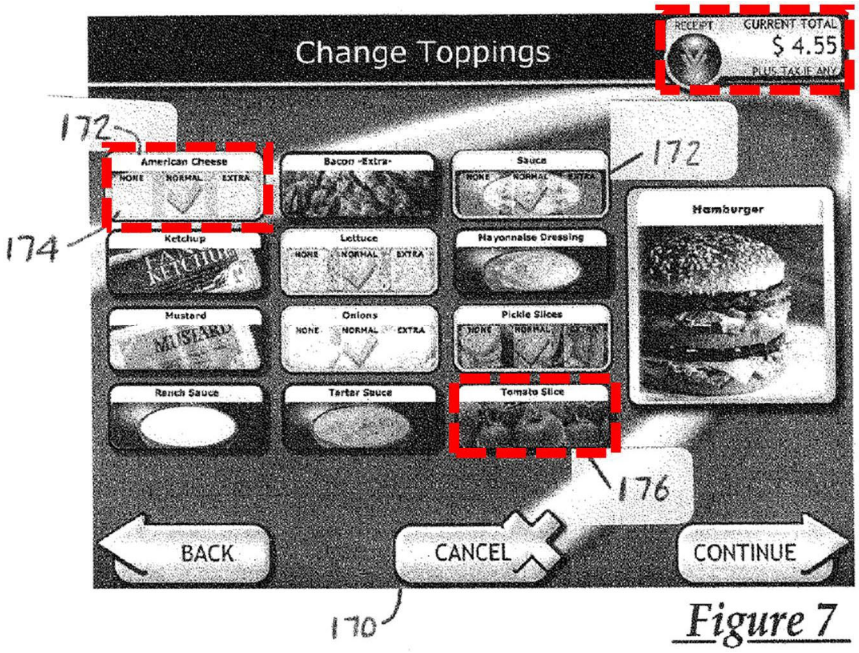


Figure 7

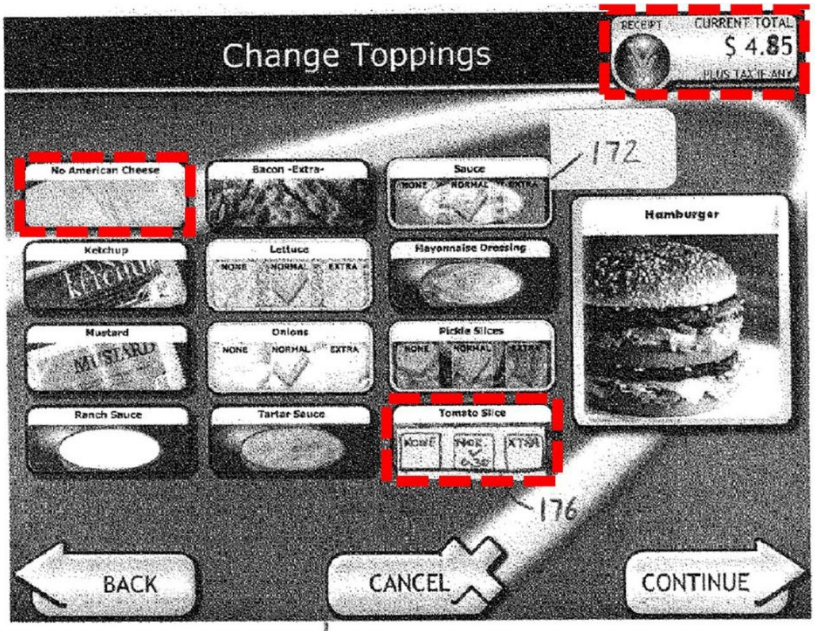


Figure 8

Woycik thus discloses and suggests managers may specify a POS screen with menu categories and subsequent POS screens for items/options within such categories for display based on a customer's category and item selections. *Id.*, Figs. 4-6 (illustrating successive POS screens based on user selections). Information from the POS builder interface is thus used to configure kiosks 16/82 (POS terminals) to display new or modified POS screens based on items and options added/updated to an order ("further information," which is also stored by server 22/84 as explained for 1[d]).

77. Woycik also discloses examples of configuring kiosks with information received from the POS builder interface to create/modify POS screens based on "further information" regarding specific customers. Such further information is transmitted from the POS terminal and received and stored at the central server. EX1004, [0028], [0030], [0089]-[0092]. The system may recognize returning customers using "a customer loyalty card" or "a credit card" and, using information from the central server relating to the customer's previous transactions, modify the POS screens displayed based on such further information. *Id.*, [0028], [0089], [0091], [0140]-[0142]. "Customer recognition has many advantages" and "allows the system to provide an enhanced customer experience." *Id.*, [0140], [0028].

78. "One advantage … is that returning customers can quickly order [and further customize] items ordered in the past without having to rebuild the order each time they use a kiosk." *Id.*, [0140]. "For this purpose, the system tracks customer orders and stores them on the server." *Id.*, [0089]. "FIG. 40 illustrates a customer recognition screen 860 displayed by the self-order application when a returning customer is recognized." *Id.*, [0142], Fig. 40:

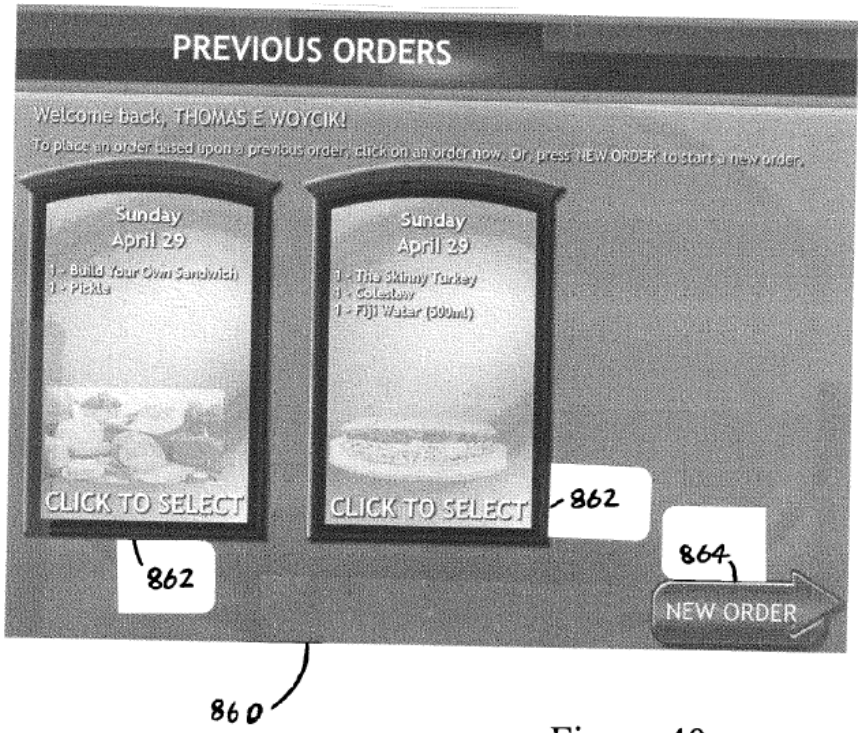


Figure 40

"The customer may choose to duplicate an entire previous order or individual items from that order," e.g., he "can simply select an item to customize (if desired) and add to his current order." *Id.*, [0089].

79. Woycik thus discloses and suggests that the central server configures kiosks with information from the POS builder interface to create or modify POS screens based on information stored at the central server concerning previous orders ("further information"). For example, a recognized customer may see the POS screen of Fig. 40. Woycik suggests if a customer selects a displayed previous order, the customer would be taken to the item customization screen, like that shown in Fig. 7, "to modify the contents of the item selected … by touching one of the contents on the touch screen" in the same manner disclosed for new orders but with the customer's prior selections pre-selected. *Id.*, [0089]; *see also id.*, [0088], Fig. 7:

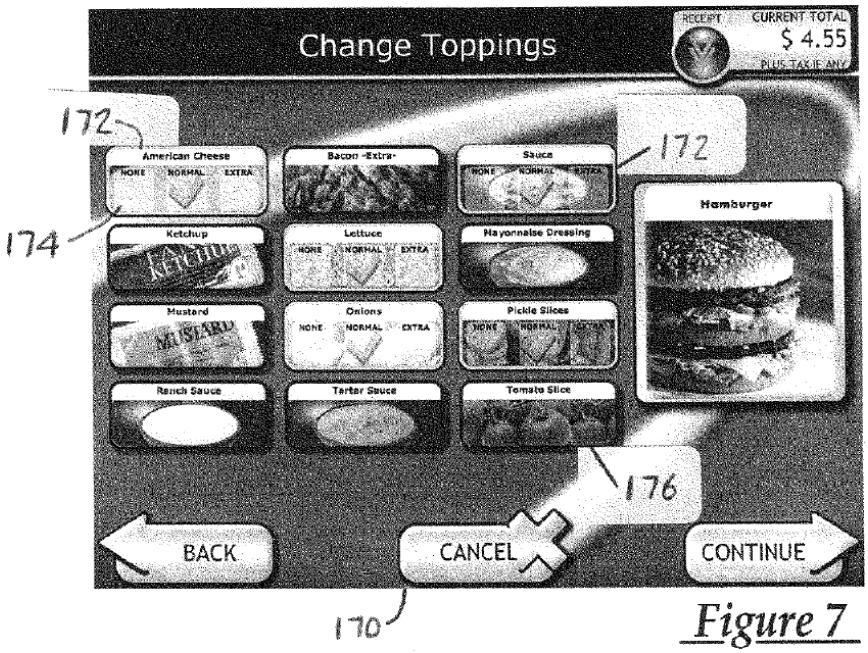


Figure 7

80. A POSITA would have understood the default toppings here would be based on the customer's prior order that was selected when starting the current order as this would be the user's expectation when adding an item from a prior order. Woycik describes this process as: "Previous orders 862 are displayed so that the customer, if desired, may quickly order one of the customer's previous orders without having to rebuilding the order." EX1004, [0142]. A user adding item(s) from a prior order would expect those item(s) to be identically configured as in the prior order to avoid having to reenter customizations.

81. A POSITA would have understood, and it would be obvious, that the layout and functionality of "customer recognition screen 860" is configured based on information received from the administrator. EX1004, [0142]. In other words, the administrator designed the layout and functionality of customer recognition screen 860. But customer recognition screen 860 is also configured based on information about and the past order history of a particular customer. In particular, the display of a customer name "Thomas E Woycik" at the top of customer recognition screen 860 indicates that the customer has been recognized—e.g., based on having previously entered his name or credit card or other information upon which the system could have identified the customer and his name. Other examples might include a phone number or loyalty card number. A POSITA would

have understood this information and the past order history would have originated from the POS terminals and be stored at the central server.

82. Another advantage enables "the system … to determine the frequency of customer visits and track customer habits and tastes. Using this information, the system can then recommend new or additional products to the customer, or can provide the customer with a customer loyalty discount or award." EX1004, [0089]. The "upsell feature" suggests items based on "the order history of recognized customers" and "shows the customer a cost savings for combining the extra item with those already selected." *Id.*, [0091]. This suggests a modified POS screen displays specific items based on historical customer/order data. A POSITA would have understood the "upsell feature" would only be useful if it modified some aspect of the POS screen to visibly suggest an additional item or item upgrade.

83. Woycik thus uses manager-defined screen layouts/functionalities to create/modify POS screens to display based on customer- and order-specific information received before and during a transaction. Because the web-based administrative tool is used to "edit the screen displays that the customers see" (EX1004, [0095], [0041]), a POSITA would have understood Woycik discloses and suggests information from the administrative tool user interface (POS builder interface) over the network (1[c]) is used to create/modify POS screens

(creating/modifying buttons, items, options, layouts, categories, etc.), including creating/modifying POS screens based on further information regarding POS transactions (e.g., customer and order information, such as a customer's identity, items added/updated in an order, customer's prior orders and selections, and loyalty/discount/promotion information) stored at central server 22/84.

84. Accordingly, it is my opinion that Woycik discloses and suggests each limitation of 1[e].

6. 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"

85. Woycik discloses information received from the administrative tool's user interface (POS builder interface) used to create/modify interactive menus (POS screens) displayed on kiosks (POS terminals) includes item add/update, customer add/update, tax, and cost information—each of which independently satisfies this limitation. *Seesupra* § VIII.A.

86. The "administration tool" provides "a menu editor" to create/modify POS screens. *Id.*, [0013], [0016]-[0017], [0041], [0050], [0108], [0112], Figs. 11-

13, 20. It further provides "an item/product editor screen … which enables the editing of the individual items/products for sale." *Id.*, [0051], Fig. 21:

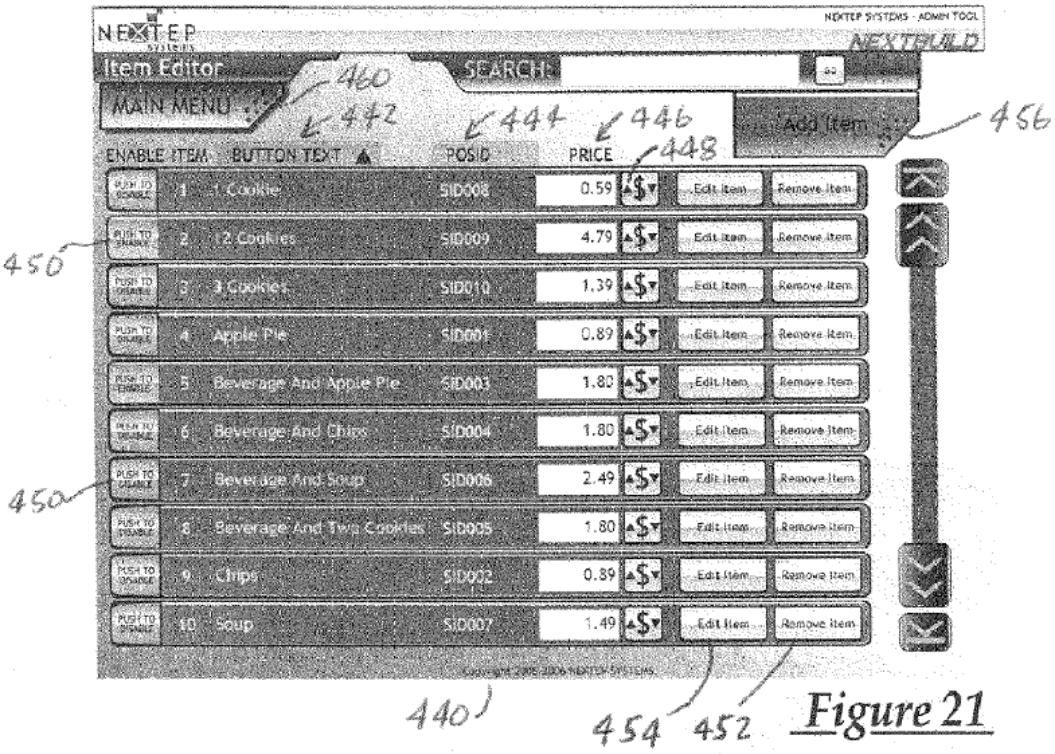


Figure 21

See also id., [0112], Fig. 22:

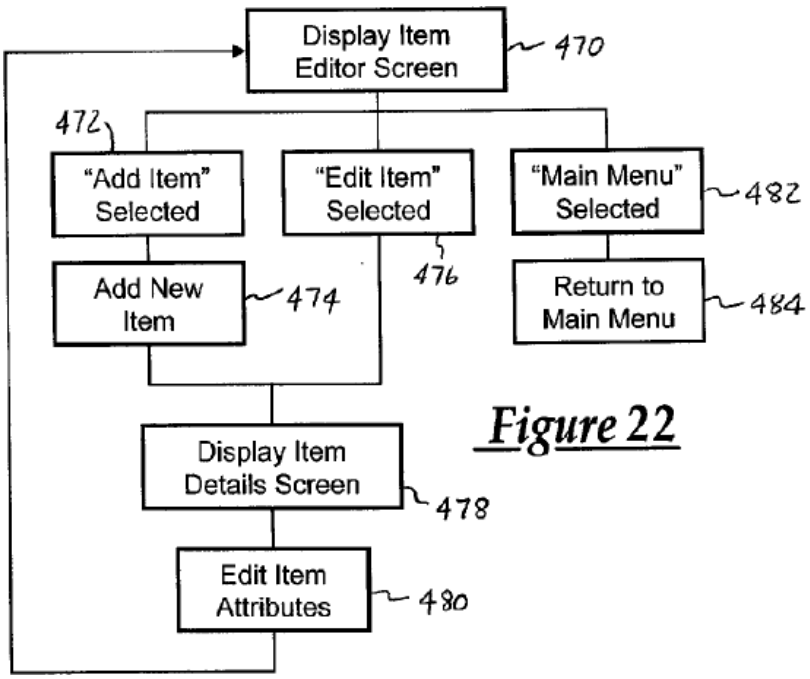


Figure 22

Thus, information from the POS builder interface is used for creating/modifying POS screens, including item add/update information. As explained below, customers also add/update items during the ordering process.

87. In addition to "creating and editing menus and available food items," the administrator can "specify[] tax and payment features of the system," including "sales tax percentages" (taxation information), "enter customer ids" (customer add/update information), and add "pricing information related to the cost" of additional items (item cost information). *Id.*, [0073], [0079], [0097], [0088].

88. As explained for 1[e] regarding further information, Woycik discloses customers provide item add/update information by "selecting items, modifying ingredients, adding side items, etc.," using the interactive menus (POS screens)

displayed on the self-service kiosks (POS terminals) to place orders (POS transactions). EX1004, [0089]. Further, returning customers "may choose to duplicate an entire previous order or individual items from that order," e.g., they "can simply select an item to customize (if desired) and add to [the] current order." *Id.*. As explained for 1[d], 1[g], order and other information retained to enhance the customer experience is transmitted from the POS terminals to central server 22/84, e.g., "in real-time throughout the day." *Id*., [0030], [0089]; *see also id.*, [0128], [0141].

89. Additionally, Woycik discloses returning customers are recognized by "swiping a customer loyalty card" and the system "can provide the customer with a customer loyalty discount or award." EX1004, [0089]. Thus, further information regarding POS transactions, alone and/or in combination with information used to create/modify screens, includes promotion, discount, and loyalty program information in addition to item add/update information. Based on the teachings of Woycik, a POSITA would have understood and been motivated to display promotions/discounts as buttons/options on the kiosk's POS screens for selection by a customer (that are configurable like other buttons/options), and information about redemption of a promotion/discount would be communicated to the central server as part of the order information. A POSITA would have understood that

promotions and discounts are only effective (i.e., they influence a user's behavior) if the user is aware of the promotions and discounts. A POSITA would have understood the benefit of displaying the promotions and discounts at the point of sale and would have been motivated to incorporate those marketing tools into the POS screen to inform and motivate the user to act, and to track/store utilization/redemption information. An alternative would be to advertise promotions and discounts external to the POS terminal, but that would lead to confusion if the user did not see the promotions and discounts reflected in the POS screens. If the promotions and discounts were not reflected in the POS screens, a user might believe the external advertisements were expired or inapplicable for some reason and would be unlikely to act on the promotions and discounts.

90. Accordingly, it is my opinion that Woycik discloses, suggests, and renders obvious limitation 1[f].

B. Claims 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*"

and 11

"*The web-based point of sale (POS) builder system of claim 1, wherein the POS builder interface is accessible via a web browser*"

91. As explained for 1[c], "the store owner or chain operator can carry out administration of the system using a simplified user interface" to the administrative tool (POS builder interface) "from any Internet-connected computer (such as a home office computer)" (computing device). EX1004, [0080]; *see also id.*, [0081], [0121], Figs. 11, 13-21, Fig. 1:

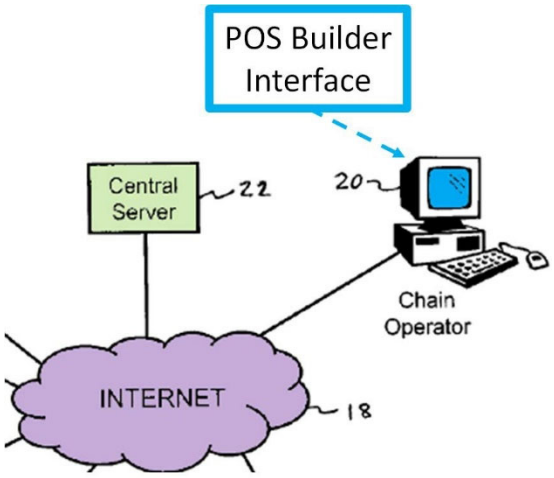


Fig. 1 (excerpted)

The administrative tool, which provides the administrative tool user interfaces, is installed on central server 22/84 (also a computing device). *Id.*, Figs. 1 & 3. The

POS builder interface thus runs on a computing device, which seemingly refers to the device accessing the POS builder interface, e.g., a manager using a computer's browser, but would also disclose this element to the extent it requires the POS builder to be installed/running on the central server. Regarding claim 11, administrators can "access the administrative tools remotely using a standard web browser." *Id.*, [0121].

92. Accordingly, it is my opinion that Woycik discloses and suggests the limitations of claims 2 and 11.

C. Claim 3

"The web-based point of sale (POS) builder system of claim 1, *wherein* ***the one or more POS terminals comprise a plurality of POS terminals in a plurality of locations"***

93. As explained for 1[b], Woycik discloses a plurality of kiosks 16/82 (POS terminals) in a plurality of locations—e.g., one in "Store 1" and one in "Store 2." EX1004, [0071], [0074], Fig. 1:

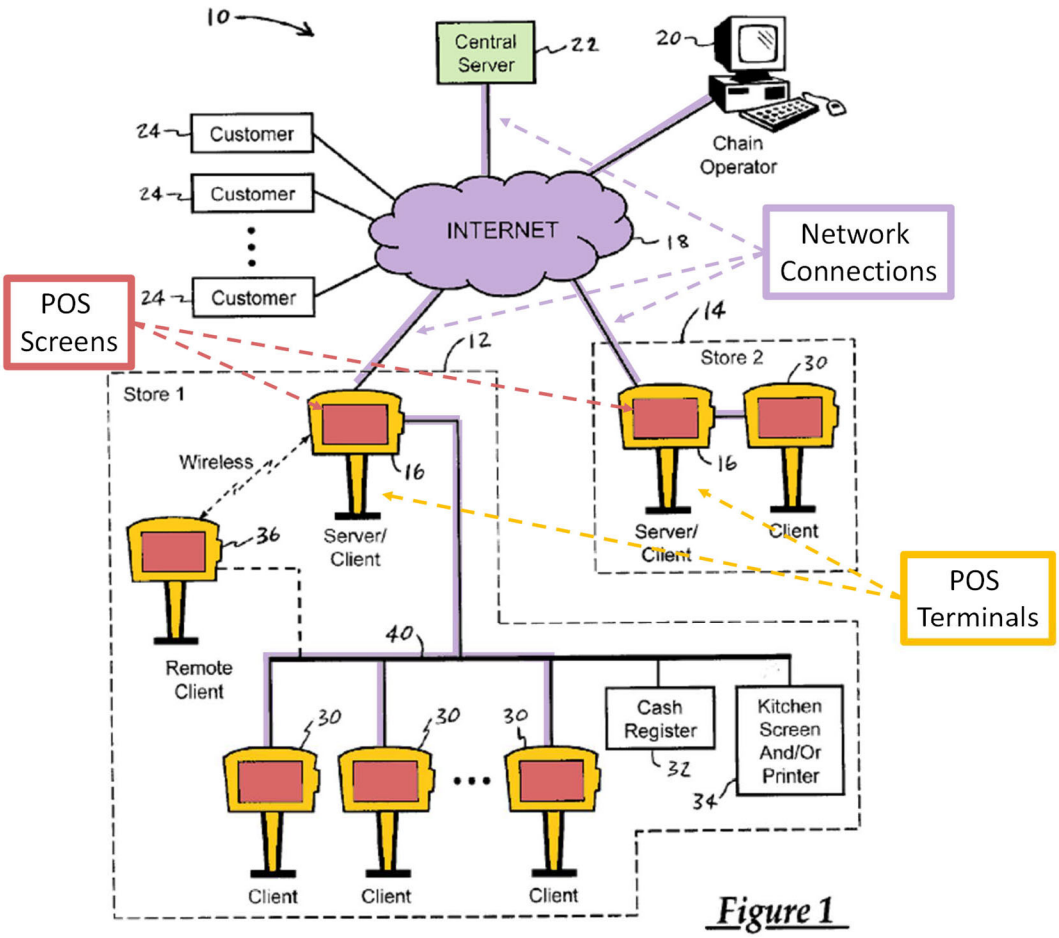


Figure 1

94. Woycik also discloses and suggests a plurality of POS terminals in *each* of a plurality of locations in two ways. First, as discussed for 1[b], Woycik discloses additional kiosks 30 and 36 in/around each store connected to central server 22/84 over Internet 18 via kiosks 16/82 and "network 40." *Id.*, [0071]-[0072], [0074], Fig. 1. Central server 22/84 thus communicates with kiosks 30/36 over the network comprising the Internet. A plurality of POS terminals in each store location would thus be connected to the central server over the network including the Internet, including indirectly via a local area network connection to

kiosks 16/82 and/or directly by using the "web installation service" provided by the central server. *Id*., [0124], Fig. 27.

95. Second, Woycik discloses "a single store location will include *at least one* local server [kiosk] 16." EX1004, [0071]. That suggests a single location may have more than one (a plurality) of kiosks 16/82, which are directly connected to central server 22/84 over Internet 18.

96. Accordingly, it is my opinion that Woycik discloses and suggests the limitations of claim 3.

D. Claims 4

> ***"The web-based point of sale (POS) builder system of claim 1, wherein the one or more items comprise at least one of: one or more items for sale, one or more promotions, or one or more loyalty points programs"***

and 9

> ***"The web-based point of sale (POS) builder system of claim 1, wherein the one or more display interfaces comprise one or more buttons or keys"***

97. As explained for 1[c], Woycik discloses and suggests items associated with buttons/keys (display interfaces) on POS screens that are created/edited via the POS builder interface include items for sale, e.g., "food items" in a restaurant, and customer loyalty discounts/awards (promotions and loyalty points programs). EX1004, [0079], [0082], [0089], [0098], [0100], Figs. 4-9, 11, 20-21:

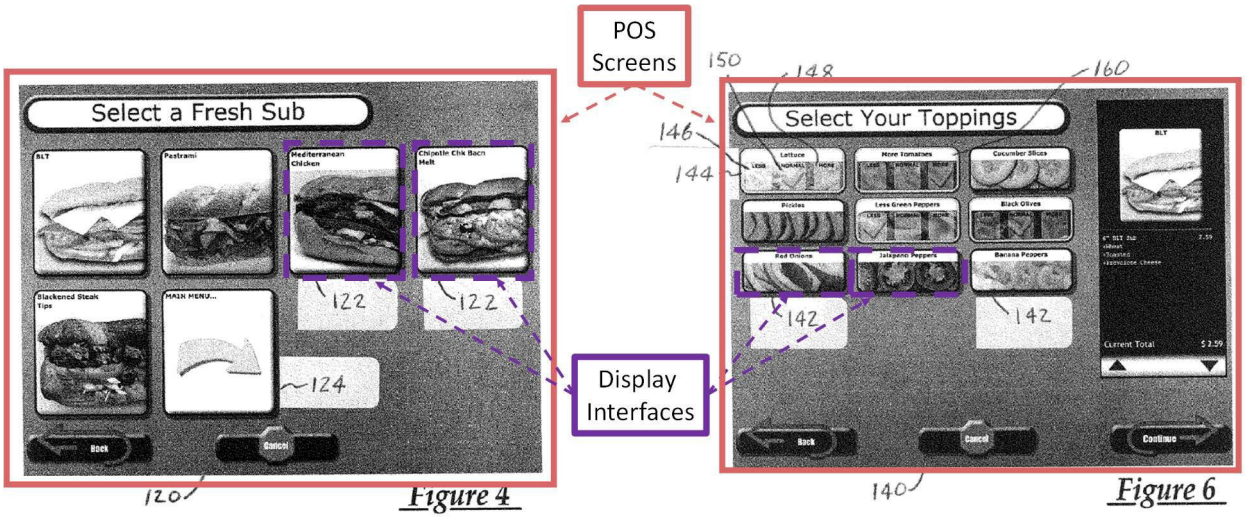


Figure 4 Figure 6

98. Accordingly, it is my opinion that Woycik discloses and suggests the limitations of claims 4 and 9.

E. Claim 7[5]

"The web-based point of sale (POS) builder system of claim 1, wherein the information regarding one or more POS transactions comprises one or more of the employee clock information, the

[5] Claims 7 and 8 recite "the information regarding one or more POS transactions," which refer to "**further** information regarding one or more POS transactions" of 1[d] and not "the information used for creating or modifying the one or more POS screens" of 1[c] because "further information" is the claimed information "regarding one or more POS transactions."

customer add/update information, the item add/update information, or the promotion information"

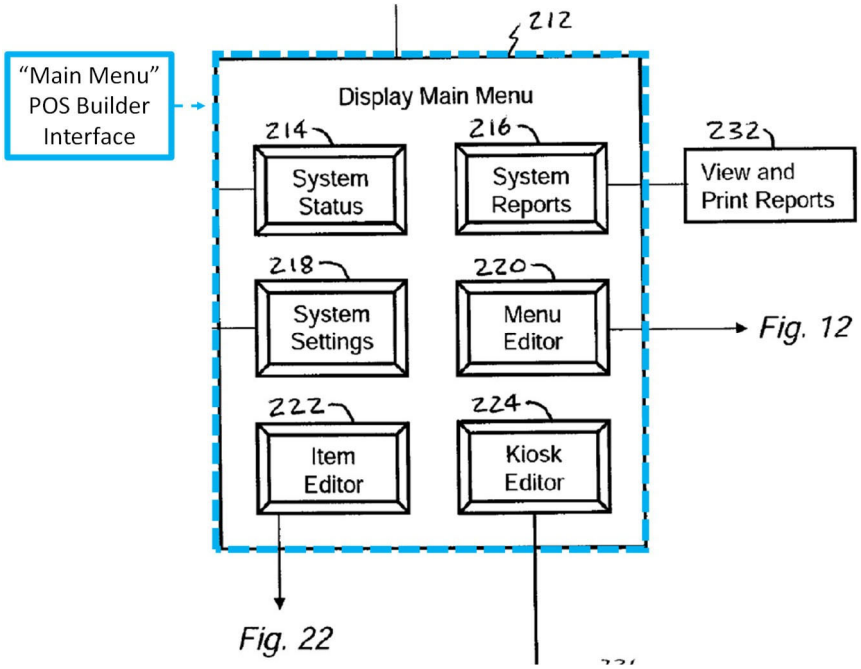
99. As explained for 1[e]-1[f], Woycik discloses central server 22/84 receiving further information from kiosks (POS terminals) regarding orders (POS transactions) that includes item add/update information based on new and/or returning customers adding or updating items in an order. EX1004, [0028], [0089], [0091], [0140]-[0142]. Additionally, Woycik discloses returning customers are recognized by "swiping a customer loyalty card," whereby the system "can provide the customer with a customer loyalty discount or award." EX1004, [0089]. A POSITA would have understood such discount/award would be displayed on a POS screen for selection by the customer, would be based on previous customer/order information stored at central server 22/84, and the POS terminal would transmit "further information" (e.g., information regarding redemption of the discount/award) to central server 22/84, which would update customer information. Thus, the (further) information regarding POS transactions also includes promotion information related to a loyalty program and customer update information.

100. Accordingly, it is my opinion that Woycik discloses and suggests the limitations of claim 7.

F. Claim 8

"The web-based point of sale (POS) builder system of claim 1, wherein the information regarding one or more POS transactions are viewable via the POS builder interface"

101. As explained for 1[c], Woycik's web-based "administrative tool" user interface (POS builder interface) allows "the administrator to perform various administrative functions." EX1004, [0073], [0075]-[0076]. The administrator can use it "to run reports, check the system status, add or remove kiosks from the system, or edit the screen displays that the customers see." *Id.*, [0095]; *see also id.*, [0096]-[0097], [0128], Fig. 10 (excerpted):



102. As explained for 1[d], the further information includes details regarding customer orders (POS transactions), which is stored by central server

22/84 for later use. EX1004, [0028], [0030], [0089], [0091], [0128]. Using the administrative tool interface, the administrator can also "view and print sales reports" for "any individual kiosk" and "time period." *Id.*, [0097]. "Administrators request data or a report from the remote server that then supplies the data or report requested (blocks 598, 600)." *Id.*, [0128], Fig. 28:

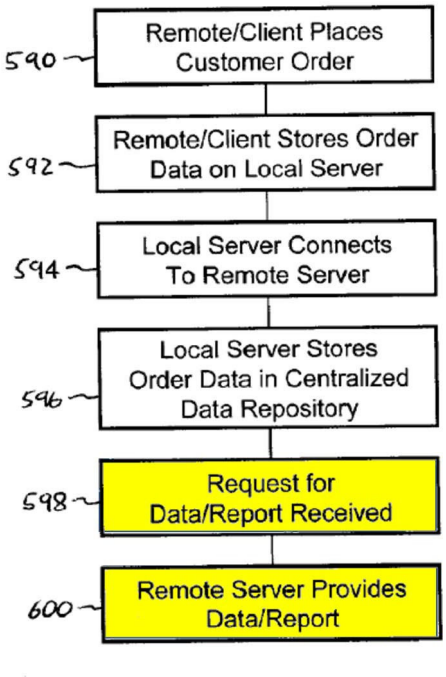


Figure 28

Woycik thus discloses and suggests reports displayed on the administrative tool user interface for a specific kiosk/terminal would include information regarding one or more POS transactions performed at that kiosk/terminal.

103. Accordingly, it is my opinion that Woycik discloses and suggests the limitations of claim 8.

G. Claim 10

"The web-based point of sale (POS) builder system of claim 1, wherein the received information comprises information** indicative **of at least one of a number, shape, or arrangement of the one or more display interfaces"

104. As explained for limitation 1[c], Woycik's administrative tool allows the manager/administrator to define the layout (number, shape, and arrangement) of buttons (display interfaces) on POS screens. EX1004, [0100]. "Buttons can be directly added one at a time by the administrator from an available collection of different buttons when building the screens." *Id.* "Alternatively … templates containing predefined groupings of buttons are used to develop the menu screens," and "define the overall layout of the menu such as the number and types of buttons, and placement of buttons." *Id.*; *see also id.*, [0101]-[0107], Figs. 14-19:

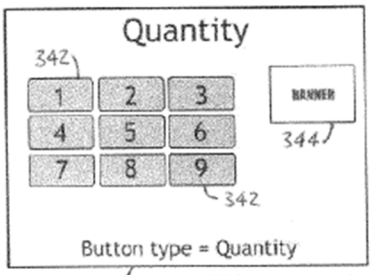


Figure 14

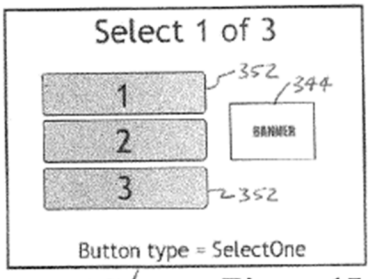


Figure 15

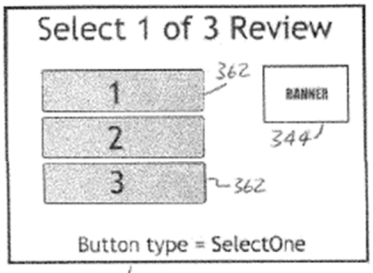


Figure 16

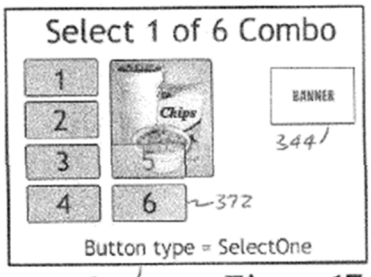


Figure 17

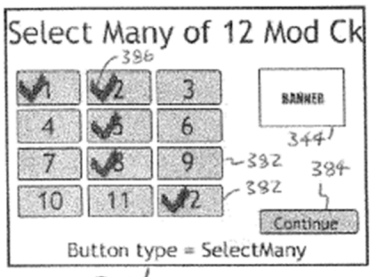


Figure 18

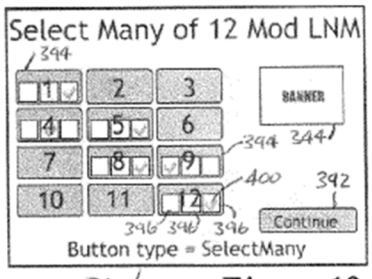


Figure 19

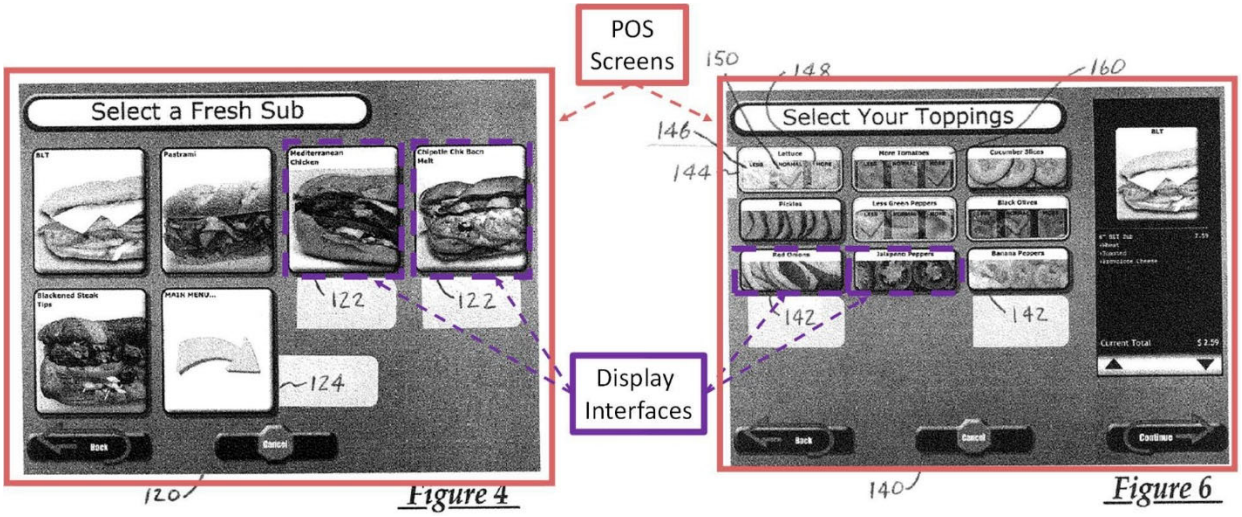
The server receives information from the POS builder interface regarding the "number of buttons a customer sees on the menu, what happens when a customer selects a button … and the overall appearance of the menu" (number, shape, and arrangement of buttons/keys and other display interfaces). *Id.*, [0100].

105. Accordingly, it is my opinion that Woycik discloses and suggests the limitations of claim 10.

H. Claim 12

"The web-based point of sale (POS) builder system of claim 1, wherein the one or more display interfaces are accessible on the POS builder interface"

106. As explained for claims 1[c] and 9-10, Woycik discloses buttons/keys on the POS screens (display interfaces). EX1004, [0079], [0082], [0098], [0100], Figs. 4-9:



Those buttons are added/edited using the administrative tool user interface (POS builder interface). *Id.*, [0098], [0100], [0108], [0112], Fig. 12. For example, the buttons may be created individually or from a template. [0100]-[0107], Figs. 14-19:

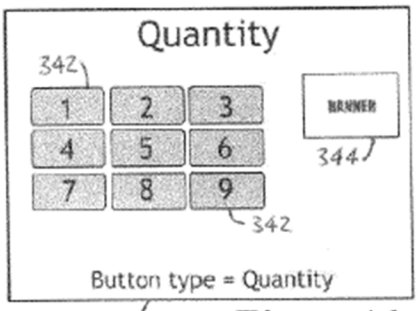


Figure 14

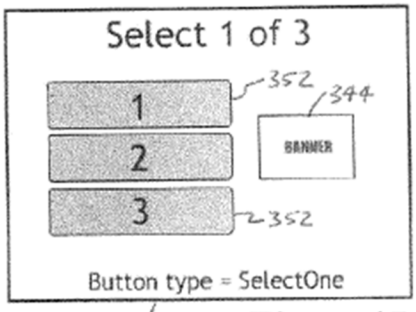


Figure 15

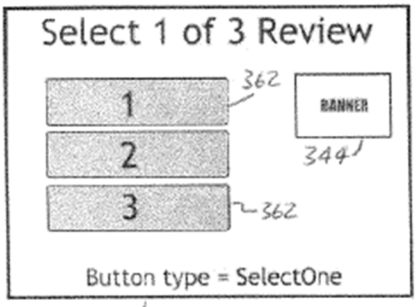


Figure 16

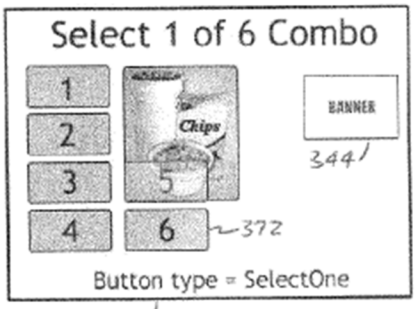


Figure 17

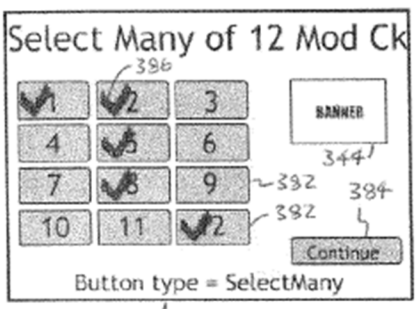


Figure 18

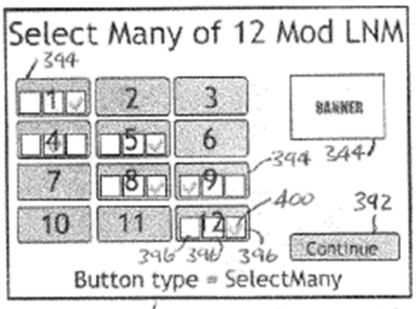


Figure 19

Woycik thus discloses and suggests the buttons (display interfaces) are accessible on the POS builder interface.

107. Further, Woycik suggests administrators can preview POS screens to accept or reject changes. EX1004, [0112]. A POSITA would have been motivated to allow administrators to access POS screens and buttons (display interfaces) to preview modified screens/buttons before transmitting changes to POS terminals to

ensure accuracy and to test functionality of the updated POS screens. Specifically, a POSITA would have understood the benefit of previewing POS screens by building confidence in a non-technical user that his/her edits resulted in the expected POS screen changes. In addition, the administrative user would be able to immediately observe and correct any errors, for example, typographical or formatting errors. Users in the 2008 timeframe were used to preview (e.g., print preview) and WYSIWYG (what you see is what you get) and would have expected a real-time preview from a modern POS system. A POSITA would have known how to implement the preview process by incorporating the POS screens into the POS builder interface (likely in a portion of the POS builder interface). And a POSITA would have been motivated to incorporate this functionality to achieve these benefits with the modest development effort required to support this functionality.

108. Accordingly, it is my opinion that Woycik discloses and suggests the limitations of claim 12.

I. Claim 13

"The web-based point of sale (POS) builder system of claim 1, wherein the at least one server is further configured to: receive, over the network from the POS builder interface, second information regarding a modification to at least one of the one or more POS screens; and update the at least one of the one or more POS screens

on the one or more POS terminals based on the second information"

109. As explained for 1[c], Woycik's web-based administrative tool interface (POS builder interface) allows administrators to "edit the screen displays that the customers see." EX1004, [0095], [0073]. "The individual items or products available for sale are edited in the item editor" where the administrator can "configure the various food items and options" including modifying pricing information, button text, and enabling/disabling buttons. *Id.*, [0112]; *see also id.*, [0110]-[0113], Figs. 20-22:

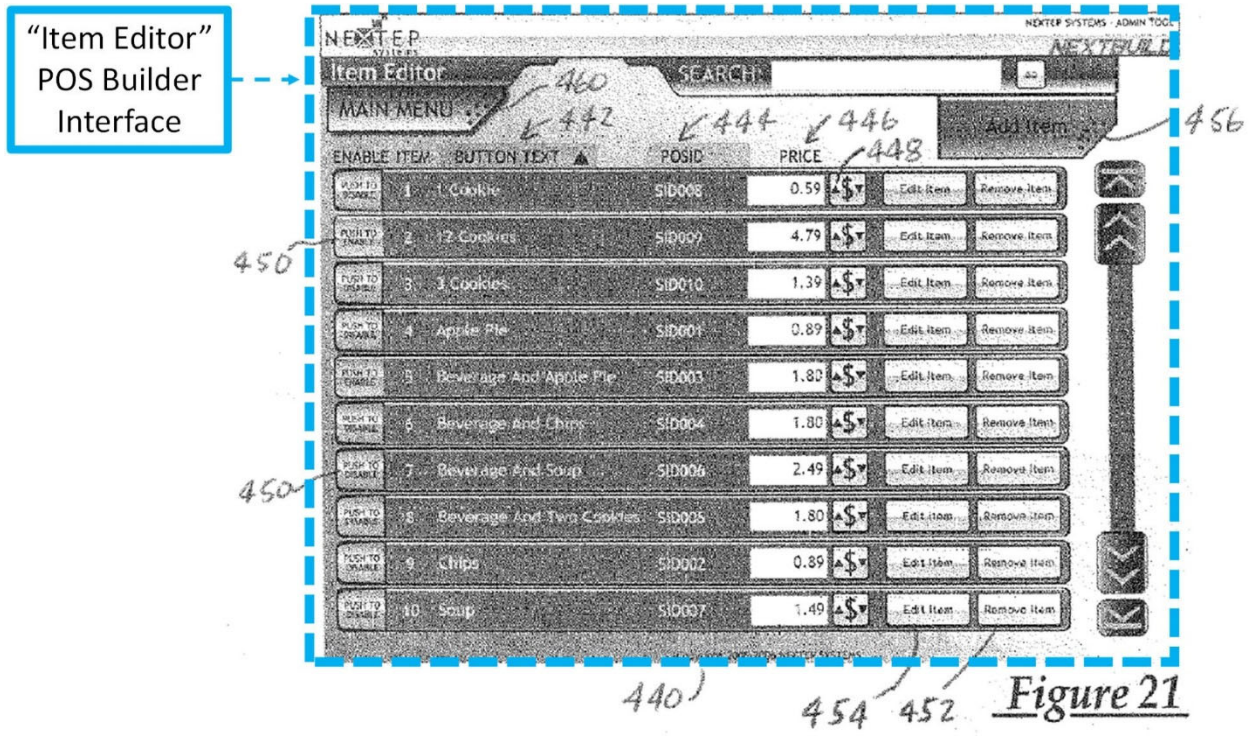
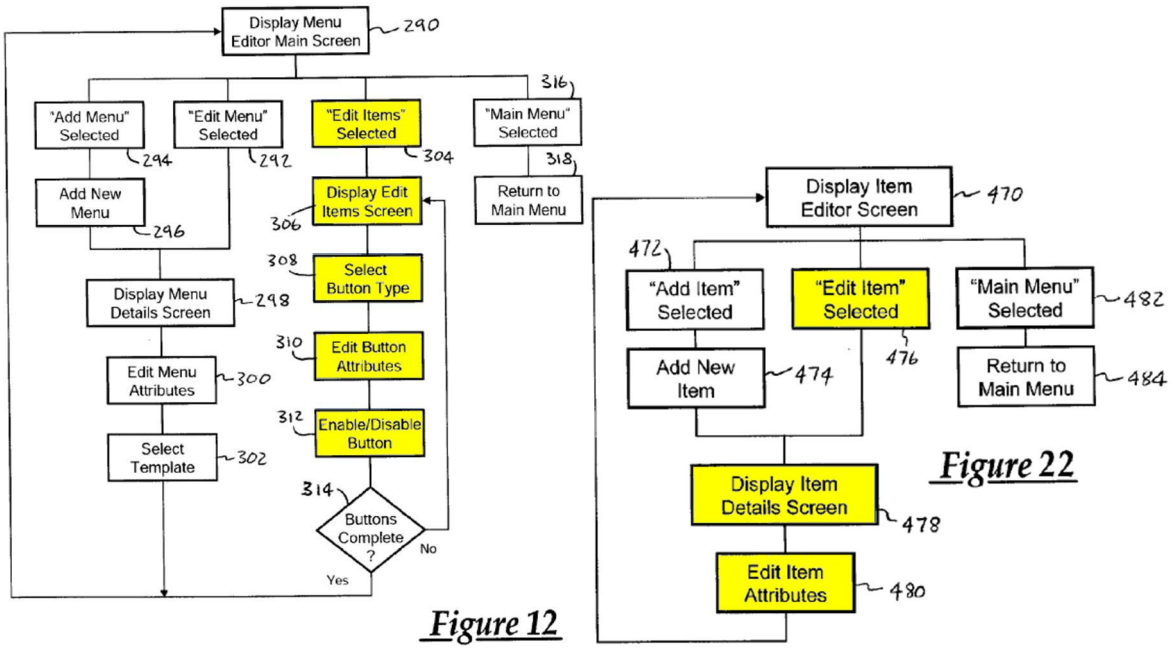


Figure 21

When "editing an existing item, the administrative tool launches a separate items detail screen with multiple tabs to edit various item attributes." *Id.*, [0112]; *see also id.*, [0108], [0113], Figs. 12, 22:



Information received by central server 22/84 from the administrative tool interface to modify the attributes of *existing* items/buttons is "second information" (EX1001, Claim 13) regarding a modification to a POS screen that is different than the information used to create the items/buttons originally. The server can create/modify POS screens iteratively, and would receive second, third, fourth…nth information regarding POS screen modifications used to update screens on POS terminals.

110. "The administrative tool saves all changes to the server after the administrator has made all desired changes to the customer interface using the administrative tool." EX1004, [0115]. Those modified "menu configurations" are then sent to the POS terminals/kiosks to update POS screens based on the second information regarding POS screen modification. *Id.*, [0120]-[0122]; *see also id.*, [0071], [0075]-[0076], [0095], [0117].

111. Accordingly, it is my opinion that Woycik discloses and suggests the limitations of claim 13.

J. Claim 14

"The web-based point of sale (POS) builder system of claim 1, wherein the at least one server is further configured to store information regarding the one or more POS screens"

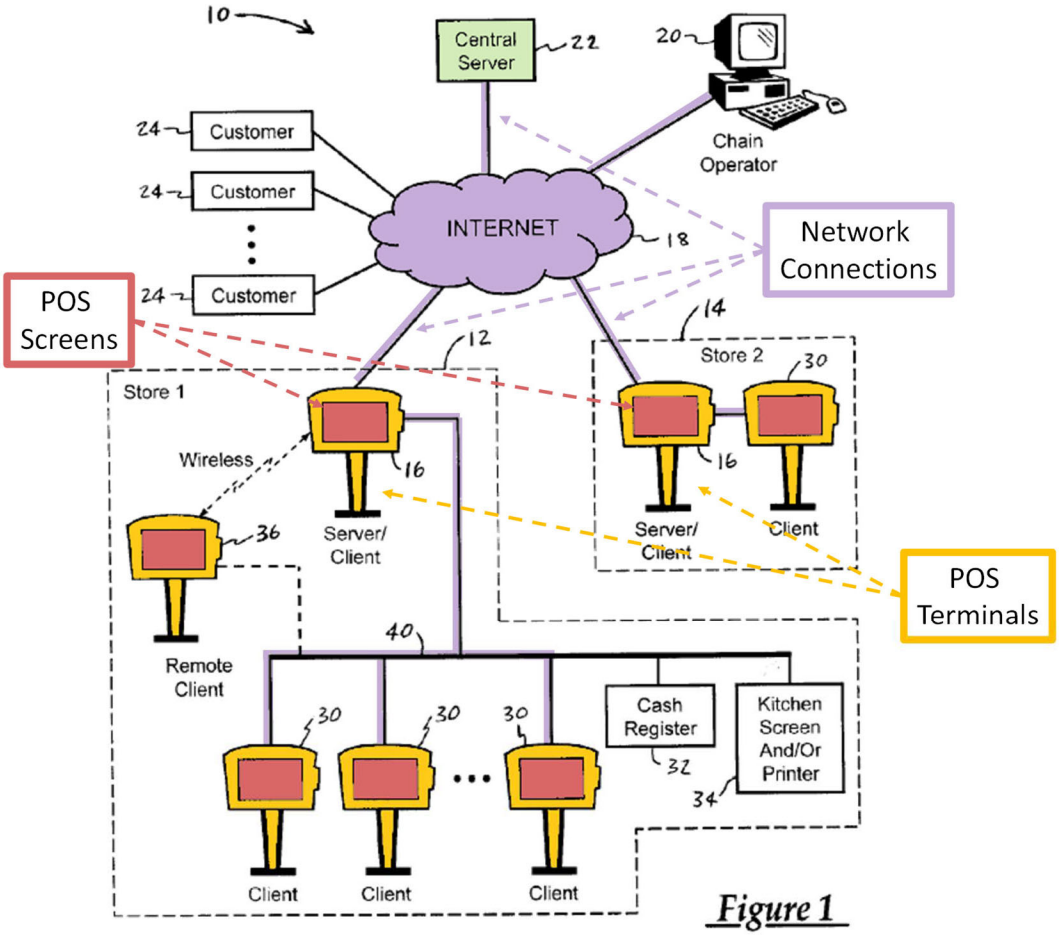
112. As explained for claim 13, Woycik's "administrative tool saves all changes to the server after the administrator has made all desired changes to the customer interface using the administrative tool." EX1004, [0115]. That server can be central server 22/84. *Id.*, [0121]-[0122]; *see also id.*, [0071], [0075]-[0076], [0095].

113. Accordingly, it is my opinion that Woycik discloses and suggests the limitations of claim 14.

K. Claim 15

"The web-based point of sale (POS) builder system of claim 1, wherein the at least one server is located remotely from the one or more POS terminals"

114. As explained for 1[a]-1[b], central server 22/84 is remote from the POS terminals/kiosks. EX1004, [0071], [0075]-[0076], Fig. 1:



115. Accordingly, it is my opinion that Woycik discloses and suggests the limitations of claim 15.

L. Claim 16

> ***"The web-based point of sale (POS) builder system of claim 1, wherein the at least one server is further configured to receive the information for creating or modifying the one or more POS screens and create or modify the one or more POS screens in real time while the one or more POS terminals are in use performing one or more POS transactions"***

116. As explained for 1[c], Woycik's central server receives from a POS builder interface information for creating/modifying interactive menu screens (POS screens). EX1004, [0013], [0016], [0019]-[0020], [0073], [0075], [0079], [0098]-[0099], [0109]-[0110], Figs. 11-13. Configuration changes (e.g., menu/item changes) made over the Internet with the administrative tool at central server 22/84 are saved to the central server, then pushed to kiosks at local stores. *Id.*, [0115], [0076], [0122]. Changes are saved to the central server when the administrator enters them and clicks the "OK button" at "block 314 … when button editing is completed to return to the menu editor main screen." *Id.*, [0108], [0111], Figs. 12, 13, 20:

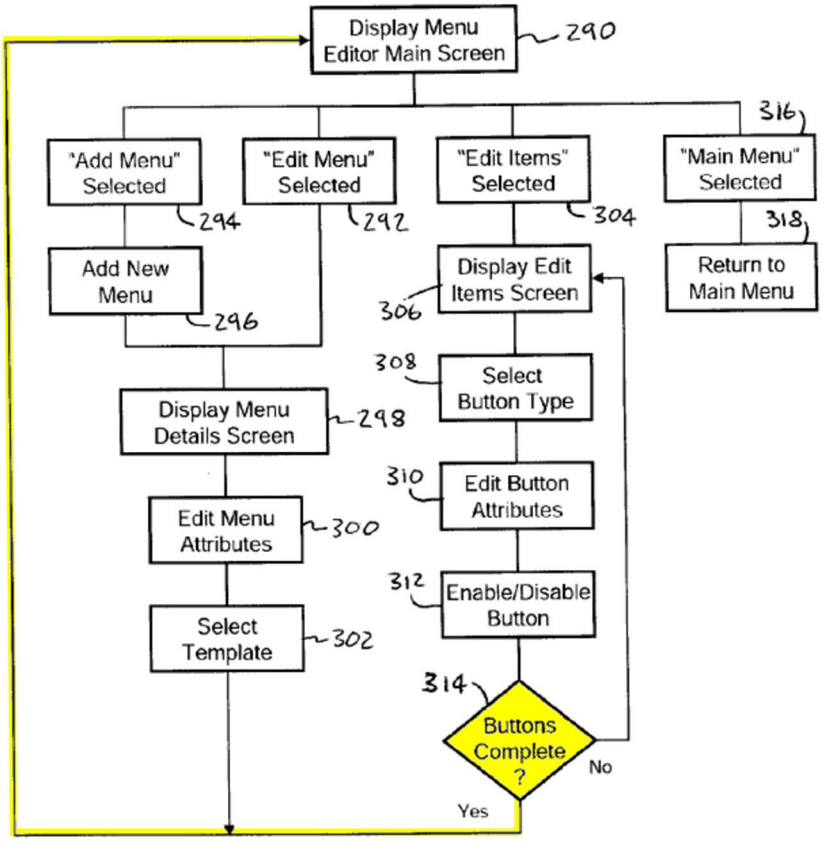


Figure 12

The central "server can automatically send the changes to the local server" kiosk 16/82 via Internet 18. *Id.*, [0122]; *see also id.*, [0071], [0072], [0115]. Woycik teaches central server 22/84 communicates with in-store kiosks/terminals 16/82 "in real-time throughout the day." *Id.*, [0030]. Woycik thus discloses and suggests receiving information for and creating/modifying POS screens stored on the central server in real time. Woycik further discloses "the kiosk's status (idle, in use, etc.)" is determined by the system, and that changes made and saved on the server can be pushed down to the kiosks when they are idle. EX1004, [0097], [0015]. That

suggests, and it would be obvious, that information can be received in real time, and changes to POS screens can be made and saved at central server 22/84 in real time, while kiosks (POS terminals) are in use by customers placing orders.

117. Additionally, as explained for claim 12, Woycik suggests administrators can preview POS screens and accept or reject changes. EX1004, [0112]. This suggests POS screens are also created/modified at the server in real time while in-store POS terminals are in use, e.g., to facilitate the preview functionality.

118. Accordingly, it is my opinion that Woycik discloses and suggests the limitations of claim 16.

M. Claim 17

"The web-based point of sale (POS) builder system of claim 1, wherein the one or more POS terminals use the one or more POS screens after completing a pending POS transaction"

119. Woycik discloses that a "kiosk's status (idle, in use, etc.)" may be determined by the system. EX1004, [0097]. Further, updates to the POS screens may be made "during idle time when the kiosk is not in use." *Id.*, [0115], [0117]. Woycik thus discloses and suggests waiting until after a POS terminal's pending transaction is complete to update POS screens during idle time. This suggests and renders obvious a POS terminal's continued use of existing POS screens for a

pending transaction before updating screens during subsequent idle time—after which the kiosk uses the updated screens. Additionally, a POSITA would have understood the kiosks reuse POS screens for each new transaction after a prior transaction is complete, thus using/re-using updated POS screens after completing each transaction.

120. Accordingly, it is my opinion that Woycik discloses, suggests, and renders obvious the limitations of claim 17.

N. Claims 18

"The web-based point of sale (POS) builder system of claim 1, wherein the at least one server is further configured to maintain information regarding POS screens for separate sets of POS terminals separately"

and 20

"The web-based point of sale (POS) builder system of claim 1, wherein the at least one server is further configured to maintain information regarding the one or more POS screens"

121. As explained for claim 16, configuration changes (e.g., menu and item changes) made over the Internet with the administrative tool at central server 22/84 are saved to (thus maintained by) the central server, then pushed to kiosks at local stores. *Id.*, [0115], [0076], [0122]. Woycik teaches that, in some situations, "not all store locations are intended to use the same menus or pricing," and "the changes can be sent to less than all stores." *Id.*, [0122]. In that scenario, a POSITA would

have understood, and be motivated to ensure, that the central server separately stores/maintains different sets of interactive menus (POS screens) used by POS terminals at those different store locations to ensure communication of correct menus/pricing. For example, a store location in an airport terminal or a sports venue often has higher prices and fewer menu items than a regular store location. A POSITA would have understood that a user would be confused to see a mix of pricing and menu items that did not match his/her current location. Thus, a POSITA would have been motivated to segregate and separately maintain data for different locations/POS terminals.

122. Accordingly, it is my opinion that Woycik discloses and suggests the limitations of claim 18.

O. Claim 19

"The web-based point of sale (POS) builder system of claim 1, wherein instructions to the POS builder interface for programmatic creation and modification of the POS terminals are not formatted in programming code"

123. The administrative tool provides "a simplified user interface that requires little if any training or experience with computers." EX1004, [0080]. It "provides a graphical representation of menu options, button selections, images, text fields, etc. on the kiosk." *Id.*, [0095]. "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." *Id.* Further, it "eliminates the need for administrators to write their own code to develop the layout and behavior of the menus." *Id.*, [0100]. Woycik thus discloses and suggests instructions to the administrative tool GUI (POS builder interface) for the creation/modification of POS screens (and thus creation/modification of POS terminals), are not formatted in programming code.

124. Accordingly, it is my opinion that Woycik discloses and suggests the limitations of claim 19.

P. Claim 21

"The web-based point of sale (POS) builder system of claim 1, wherein the received information comprises one or more attributes of the one or more items"

125. As discussed for 1[c], Woycik discloses the central server receives information for creating/modifying buttons associated with items (including food items) and their attributes. Woycik's administrative tool "include[s] the same functions typically used to add, delete, and configure kiosks in the system, to make global settings (e.g., set time, sales tax percentage, receipt header and footer text), to create display screens (e.g., food ordering menus), to create a list of food items that are used on the different menu screens, and to **edit the various attributes of the food items (e.g., name, associated graphic, price)**." EX1004, [0079]. "The

individual items or products available for sale are edited in the item editor" where the administrator can "configure the various food items and options." *Id.*, [0112]; *see also id.*, [0110]-[0113], Figs. 20-22:

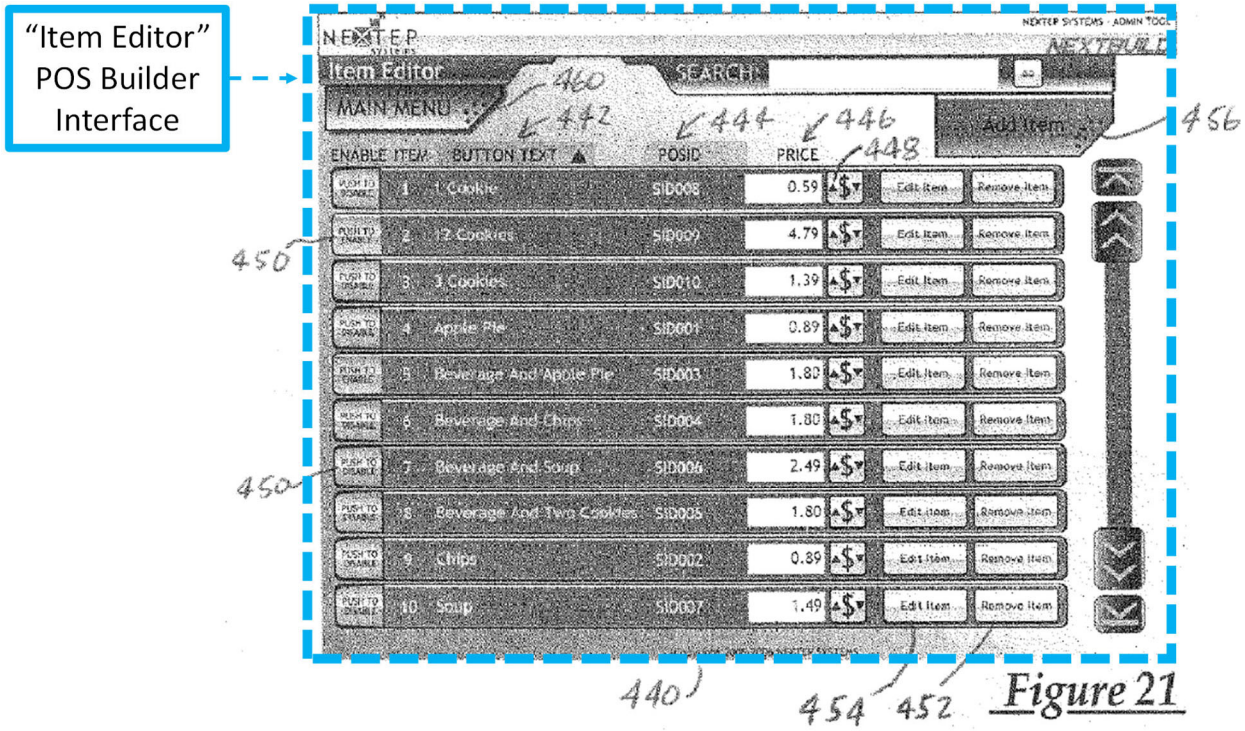


Figure 21

126. Accordingly, it is my opinion that *Woycik* discloses and suggests the limitations of claim 21.

Q. Claim 22

"The web-based point of sale (POS) builder system of claim 1, wherein the POS terminals are configured to perform** transactions **independently of a connection with the network"

127. Woycik discloses that generally "a single store location will include at least one" local server kiosk 16/82 (POS terminal), and that some store locations "can have a simple one … kiosk" setup and thus would have one network

connection between kiosk 16/82 and central server 22/84. EX1004, [0071], [0074]. Woycik teaches "[t]he local server [kiosk 16/82] connects to the remote server [central server 22/84] periodically and stores the customer and order data in the centralized data repository." *Id.*, [0128]. That suggests, and it would be obvious, that some of the time a store with a single POS terminal (kiosk 16/82) operates without an Internet 18 connection to the central server but is configured to perform transactions independently of that connection, store customer/order data locally, and then later reconnect to transmit the data to the central server. Further, it suggests that stores may have multiple local server kiosks 16/82 that would operate the same way.

128. Accordingly, it is my opinion that Woycik discloses and suggests the limitations of claim 22.

R. Claim 23

"The web-based point of sale (POS) builder system of claim 1, wherein the input interface element comprises a data interface for inputting at least some of said further information"

129. In Woycik, "[t]he primary components of each menu screen are **buttons** that allow the customer to make selections among available items and options." EX1004, [0100]. As explained in 1[c], those buttons are display interfaces associated with items. *Id.*, [0098], [0100], [0112], Figs. 4-9:

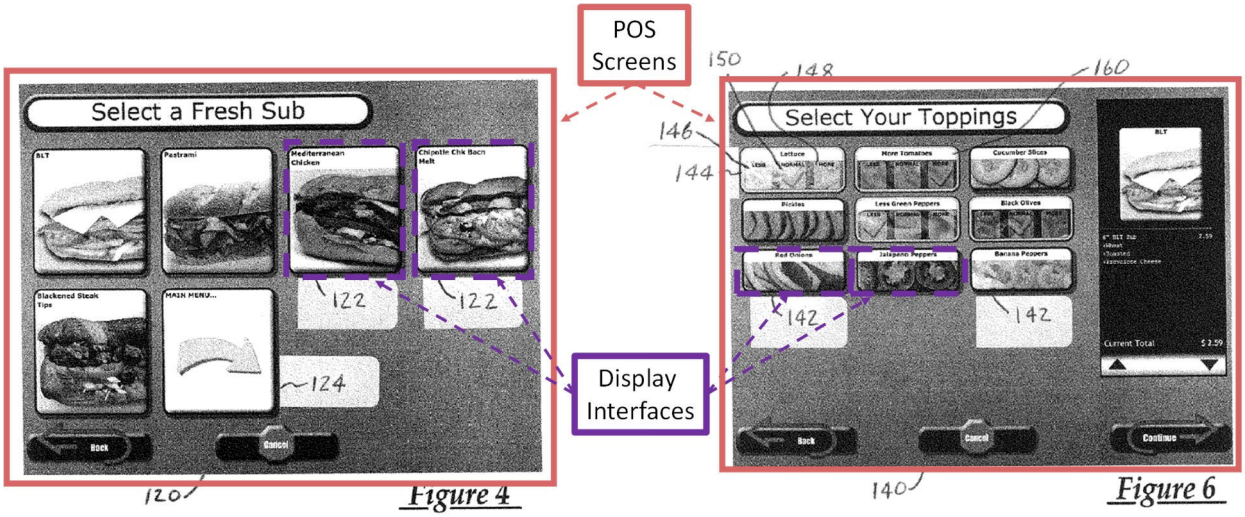


Figure 4 Figure 6

130. As explained in 1[d] and 1[g], the further information includes information regarding items ordered by a customer. Customers input that information using buttons (display interfaces) on the interactive menu (POS screen) at the POS terminal "to select and customize the desired food items, specify quantity, options, etc." *Id.*, [0073]. Woycik discloses multiple types of buttons, including "quantity buttons (Quantity), less/normal/more buttons (LNM), and none/normal/extra buttons (NNE)." *Id.*, [0083]; *see also id.*, [0084]-[0088], Figs. 6-9:

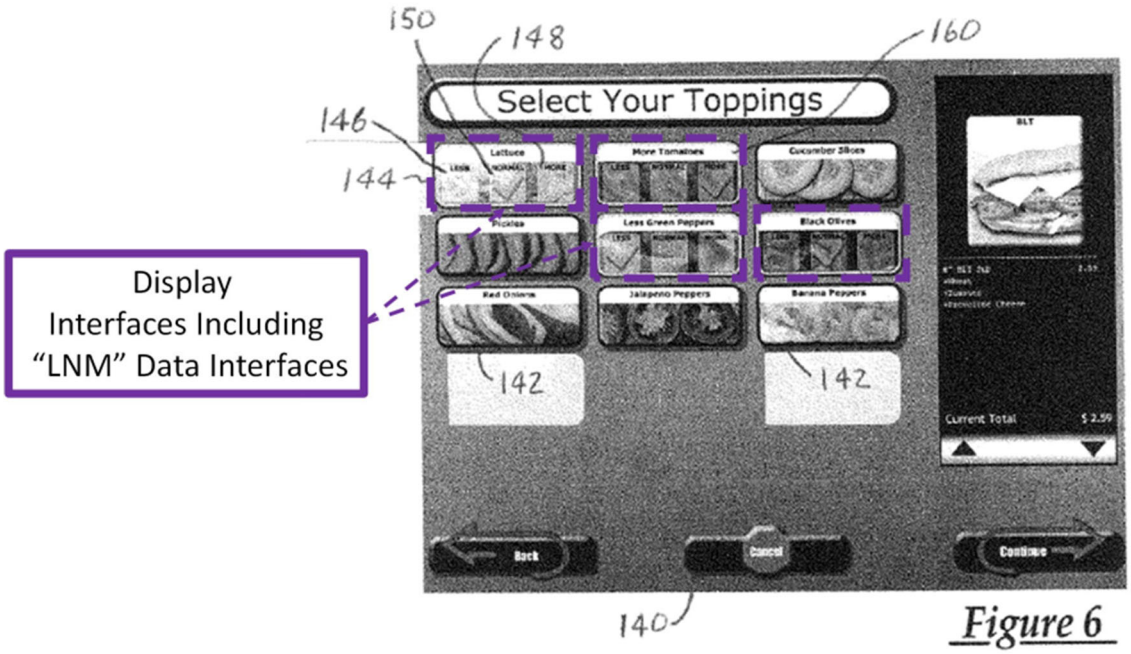


Figure 6

131. Such buttons thus are or include a data interface for inputting at least some of the further information (data) regarding an order/transaction, e.g., items and options selected (item add/update information). An additional use case in the fast-food context is whether to make a sandwich into a meal by adding a drink and a side. In the retail electronics context, a common use case is to offer an extended warranty, compatible accessories, or upgrades. Additionally, a POSITA would have been motivated to include a display interface for text entry, e.g., to allow a customer to make special requests for further item customization. Users often have special requests that can easily be accommodated by the store but may not be anticipated by the POS administrator. Further, some special requests may be infrequently used, thus, a POS administrator would not want to build these

infrequent selections into a menu and instead would be motivated to simplify the screen design and include a text entry.

132. Accordingly, it is my opinion that Woycik discloses and suggests the limitations of claim 23.

S. Claim 24

"The web-based point of sale (POS) builder system of claim 1, wherein to configure the one or more POS terminals comprises dynamically configuring the one or more POS terminals specific for the corresponding customer based on the one or more transactions by the corresponding customer"

133. As explained for 1[c], Woycik discloses several examples of dynamically configuring kiosks to display POS screens specific to individual customers based on "further information" received from the kiosks regarding POS transactions by the customer. The system may recognize returning customers using "a customer loyalty card" or "a credit card" and create/modify the POS screens displayed based on such further information. *Id.*, [0089]; *see also id.*, [0028], [0091], [0140]-[0142]. For example, "FIG. 40 illustrates a customer recognition screen 860 displayed by the self-order application when a returning customer is recognized." *Id.*, [0142]:

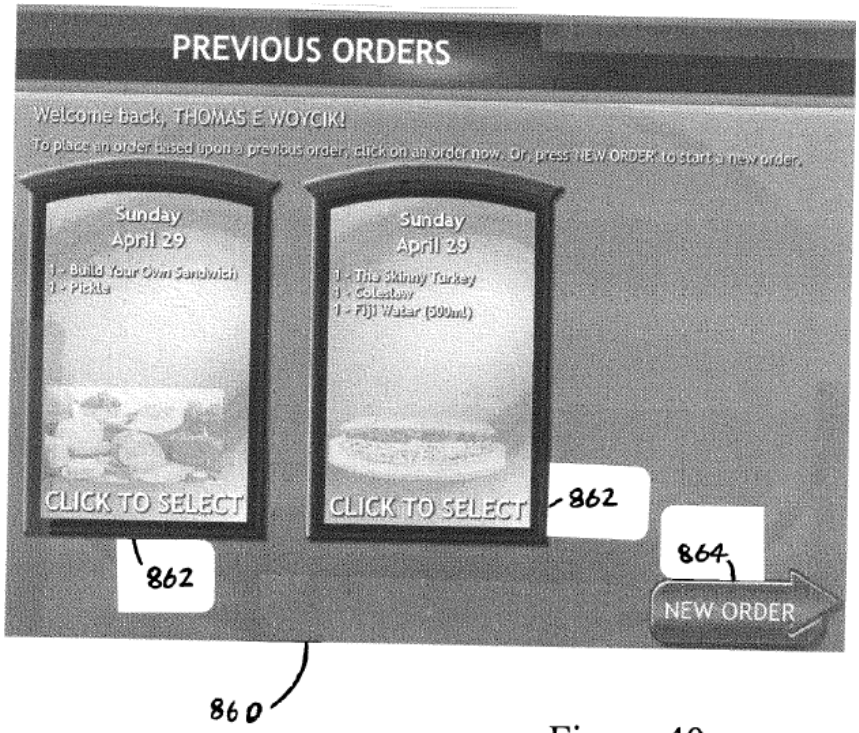


Figure 40

"The customer may choose to duplicate an entire previous order or individual items from that order," e.g., he "can simply select an item to customize (if desired) and add to his current order." *Id.*, [0089]. Further, "the system includes some button types … that have dynamically generated content based on the customer's order." *Id.*, [0103].

134. Accordingly, it is my opinion that Woycik discloses and suggests the limitations of claim 24.

T. Claims 25

"The web-based point of sale (POS) builder system of claim 1, *wherein* ***the POS builder interface is configured to create or modify at least one of: a position or operation of a first display interface of***

the one or more display interfaces, wherein the first display interface comprises an input interface element"

and 26

"The web-based point of sale (POS) builder system of claim 25, wherein the input interface element comprises a touch screen input interface element"

135. As explained in 1[c] and claim 10, Woycik discloses the administrative tool GUI (POS builder interface) is configured to create/modify the layout of the POS screens, including the position and operation of buttons (display interfaces). Buttons can be added individually or from a template that "define[s] the overall layout of the menu such as the number and types of buttons, and **placement of buttons**." EX1004, [0100]; *see also id.*, [0101]-[0107], Figs. 14-19:

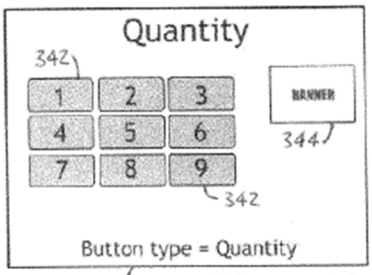


Figure 14

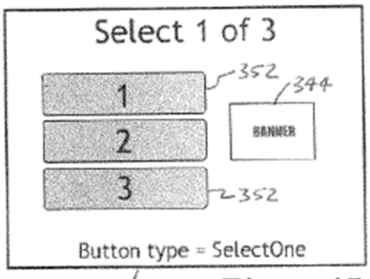


Figure 15

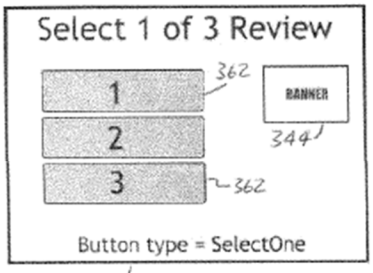


Figure 16

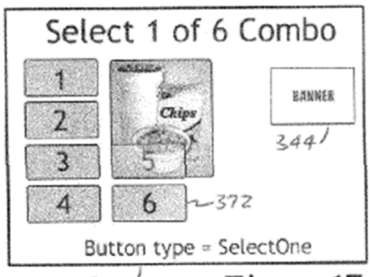


Figure 17

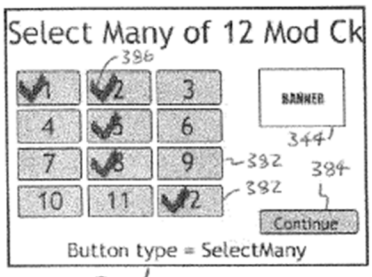


Figure 18

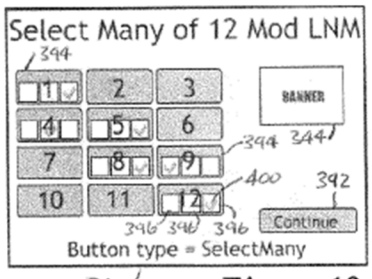


Figure 19

Further, the button operation or "behavior (e.g., how the system responds to the button being pressed)" can be configured. *Id.*, [0102]; *see also id.*, [0100], [0108].

136. As explained for claim 23, Woycik discloses buttons (display interfaces) that are themselves and/or include input interface elements such as "LNM" data interfaces. *Id.*, [0073], [0083]-[0088], Figs. 6-9:

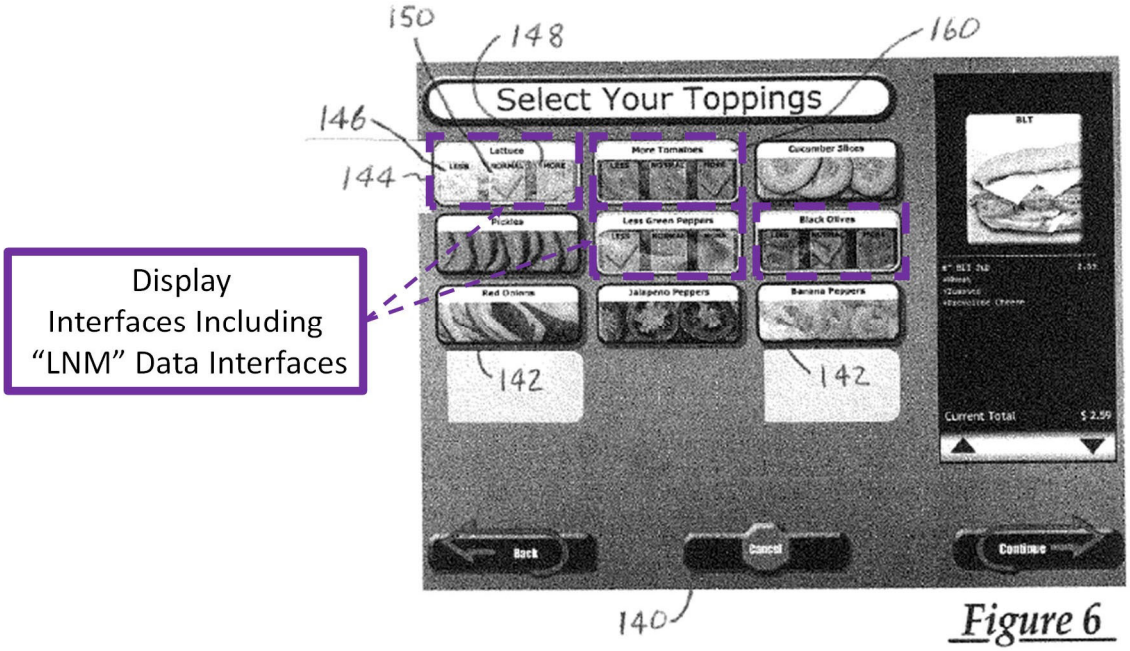


Figure 6

Woycik's POS terminals include "a display screen having a touch screen for accepting user input." *Id.*, [0014], *see also id.*, [0077]-[0078], [0089]. The buttons thus include touch screen input interface elements.

137. Accordingly, it is my opinion that Woycik discloses and suggests the limitations of claim 26.

U. Claim 27

138. Claim 27 is similar to claim 1, but is drafted from the perspective of a POS terminal rather than a server. Similar elements are obvious for the same reasons as in claim 1 and as explained below.

1. 27[pre]

"A web-based point of sale (POS) builder system comprising"

139. *See* 1[pre], § XII.A.1.

140. Accordingly, it is my opinion that Woycik discloses and suggests the limitations of claim 27[pre].

2. 27[a]

"at least one POS terminal configured to: display one or more POS screens"

and 27[b]

"communicate with at ***least one server over a network comprising the Internet"***

141. *See* 1[a]-1[b], § XII.A.2. *See also* EX1004, [0071], Fig. 1:

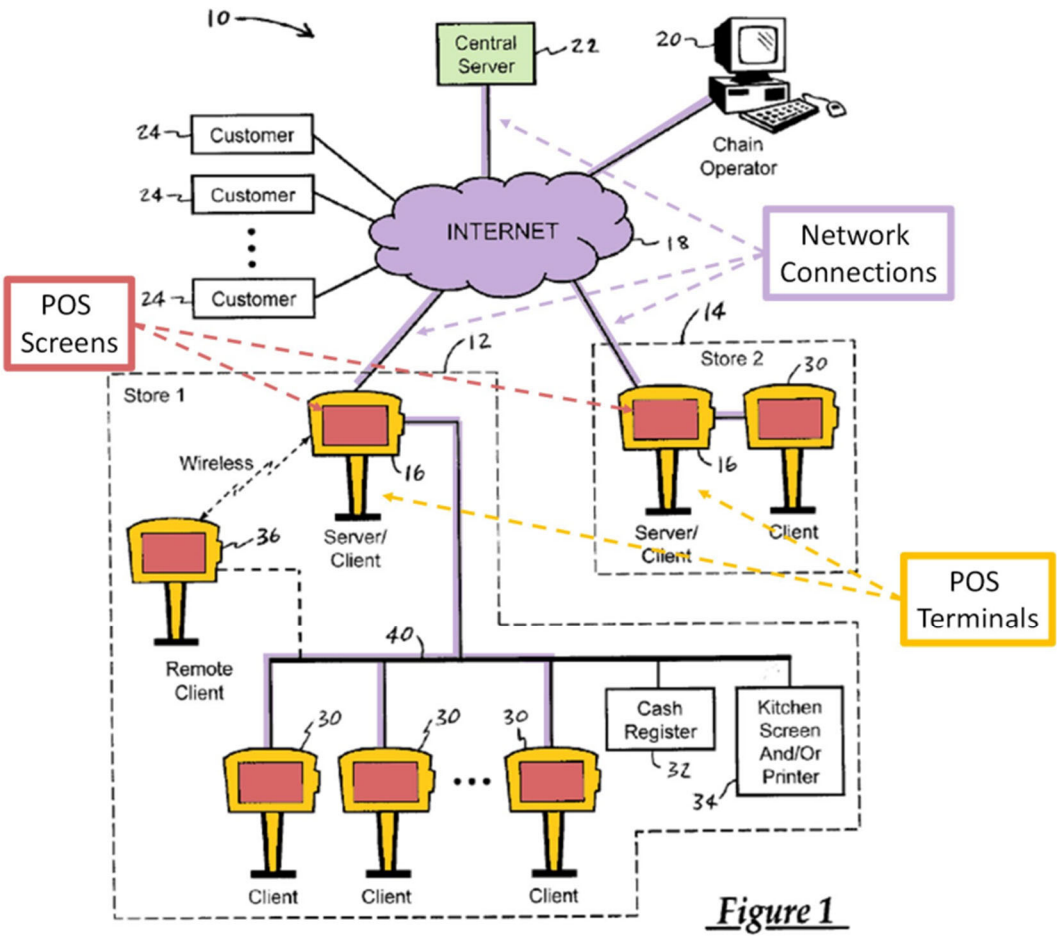


Figure 1

142. Accordingly, it is my opinion that Woycik discloses and suggests the limitations of claim 27[a] and 27[b].

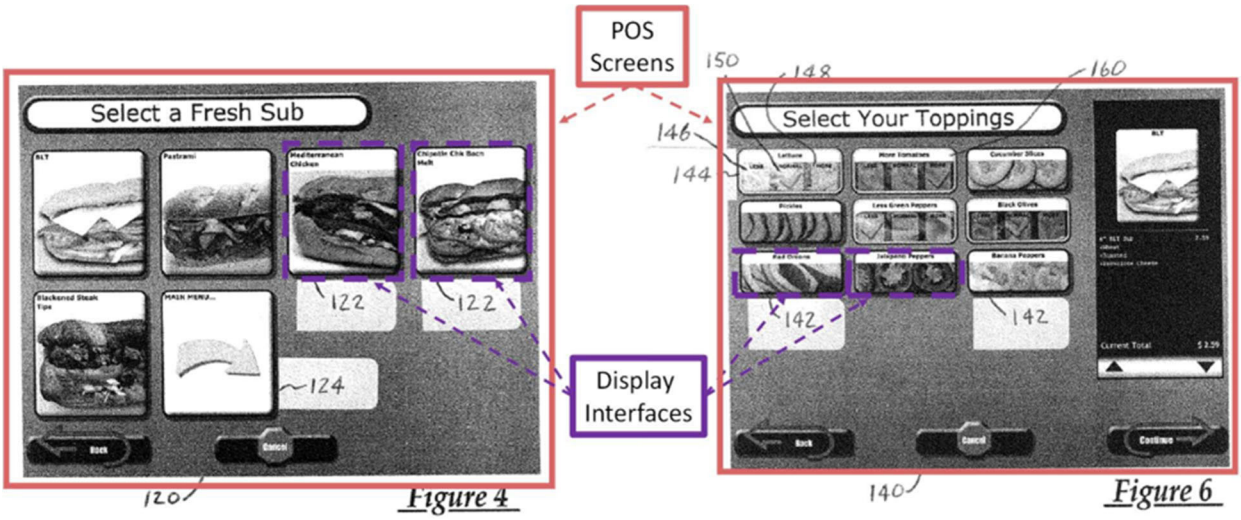
3. 27[c]

"receive, over the network from the at least one server, 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"

143. This limitation is similar to 1[c], but it recites a POS terminal receiving information from the server rather than a server receiving information

from a POS builder interface. That difference is addressed below. The remaining limitations of 27[c] are obvious for the same reasons explained for 1[c].

144. As explained in 1[c], central server 22/84 receives from the POS builder interface information used for creating/modifying interactive menus (POS screens), including creating/modifying buttons (display interfaces) on POS screens associated with food items. EX1004, Abstract, [0013], [0016]-[0017], [0073], [0079], [0100]-[0107], Figs. 4-9:



145. As explained for claim 16, Woycik's configuration changes (e.g., menu/item changes) made over the Internet with the administrative tool at central server 22/84 are saved to the central server, then pushed to POS terminals at local stores. *Id.*, [0115], [0076], [0122]. Kiosks 16/82 (POS terminals) thus receive that information from central server 22/84 over the network comprising the Internet.

146. Accordingly, it is my opinion that Woycik discloses and suggests the limitations of claim 27[c].

4. 27[d]

"perform one or more transactions with respect to the one or more items"

147. Woycik discloses POS terminals are "used directly by the customer to order goods" and complete "the transaction (order, payment, etc.)." EX1004, [0006]; *see also id.*, [0007], [0074], [0089]-[0092]. "Once an order is placed, regardless if at a kiosk or through a web client, the system generates a kitchen build for each item in the order." *Id.*, [0094]; *see also* 1[d], 1[g], § XII.A.4.

148. Accordingly, it is my opinion that Woycik discloses and suggests the limitations of claim 27[d].

5. 27[e]

"transmit, 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"

and 27[h]

"wherein said one or more POS transactions relate to one or more transactions by corresponding customers respectively associated with at least one of said one or more POS terminals"

149. *See* 1[d], 1[g], § XII.A.4.

150. As explained in 1[d], the POS transactions and the further information regarding POS transactions relate to transactions by corresponding customers using respective POS terminals to order items, and the further information is transmitted from the POS terminal and received and stored at the central server. EX1004, [0028], [0030], [0089]-[0092].

151. Accordingly, it is my opinion that Woycik discloses and suggests the limitations of claim 27[e] and 27[h].

6. 27[f]

"create or modify based on the received information or further information the one or more POS screens"

152. *See* 1[e], § XII.A.5.

153. Additionally, 27[f] allows a POS terminal to create or modify POS screens "based on the received information ***or*** further information." EX1001, Claim 27. As explained for 1[e], Woycik discloses POS terminals configured to create and modify POS screens based on information provided by a manager using the administrative tool ***and*** based on further information about POS transactions transmitted from POS terminals to the central server.

154. Accordingly, it is my opinion that Woycik discloses and suggests the limitations of claim 27[f].

7. 27[g]

> ***"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"***

155. *See* 1[f], § XII.A.6.

156. Accordingly, it is my opinion that Woycik discloses and suggests the limitations of claim 27[g].

V. Claims 28, 31-38, 40-41

157. Claims 28, 31-38, and 40-41 recite similar limitations to those depending from claim 1 and are obvious for the same reasons. The following chart identifies the corresponding claims with additional argument/support to address differences.

Limitation	Discussion Incorporated by Reference
Claim 28	Claim 3, § XII.A.C
Claim 31	Claim 9, § XII.A.D
Claim 32	Claim 10, § XII.A.G
Claim 33	Claim 15, § XII.A.K

Limitation	**Discussion Incorporated by Reference**
Claim 34	Claim 16, § XII.A.L Claim 34 is like claim 16 but from the perspective of the POS terminal rather than the server. As explained for claim 16, Woycik's administrative tool can be used to create/modify POS screens on the server in real time while the POS terminal/kiosk is in use by a customer, then the server pushes the changes down to reconfigure the terminal/kiosk to display new/modified POS screens after the transaction is complete.
Claim 35	Claim 17, § XII.A.M
Claim 36	Claim 21, § XII.A.P
Claim 37	Claim 4, § XII.A.D
Claim 38	Claim 22, § XII.A.Q Claim 38 is like claim 22 but adds additional limitations regarding transmitting the information regarding POS transaction to the server after the transaction is complete and the POS terminal establishes a connection to the server. Woycik's central server is also a web server. EX1004, [0076], [0081], [0093]. As explained for claim 22, Woycik's POS terminals/kiosks can perform transactions, store information locally, and then later establish a connection to central server 22 and upload the transaction data periodically. *Id.*, [0128]. That suggests and renders obvious performing the transactions without a connection to the server, e.g., for a store with a single POS terminal.
Claims 40-41	Claim 25-26, § XII.A.T

158. Accordingly, it is my opinion that Woycik discloses and suggests the limitations of claims 28, 31-38, 40-41.

W. Claim 39

"The web-based point of sale (POS) builder system of claim 38, wherein the at least one POS terminal is further configured to store locally the information regarding the one or more transactions"

159. As explained for claims 22 and 38, Woycik's kiosks 16/82 can perform transactions, store transaction data locally, and then periodically establish a connection to central server 22 and upload the transaction data. EX1004, [0128], *see also id.*, [0093].

160. Accordingly, it is my opinion that Woycik discloses and suggests the limitations of claim 39.

X. Claim 42

161. Claim 42's method steps are similar to the functions of claim 1's system, but claim 42 omits the server limitation (hence the numbering discrepancy) and is obvious for the same reasons.

Limitation	Discussion Incorporated by Reference
42[pre]	1[pre], § XII.A.1.
42[a]	1[a]-1[b], § XII.A.2.
42[b]	1[c], § XII.A.3.

Limitation	Discussion Incorporated by Reference
42[c], 42[f]	1[d], 1[g], § XII.A.4.
42[d][6]	1[e], § XII.A.5.
42[e]	1[f], § XII.A.6.

162. Accordingly, it is my opinion that Woycik discloses and suggests the limitations of claim 42.

Y. Claim 43

163. Claim 43's method steps are nearly identical to the functions of claim 27's system. The following chart identifies the corresponding limitations of claim 27, which in turn reference claim 1 for several limitations (parallel references to claim 1 are provide for convenience).

Limitation	Discussion Incorporated by Reference
43[pre]	27[pre], § XII.U.1 (1[pre], § XII.A.1)
43[a]-43[b]	27[a]-27[b], § XII.U.2 (1[a]-1[b], § XII.A.2)
43[c]	27[c], § XII.U.3 (1[c], § XII.A.3)
43[d]	27[d], § XII.U.4

[6] Limitation 42[d] replaces "configure" with "provisioning," a broad term without meaningful distinction here.

Limitation	Discussion Incorporated by Reference
43[e], 43[h]	27[e], 27[h], § XII.U.5 (1[d], 1[g], § XII.A.4)
43[f]	27[f], XII.U.6 (1[e], XII.A.5) As explained for 1[e] and 27[f], *Woycik* discloses POS terminals configured to create and modify POS screens based on information provided by a manager using the administrative tool ("received information" of 43[f]) ***and*** based on further information about POS transactions transmitted from POS terminals to the central server ("transmitted information" of 43[e]).
43[g]	27[g], XII.U.7 (1[f], XII.A.6)

164. Accordingly, it is my opinion that Woycik discloses and suggests the limitations of claim 43.

Z. Claim 44

165. Claim 44 recites similar limitations to claim 1, but more broadly claims "creating or modifying *functionality of the one or POS terminals*" (44[c]). Woycik's disclosure regarding creating and modifying a series of interactive POS screens and buttons displayed on POS terminals/kiosks (including button behavior/functionality) renders obvious the broader limitations regarding functionality for the same reasons as explained for claim 1. Additionally, Woycik's button functionality or "behavior (e.g., how the system responds to the button

being pressed)" can be configured. EX1004, [0102], [0100], [0108]. The following chart identifies the corresponding limitations.

Limitation	Discussion Incorporated by Reference
44[pre]	1[pre], § XII.A.1.
44[a]-44[b]	1[a]-1[b], § XII.A.2.
44[c]	1[c], § XII.A.3.
44[d], 44[g]	1[d], 1[g], § XII.A.4.
44[e]	1[e], § XII.A.5.
44[f]	1[f], § XII.A.6.

166. Accordingly, it is my opinion that Woycik discloses and suggests the limitations of claim 44.

XII. GROUND 2: TENGLER IN VIEW OF THE KNOWLEDGE OF A PERSON OF ORDINARY SKILL IN THE ART RENDERS CLAIMS 1-4, 7-28, AND 31-44 OBVIOUS

A. Claim 1

1. 1[pre]

"A web-based point of sale (POS) builder system comprising"

167. Tengler discloses an order processing/management system for use in restaurants/stores. EX1005, [0002]-[0006], [0024], [0026]. It includes a server and point-of-sale "stations enabling users to enter orders" and "a network

interconnecting the stations," which may be "a Transmission Control Protocol/Internet Protocol (TCP/IP) network" (e.g., the Internet). *Id.*, [0011], [0078]. Cashiers may use "order terminals 52[] to enter orders" and customers may use "self-service kiosks 74." *Id.*, [0062], [0074]; Fig. 2:

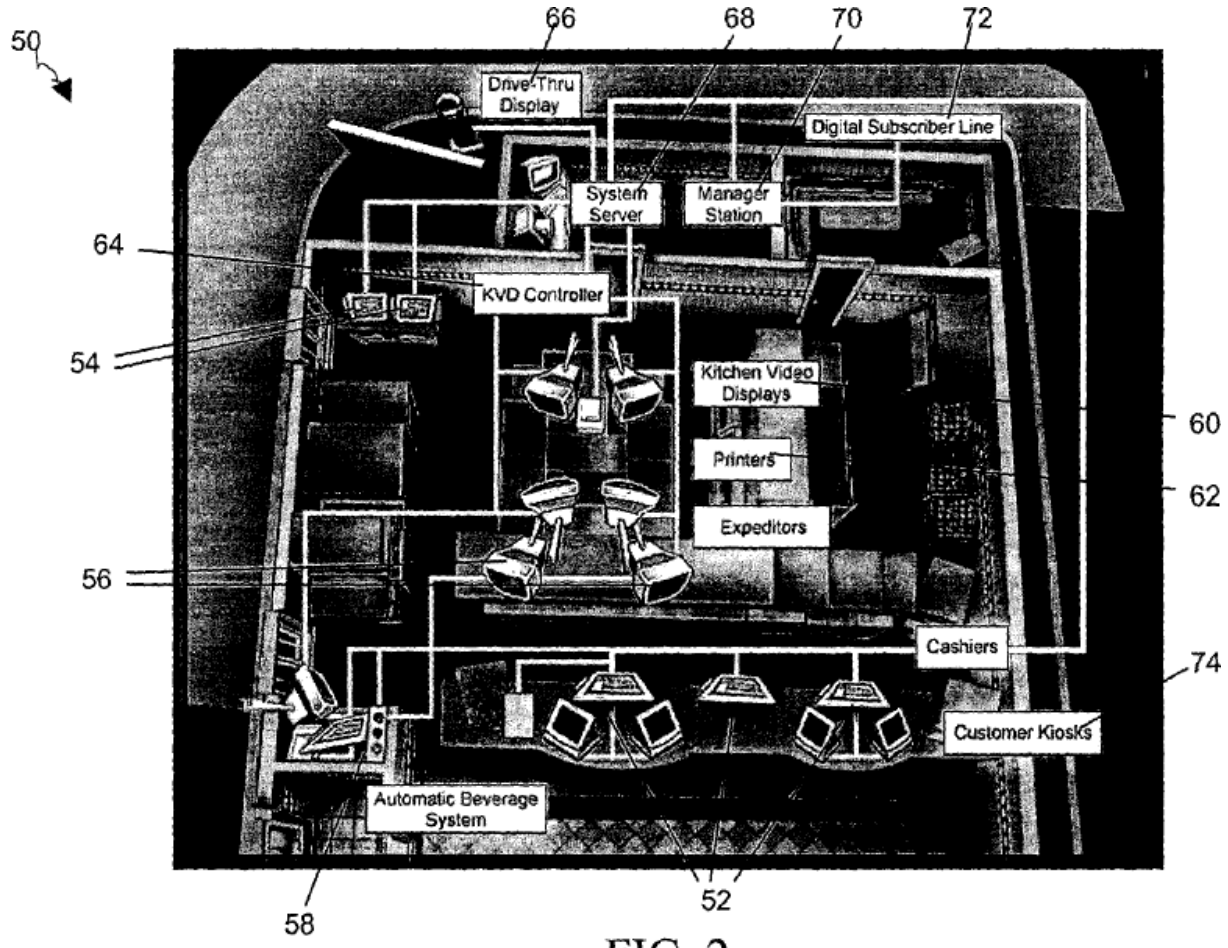


FIG. 2

168. Tengler's system is web-based because it allows "a manager to access a management database of a quick-serve restaurant location remotely through a web interface" "over the Internet." *Id.*, [0022], [0073]; *see also id.*, Claim 42, Figs. 7, 7A, 7B. As explained further for 1[c], Tengler discloses a POS builder system

because it includes "[a] user interface designer 614 [that] 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." *Id.*, [0103]; *see also id.*, [0107], [0121]. "Graphical User Interface (GUI) design objects 756 implement the logic for the user interface designer 614." *Id.*, [0121].

169. Accordingly, it is my opinion that Tengler discloses and suggests the limitations of 1[pre].

2. 1[a]

"at least one server configured to"

and 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"

170. Tengler discloses the "restaurant management software run[s] on server 464," which communicates with other in-store devices "using network 462" and is accessible "over the Internet." EX1005, [0078], [0073]; *see also id.*, [0109]-[0111]. Server "464 runs the system server software 604, the database 602, the

store-resident web server software 620, the production display service 609, and the drive-through display service 610." *Id.*, [0116]; see also [0115], Fig. 15:

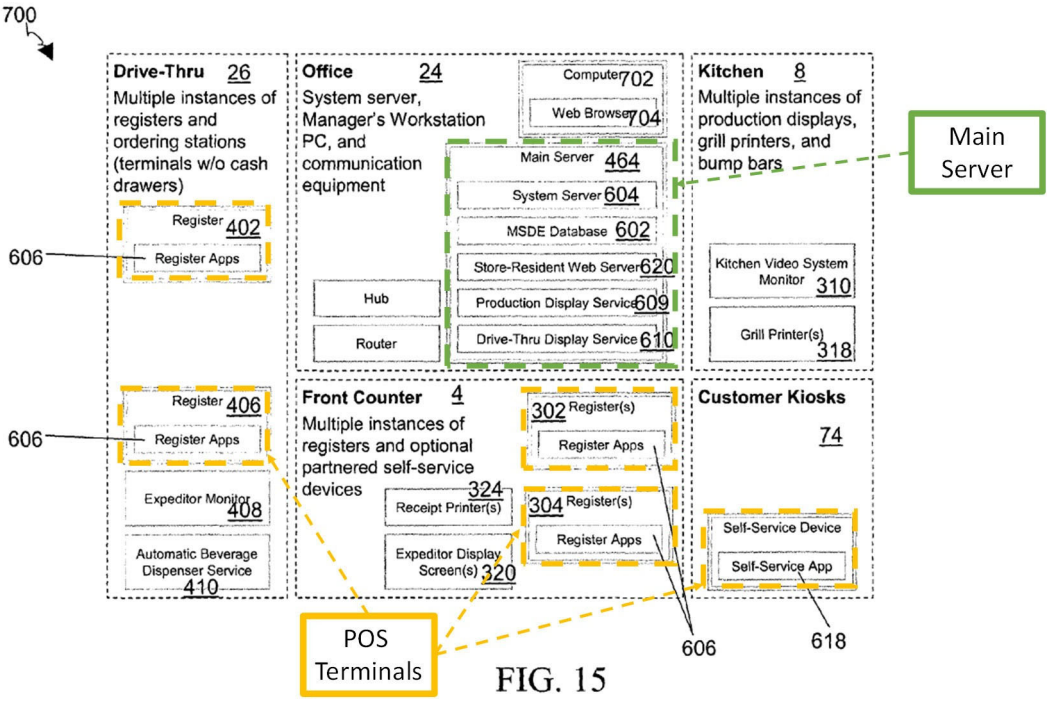


FIG. 15

171. As explained for 1[pre], Tengler discloses "stations enabling users to enter orders" (POS terminals) and "a network interconnecting the stations." *Id.*, [0011]; *see also id.*, [0014], Fig. 2 (order terminals 52). POS terminals/stations (e.g., "cashier stations 302, 304, 382, 402 [sic, 404], order entry stations 370, 372, 374," and "customer self-service kiosks 74") "are linked together using a network 462 (shown in FIG. 12)," which may be "a Transmission Control Protocol/Internet Protocol (TCP/IP) network" (e.g., the Internet). *Id.*, [0077]-[0078], [0074]; *see also id.*, Figs. 11-12:

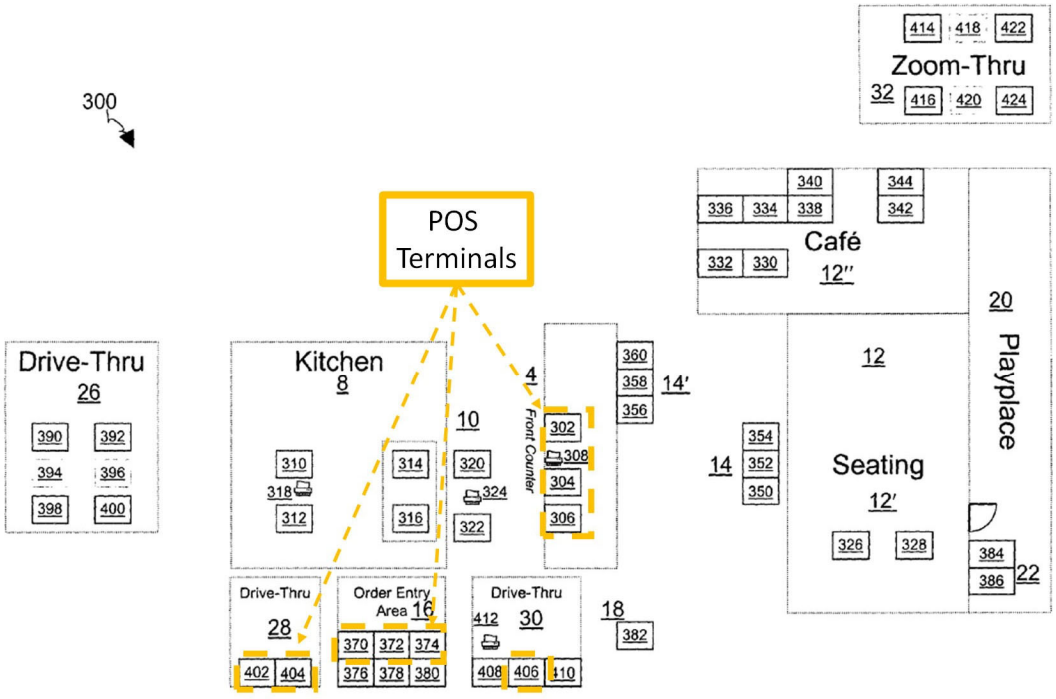

300
Zoom-Thru
32
414
418
422
416
420
424
POS Terminals
Café
12''
340
344
336
334
338
342
332
330
20
Playplace
Drive-Thru
26
390
392
394
396
398
400
Kitchen
8
10
4
Front Counter
360
358
356
14'
302
308
304
306
310
318
312
314
316
320
322
324
354
14
352
350
12
Seating
12'
326
328
384
386
22
Drive-Thru
28
402
404
Order Entry Area
16
370
372
374
376
378
380
Drive-Thru
412
30
408
406
410
18
382

FIG. 11

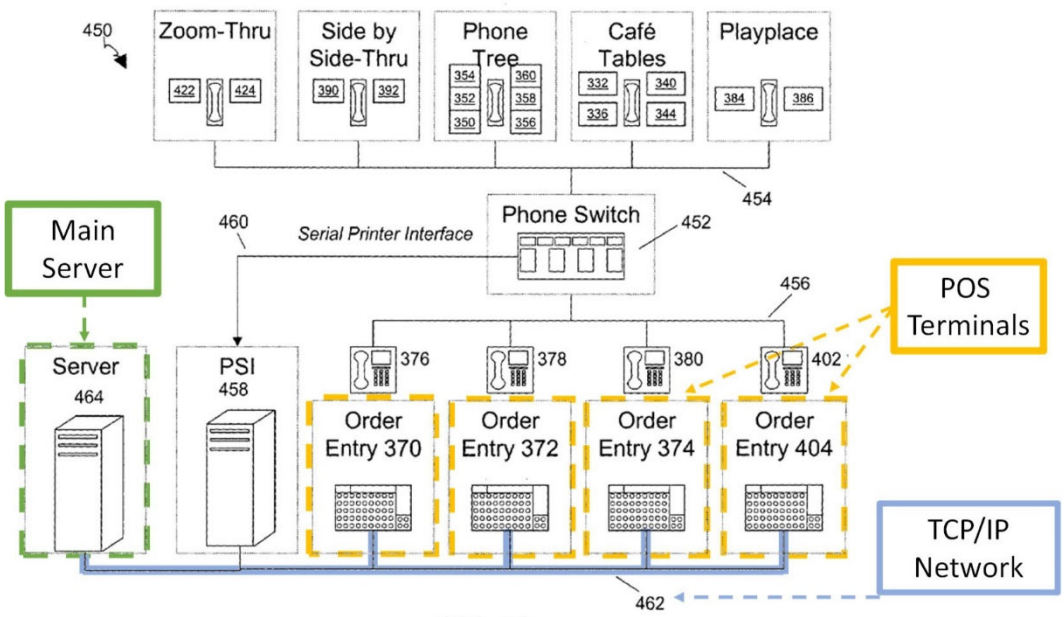

450
Zoom-Thru
422
424
Side by Side-Thru
390
392
Phone Tree
354
352
350
360
358
356
Café Tables
332
336
340
344
Playplace
384
386
454
Main Server
460
Serial Printer Interface
Phone Switch
452
456
POS Terminals
Server
464
PSI
458
376
378
380
402
Order Entry 370
Order Entry 372
Order Entry 374
Order Entry 404
TCP/IP Network
462

FIG. 12

Tengler further teaches that POS terminals in a remote call center communicate with server 464 via a remote proxy session (*id.*, [0060], [0114]) and managers can access server 464 "over the Internet" (*id.*, [0073], [0109]-[0111]). Tengler thus discloses and suggests that the network interconnecting Tengler's devices (manager computers, main server 464, and POS terminals) includes the Internet between at least some of its nodes (it is a network comprising the Internet).

172. Additionally, Tengler discloses that a central server may be used "for enterprise management of multiple restaurants," including by "provid[ing] features to manage a set of restaurants 2." EX1005, [0108]. By 2008, a POSITA would have known the Internet was a well-understood and routine way to implement a TCP/IP connection between nodes of a network, including between a shared central server and its clients. A POSITA would have been motivated to use the Internet to connect the central (web) server to in-store POS terminals (e.g., via server 464) to provide a simple, ubiquitous, and inexpensive network connection to stores in different locations managed from the central server. Internet connections would have been understood as a commodity or utility offering in the 2008 timeframe and would have been far easier to set up and configure than alternative long-distance networking technologies, such as a proprietary WAN. The Internet was the most logical and economical option from among a finite number of

predictable network options. A POSITA would be motivated to locate software and functionality common to multiple restaurants at the central server to facilitate Tengler's suggested "centralized point of control for enterprise management of multiple restaurants." EX1005, [0108]. For example, locating the POS builder and database functionality at the central server would enable a manager to update menus at multiple locations quickly and efficiently, and centralizing database functionality at the central server would provide accounting and reporting functionality for multiple restaurants from a single database. Installing Tengler's POS builder on an Internet-connected central server would also eliminate the need for redundant POS builder software at restaurants with multiple locations (e.g., a franchisee with multiple locations and common menus). In such an arrangement, the central server would communicate with POS terminals (e.g., to update POS screens and to receive order/customer data) over a network comprising the Internet and network 462. This simplified arrangement would have been easier to maintain from an information technology perspective and would have made it easier for an administrative user to maintain multiple locations, for example to roll out price or menu changes across a single franchisee's multiple locations.

173. Tengler's POS terminals/stations/kiosks display POS screens. Tengler discloses employees/cashiers "use a graphical user interface 100 displayed on the

order terminals 52, to enter orders" and "graphical user interface 125 displayed on the order terminals 52 to enter payment." EX1005, [0062]; *see also id.*, [0026], [0032]-[0033]; Figs. 3-4:



FIG. 3



FIG. 4

"Telephone order takers at the call center use the same interface 100 as order takers in restaurant 2 to enter customer orders." *Id.*, [0114]. Similarly, "[c]ustomer self-service kiosks 74 enable customers to enter orders" themselves, e.g., via "an interface 200 on a touch screen" and "interface 220 [that] enables the customer to add different food items to the order." *Id.*, [0074]-[0076]; Figs. 8-9:



FIG. 9

Customers may also pay "by swiping a credit card at the customer self-service kiosk." *Id.*, [0055].

174. Accordingly, it is my opinion that Tengler discloses and suggests limitations 1[a]-1[b], and further renders them obvious in view of the knowledge of a POSITA.

3. 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"

175. Tengler discloses a POS builder as "user interface designer 614"[7] remotely accessible via web server 620. EX1005, [0103], [0107], [0121]. Managers access the POS builder through manager graphical user interfaces such as 190, 192, and 196 (POS builder interface), which are accessible via Internet communication with server 464 (over the network including the Internet). EX1005, [0073], [0103], [0107], [0109]; Figs. 7, 7A-7B.

176. As explained for limitations 1[a]-1[b], main server 464 includes web server software 620 and database 602. *Id.*, [0115]-[0116], Fig. 15. "The store resident web server 620 also allows managers to edit the user interface of the register [606] and self-service applications [618] and also saves the specifications in the database 602." *Id.*, [0103]; *see also id.*, [0073], [0107]. The server thus receives information over the network from the POS builder interface used to create/edit user interfaces of register and self-service applications (POS screens). Server 464 then configures POS terminals running register application 606 and

[7] References herein to "user interface designer" include related software and system objects. *See* EX1005, Fig. 16, [0121].

self-service application 618 to display the new/modified POS screens. *Id.*, [0103], [0107], Fig. 15:

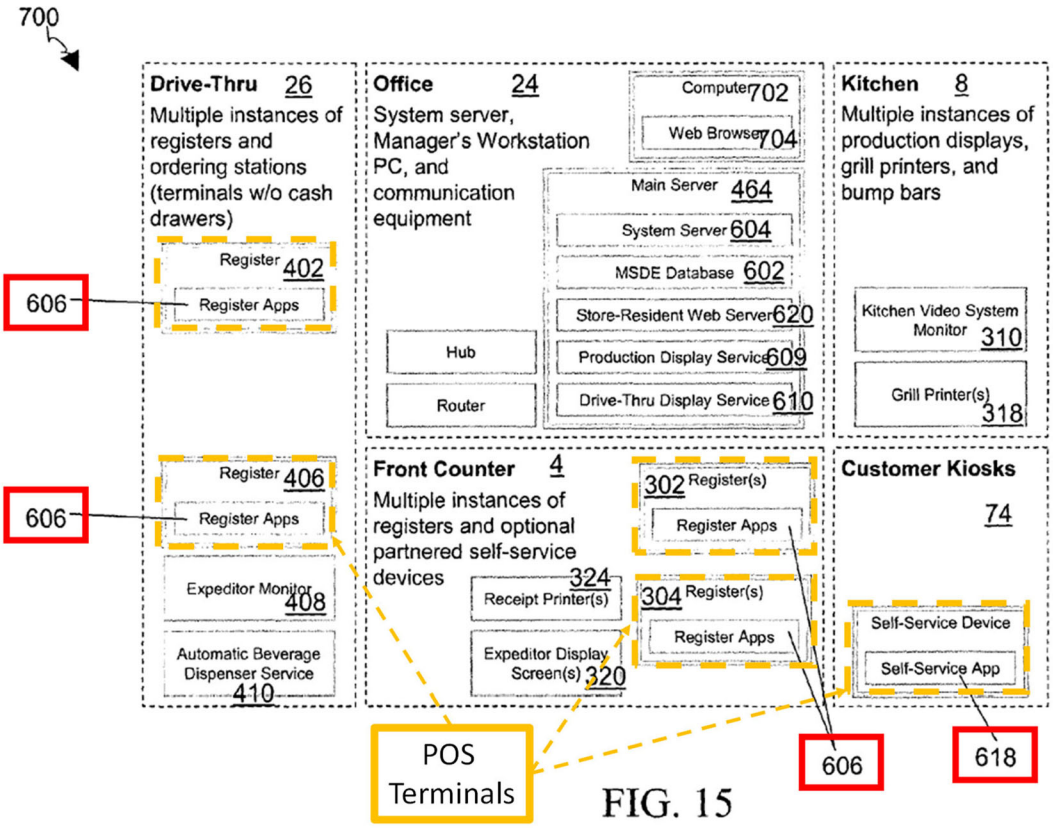


FIG. 15

177. For example, "managers can … modify menus and change prices using interfaces 192 and 196." *Id.*, [0073]. The "product editor" screen of interface 192 lists various categories that can be modified/edited (including recipe, item type, product info, and pricing) and options for adding new products and copying existing products. *Id.*, [0073]; Fig. 7A:

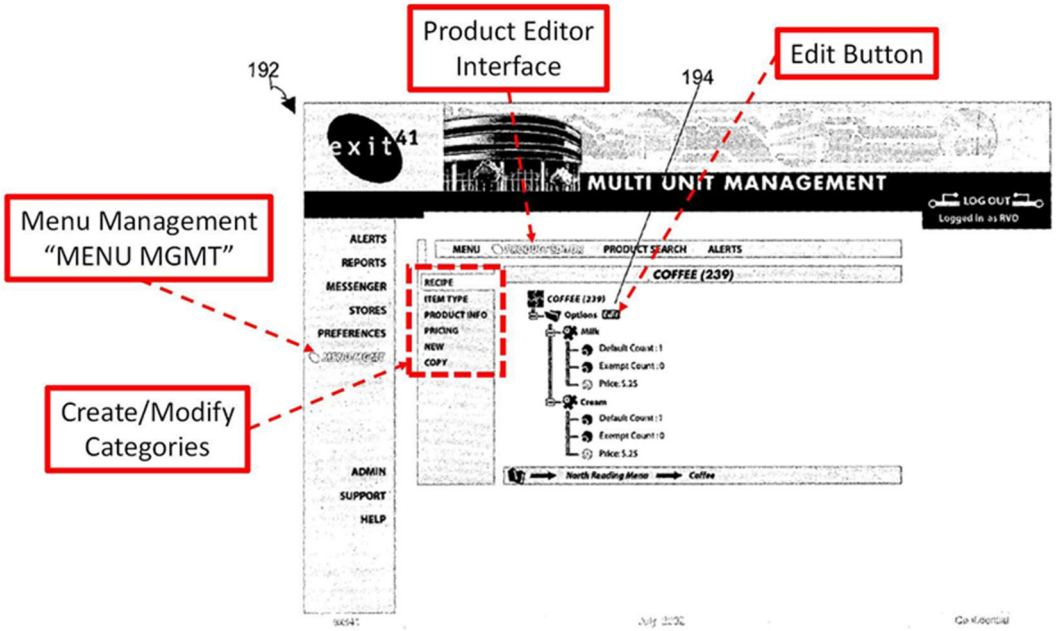


FIG. 7A

The "MENU MGMT" screen of Interface 196 includes options for managing and modifying menus, items, "discounts," and "merchandise." EX1005, Fig. 7B:

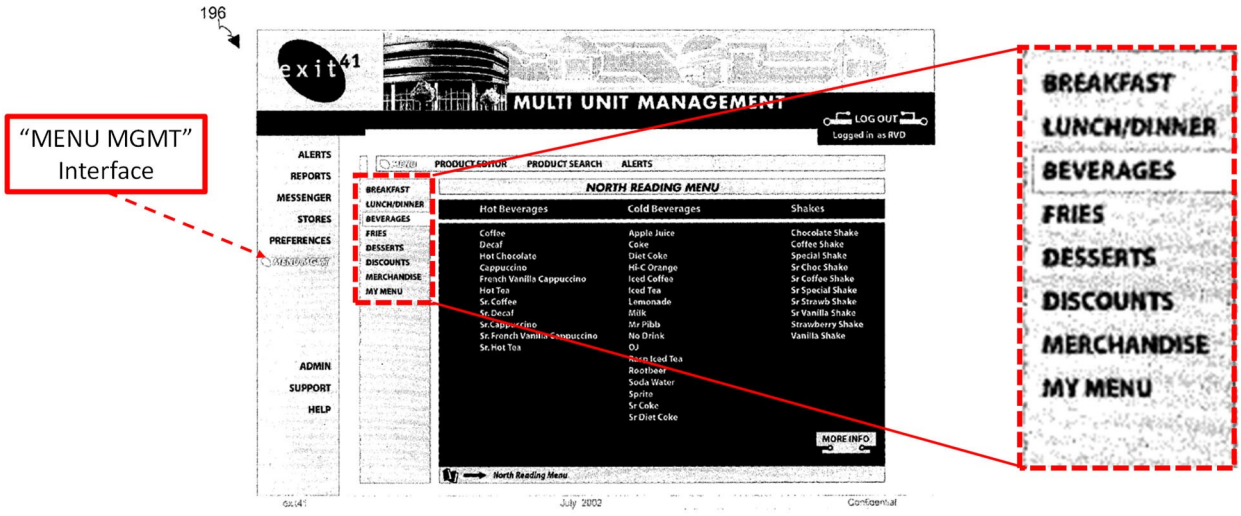


FIG. 7B

178. Using POS builder interfaces such as 192 and 196, managers provide information used to create/modify items associated with buttons displayed on POS

screens. *See* § XIII.2; EX1005, [0062], [0074]-[0076], [0032], [0040]. Tengler discloses various layouts of circular, square, and rectangular buttons (display interfaces) of POS screens associated with items for sale and placing orders, and also display interfaces associated with order information (items, price, tax, total, etc.). *See id.*, Figs. 3-4, 8-10. Figs. 3, 9:

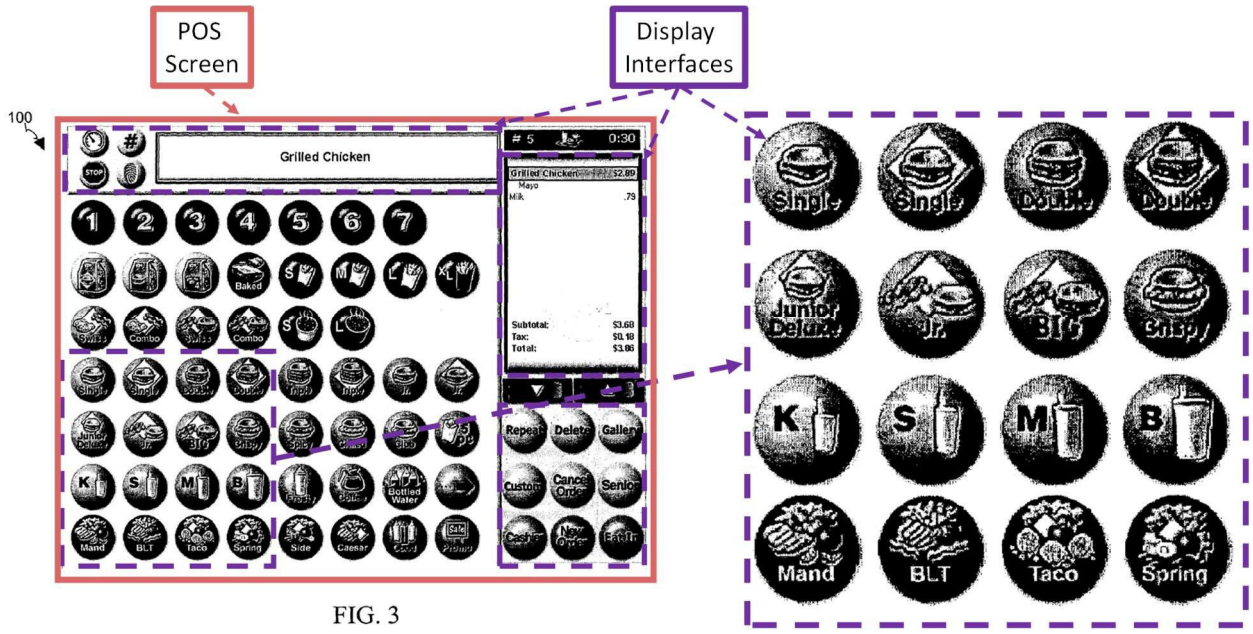


FIG. 3

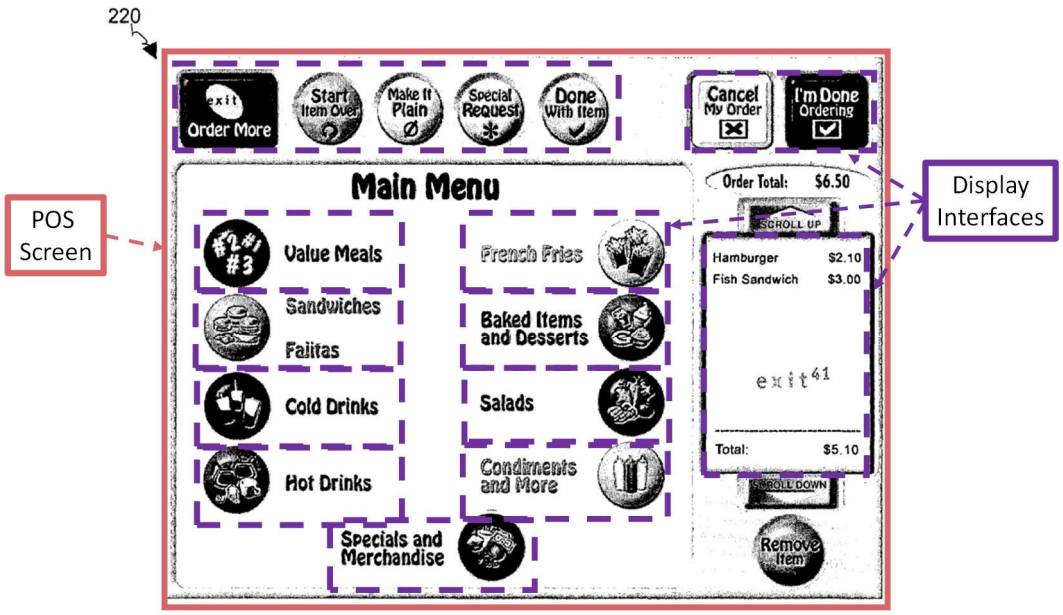


FIG. 9

179. Tengler thus discloses and suggests interfaces 192 and 196 allow managers to create/modify POS screens displayed on POS terminals, including by providing information used to create or modify display interfaces (buttons and their associated text/labels) associated with new/modified menu and other categories/items (food, discounts/specials, and merchandise). When managers add or remove items (and/or their associated attributes/options) from a menu, such interfaces (192/196) would provide information to the server used to modify POS screens (including screen layouts and hierarchies, buttons, and item names/prices). For example, adding a new food/beverage category, item, or option would create a new or modify an existing button/key on the POS screens for displaying the new category/item/option. EX1005, [0076], [0062] (citing Fig. 9 (showing categories of

items for sale) & Figs. 3/4 (POS screens with "Grilled Chicken" with "Mayo" added to order)). Additionally, a POSITA would have understood modifying an item name would in turn modify the button on the series of POS screens associated with that item (e.g., if the button for "Bottled Water" in Fig. 3 was edited to "Water").

180. Tengler thus discloses and suggests server 464 receiving, from a POS builder interface over the network comprising the Internet, information used to create/modify POS screens, including information used to create/modify display interfaces (buttons) associated with items.

181. Accordingly, it is my opinion that Tengler discloses and suggests the limitations of 1[c].

4. 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"

and 1[g]

"wherein said further information regarding the one or more POS transactions relate to one or more transactions by corresponding customers respectively

associated with at least one of said one or more POS terminals"

182. Server 464 receives, via the network, further information regarding POS transactions from ordering stations/kiosks (POS terminals) used by employees and customers to order items. For example, database 602 of server 464 "maintains a complete history of orders for a long period of time for later analysis and display." EX1005, [0089]. Figure 17 illustrates "an example of an object-oriented data structure for an order" corresponding to items purchased. *Id.*, [0125], Fig. 17:

800

802 Class	804 Guest Check		Level 806
808 Order	Order #591456		0
ItemLine	810 1 Hamburger	2.29	1
OptionLine	812 *no onions*		2
OptionLine	814 *extra mustard*		2
ItemLine	816 2 EVM #1 SS	9.78	1
ComponentLine	818 Dbl QP/Cheese		2
OptionLine	824 *side pickles*		3
ComponentLine	820 SS Fries		2
OptionLine	826 *no salt*		3
ComponentLine	822 SS Coke		2
ItemLine	1 Lg Coke	1.29	1
SubtotalLine	Subtotal	13.36	1
TaxLine	MA Tax 5%	0.67	1
TotalLine	Total	14.03	1
TenderedLine	Cash Tendered	20.00	1
ChangeLine	Change	5.97	1

FIG. 17

An object-oriented class definition encapsulates definitions of data and functions relating to a class of software objects. Object oriented programming (OOP) can ease system programming by providing implementation specific details as appropriate to specific classes of objects. For example, a Guest Check might have an amount tendered ("TenderedLine" in Fig. 17) and might have a method for TenderPayment. If the payment is cash, the system might invoke a more specific

implementation of a Guest Check object called Cash Guest Check. The TenderPayment method would likely prompt the user to enter the payment amount (e.g., a $20 bill), open the cash drawer, and display the change due (e.g., $5.97). If the payment is credit, the system might invoke a more specific implementation called Credit Guest Check. The TenderPayment method for a credit card payment might prompt the user to swipe a card through a magstripe reader and sign the screen with their finger. The POS system need only know the names of the specialized guest check class definitions and can call the TenderPayment method without knowing anything else about the underlying details of that payment type. *See* EX1005, [0121] (explaining that object-oriented programming allows implementation details to be isolated so an object definition corresponding to a device, e.g., a bump bar for signaling completing of an order, from one vendor does not change if a different vender begins offering a similar kitchen input device). Tengler identifies Java as a suitable object-oriented programming environment.

183. Tengler discloses all of this further information about customer transactions (e.g., items ordered, options selected, prices, tax, total) is transmitted from POS terminals over the network and stored in a database of server 464. *Id.*, [0049]-[0051], [0055], [0058], [0089], [0116], [0121]. Tengler discloses and

suggests such information relates to transactions by corresponding customers respectively associated with POS terminals because employees and customers use POS terminals to enter customer orders and because, as explained below, customer images are associated with their respective orders.

184. Tengler also discloses and suggests 1[b] and 1[g] by "receiving an electronic image of a customer, and associating the electronic image with an order of a customer," which "is captured at the ordering location." EX1005, [0017]-[0018]. When customers "directly place orders using customer self-service kiosks" where they can also make payment, "the kiosk takes a picture of the customer with a digital camera and enters the image into the database, linked to the order." *Id.*, [0055]; *see also id.*, Claims 16, 19, 21-22, 36. Additional examples include, "an order number 132, the contents of the order 134," changes to the order, and "information about the location of the customer placing the order." *Id.*, [0063], [0079], Claim 37; *see also id.*, [0051], [0083], [0125]-[0127]. When an order is created or updated, server 464 receives Information regarding "the current time," the POS terminal/station "that issued the update, and the worker using the station." *Id.*, [0128].

185. Thus, Tengler discloses and suggests server 464 receives from POS terminals over the network further information regarding POS transactions

corresponding to items (1[d]) that is related to transactions by corresponding customers respectively associated with the POS terminals (1[g]).

186. Accordingly, it is my opinion that Tengler discloses and suggests the limitations of 1[d] and 1[g].

5. 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"

187. As explained for limitations 1[a]-[c], server 464 configures stations/kiosks (POS terminals) to display POS screens based on information provided from manager user interfaces (POS builder interface) over the network comprising the Internet. EX1005, [0073], [0103], [0109], [0115]-[0116], Fig. 15. Tengler also discloses and suggests server 464 configures POS terminals with information from the POS builder interface to create/modify a series of POS screens based at least in part on further information provided by employees/customers relating to orders (e.g., user selections) and received/stored by server 464.

188. For example, "interface 220 enables the customer to add different food items to the order" and to "add more food to the order by clicking on a button

234." *Id.*, [0076], Figs. 9-10. During the ordering process, in which the customer inputs this "item add/update information" as the "further information," POS terminals display successive POS screens with such items (and options/prices) added to the order. *See id.*, Figs. 3-4, 9-10 (displaying selected items, options, and prices).

189. When paying for an order, the POS screen is modified to include details of the order (e.g., items, options, price, tax, etc.) based on user selections, which is further information about the transaction transmitted to server 464 as part of order information. EX1005, [0062], Fig. 17, Fig. 4:



FIG. 4

The same applies for self-service kiosks. EX1005, [0076], Fig. 10:

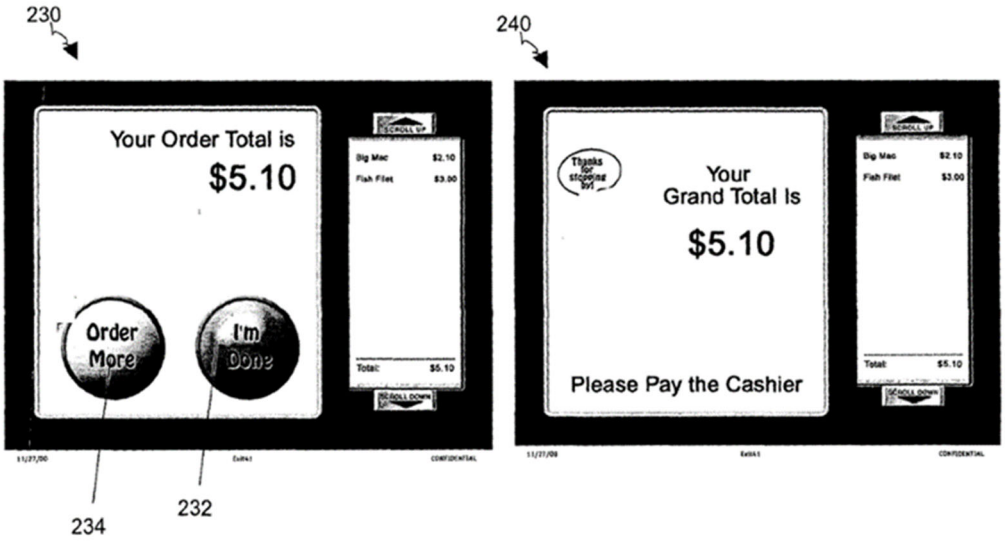


FIG. 10

190. As another example, Tengler discloses and suggests interface 196 allows managers to create/modify menu categories and sub-categories and items/options, thus creating/modifying hierarchical POS screens to be displayed based on user selections (further information regarding POS transactions). *Id*., Figs. 7B & 9. Addition of a new category item (e.g., "brunch" in Fig. 7B) or sub-category item (e.g., "other beverages" in Fig. 7B) and food/beverage items therein would create new or modified POS screens and buttons for such categories/items to be displayed on POS terminal. *Id.*, Fig. 7B Addition or modification of a food/beverage/discount item, its price, or its options would likewise create or modify successive POS screens with buttons for such items/options to be displayed

on POS terminals during a transaction, and further information about user selections/POS transactions would be transmitted to and stored at server 464.

191. POS screens are also modified based on the customer's image so orders can be delivered to the right person. EX1005, [0017]-[0019]. "The contents of the order may be displayed with the electronic image" of the customer captured "at the kiosk when the order is placed." *Id.*, [0018]-[0019]; *see also id.*, Claim 17, [0055], [0065], [0074], [0088]. Images can be displayed on the same terminals used to enter orders and take payment, e.g., when ordering from a drive-through cashier. *Id.*, [0058], [0062], [0065], [0077] ("drive-through cashier payment area 28 includes … order entry"), [0088]. Tengler thus discloses and suggests that a POS screen displaying order contents is modified to include the electronic image of the customer. A POSITA would have understood instructions from manager user interfaces (the POS builder interface) dictate such POS screen behavior/display.

192. When orders are updated, the "server 604 broadcasts the new version of the order to applications such as register applications 606 [and] self service ordering 618" applications running on the stations/kiosks. EX1005, [0126]. Specifically, "RMS server 604 updates the display model and rebuilds the display"

(POS screen) shown on the station/kiosk. *Id.*, [0127]. Server 604 is part of main server 464. *Id.*, [0116], Fig. 15.

193. In all these examples, Tengler discloses and suggests that the information provided by the manager regarding menu items, categories, options, pricing, tax rate, customer recognition, etc., is used to configure POS terminals/stations/kiosks to create and modify POS screens based on further information about orders (POS transactions) and respective corresponding customers.

194. Accordingly, it is my opinion that Tengler discloses and suggests 1[e].

6. 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"***

195. Tengler discloses information received from the POS builder interface to create/modify POS screens includes information for adding/updating items, tax information, discount information, and promotion information, any of which is alternatively sufficient to show 1[f].

196. As explained for 1[b], Tengler discloses POS screens showing menu, item, and price information with buttons/keys allowing employees and customers to order items in a restaurant. EX1005, [0062], Figs. 3-4:



FIG. 3



FIG. 4

See also id., [0074]-[0076]; Fig. 9:

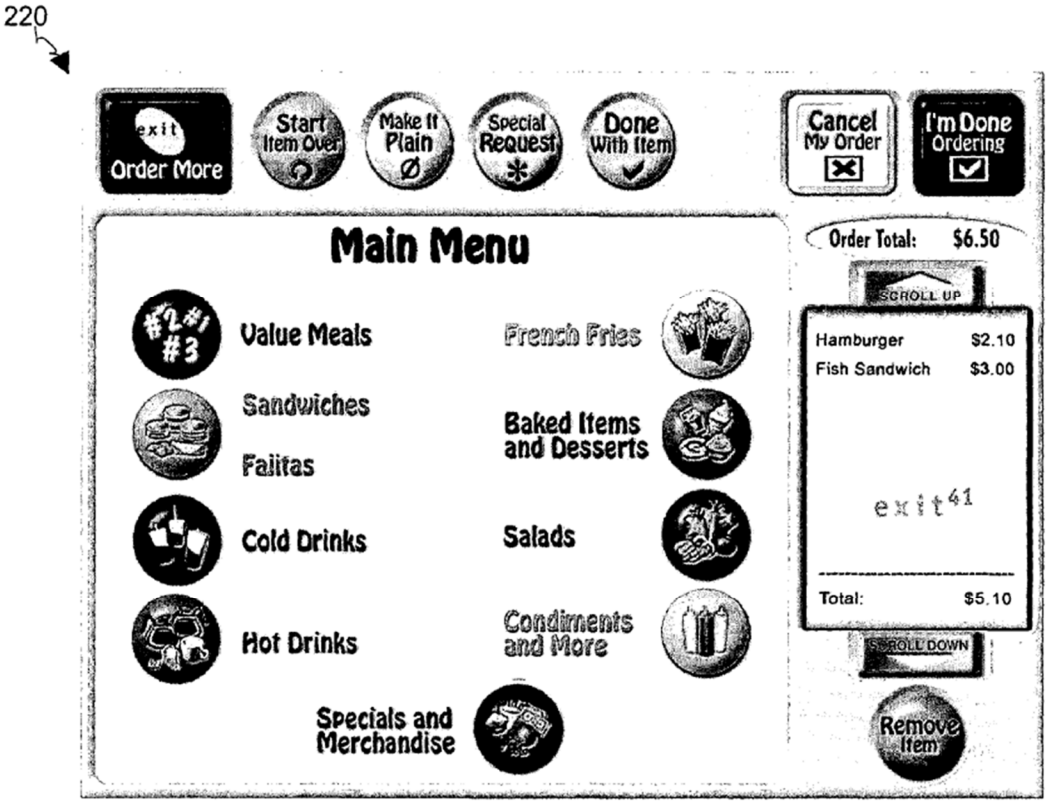
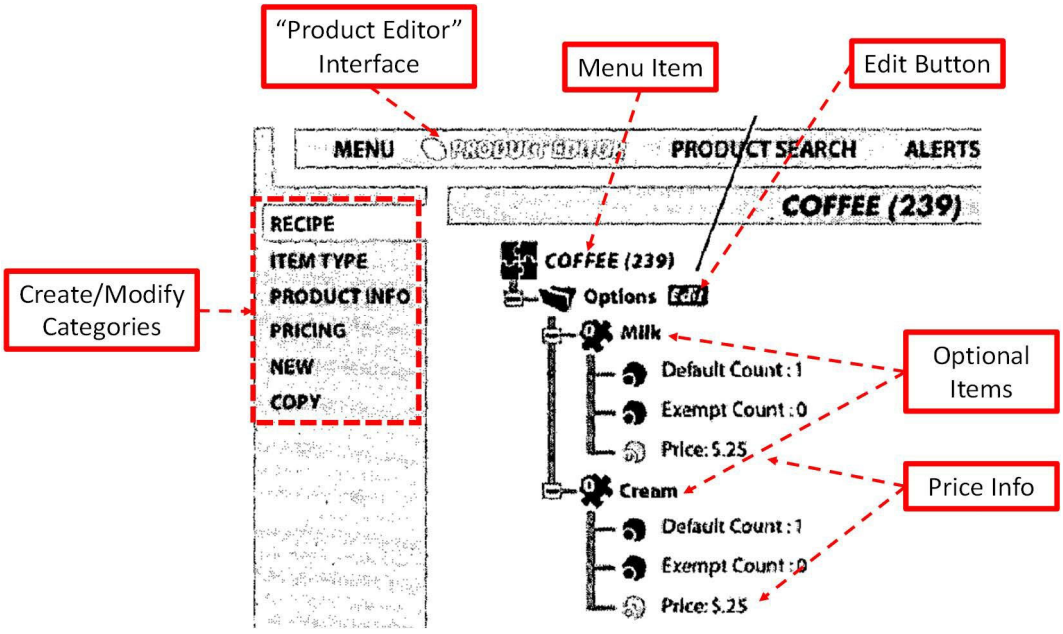


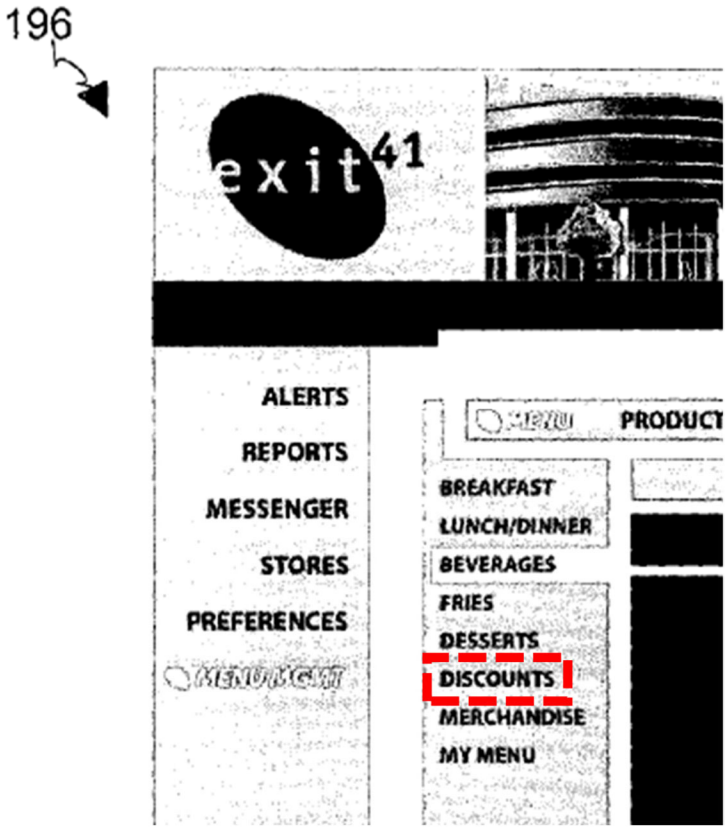
FIG. 9

197. As explained for 1[c], Tengler discloses and suggests a POS builder interface managers use to create/modify such POS screens and items associated with buttons. EX1005, [0073], [0103], [0107], [0109]; Figs. 7A-7B. That interface includes options for creating new menu items and editing the "recipe," the "item type," "product info," and "pricing" information for menu items. *Id.*, Fig. 7A (excerpted):



The information managers provide via interface 192 to create/modify menu items that appear on POS screens thus includes item add/update information.

198. Tengler also discloses managers can input discount information using interface 196. EX1005, Fig. 7B (excerpted):



Tengler thus discloses and suggests using interface 196, managers can create/modify "discounts," thus providing "discount information" for display on POS screens (and suggesting "promotion information") is provided by managers and used for creating/modifying POS screens displayed on POS terminals. *Id.*, Fig. 7B; *see also id.*, Fig. 9 (button for "Specials and Merchandise"), [0103] ("The self service ordering application 618 is a platform for promotions and loyalty programs.").

199. Further, Tengler discloses and suggests managers can enter state/local sales tax percentages (e.g., "MA Tax 5%"). *Id.*, Fig. 17:

800

802 Class	804 Guest Check		Level 806
808 Order	Order #591456		0
ItemLine	810 1 Hamburger	2.29	1
OptionLine	812 *no onions*		2
OptionLine	814 *extra mustard*		2
ItemLine	816 2 EVM #1 SS	9.78	1
ComponentLine	818 Dbl QP/Cheese		2
OptionLine	824 *side pickles*		3
ComponentLine	820 SS Fries		2
OptionLine	826 *no salt*		3
ComponentLine	822 SS Coke		2
ItemLine	1 Lg Coke	1.29	1
SubtotalLine	Subtotal	13.36	1
TaxLine	MA Tax 5%	0.67	1
TotalLine	Total	14.03	1
TenderedLine	Cash Tendered	20.00	1
ChangeLine	Change	5.97	1

FIG. 17

As explained for 1[e], the item, price, and tax information are used to create/modify POS screens. EX1005, Fig. 4:

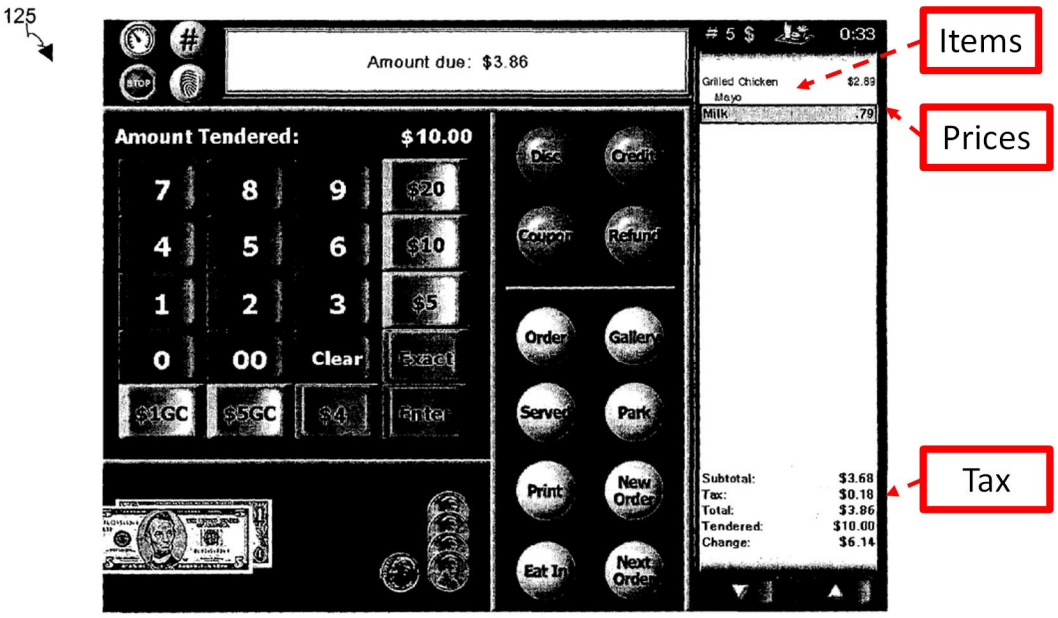


FIG. 4

200. Limitation 1[f] can be alternatively shown by further information. Tengler's further information regarding POS transactions received from the POS terminals includes item add/update, employee clock, customer add/update, and loyalty/promotion information.

201. Tengler discloses receiving information from POS terminals regarding orders (POS transactions) that includes item add/update information, e.g., for items purchased and options selected. EX1005, [0076], Figs. 9-10; *see also id.*, [0127]-[0131]. As explained for 1[d], orders may be updated with further information regarding changes to the order. *Id.*, [0125]-[0127]. For example, a "customer may add items to the order" before payment (item add/update information). *Id.*, [0083]; *see also id.*, [0076]. When an order is updated, the server also receives information

regarding "the current time … and the worker using the station" (employee clock information). *Id.*, [0128].

202. As explained for 1[d], the ordering station/kiosk (POS terminal) takes a picture of the customer and stores the image in the database. EX1005, [0017]-[0019], [0055]. There, "[i]nformation about the customer is associated with the electronic image of the customer," e.g., "historical information about orders previously placed by the customer." *Id.*, [0018]. A POSITA would have understood that creating/updating a customer record or order in the database with an image of the customer is or suggests customer add/update information received from POS terminals.

203. Tengler also discloses "[t]he self service ordering application 618" running on the kiosk "is a platform for promotions and loyalty programs." EX1005, [0103]; *see also id.*, Fig. 9 (button for "Specials and Merchandise"). Tengler's disclosure of a "discounts" category for interface 196 and a "Specials and Merchandise" button on a POS screen for the self-service ordering application suggests further information provided by POS terminals to server 464 includes promotion information, discount information, and/or loyalty point information (e.g., to track performance of promotions and discounts and to associate the order with the customer's loyalty point account). *Id.*

204. Accordingly, it is my opinion that Tengler discloses and suggests limitation 1[f].

B. Claims 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"

and 11

"The web-based point of sale (POS) builder system of claim 1, wherein the POS builder interface is accessible via a web browser"

205. As explained for 1[c], Tengler's POS builder (which provides manager GUIs such as 190, 192, and 196) runs on "main server computer 464" (computing device). EX1005, [0109], [0116], *see also id.*, [0103], [0107]. Managers access manager GUIs (POS builder interface) "over the Internet" "using the web server 620," which allows access "remotely through a web interface." *Id.*, [0073], [0022]; *see also id.*, [0109]. For example, managers can use "web browser 704" on computer 702 (computing device). *Id.*, [0116], [0073]. Tengler thus discloses the POS builder interface is configured to run on a computing device and is accessible via a web browser.

206. Accordingly, it is my opinion that Tengler discloses and suggests the limitations of 2 and 11.

C. Claim 3

"The web-based point of sale (POS) builder system of claim 1, wherein the one or more POS terminals comprise a plurality of POS terminals in a plurality of locations"

207. As explained for limitation 1[b], Tengler discloses a plurality of POS terminals in a plurality of locations within and around "restaurant 2." EX1005, [0073]-[0075], [0077]-[0078], [0115]-[0120], Figs. 11, 15:

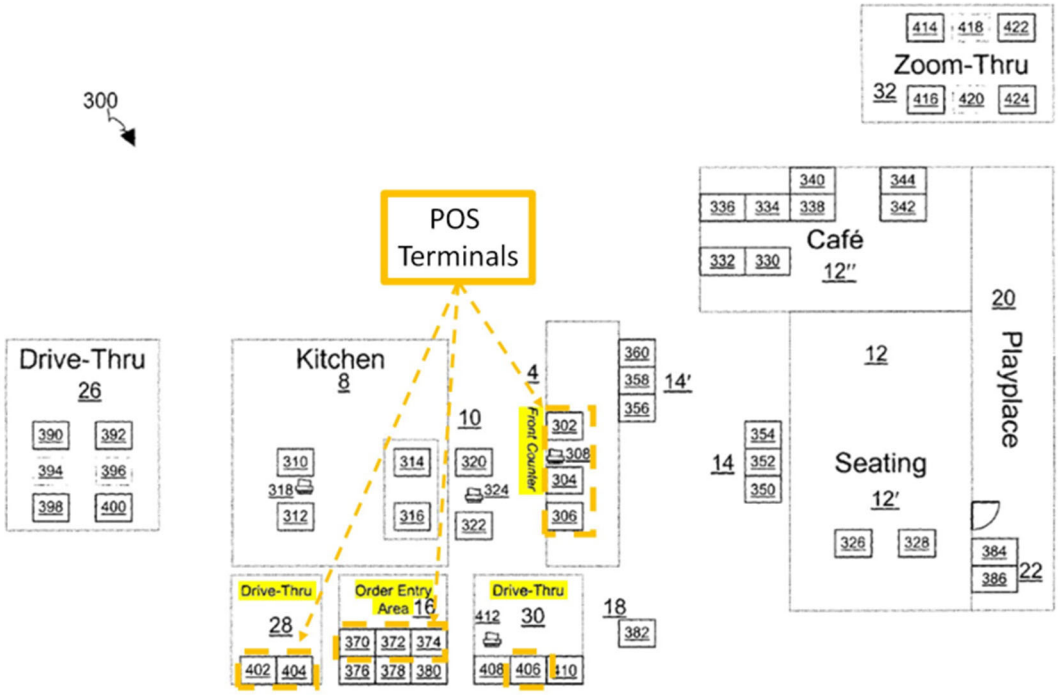


FIG. 11

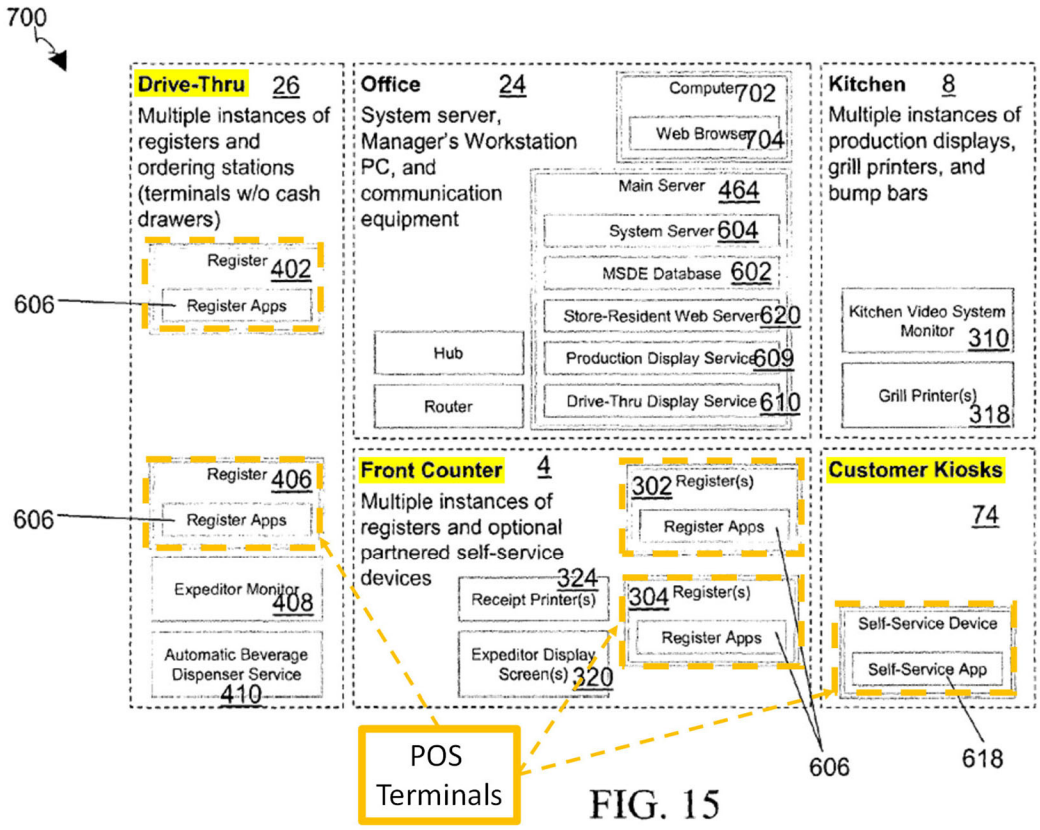


FIG. 15

208. Tengler also discloses a plurality of POS terminals running register applications 627 via proxies with RMS server 604 (on server 464) at a remote call center where "telephone order takers at the call center use the same interface 100 as order takers in restaurant 2 to enter customer orders." *Id.*, [0114]; *see also id.*, [0016], [0025], [0112]-[0113]. Tengler thus discloses and suggests a plurality of in-store terminals and a plurality of remote call center terminals in network communication with server 464. Additionally, Tengler discloses "enterprise management of multiple restaurants 2," which suggests a plurality of POS

terminals in a plurality of restaurant locations. *Id*., [0108]; *see also id.*, [0109]-[0111].

209. Accordingly, it is my opinion that Tengler discloses and suggests the limitations of 3.

D. Claims 4

"The web-based point of sale (POS) builder system of claim 1, wherein the one or more items comprise at least one of: one or more items for sale, one or more promotions, or one or more loyalty points programs"

and 9

"The web-based point of sale (POS) builder system of claim 1, wherein the one or more display interfaces comprise one or more buttons or keys"

210. As explained for 1[c], Tengler discloses items associated with buttons/keys (display interfaces) on POS screens that are added/updated via the POS builder interface are items for sale, e.g., "food items" in a restaurant (claim 4). EX1005, [0003], [0062], [0076], Figs. 3, 9:

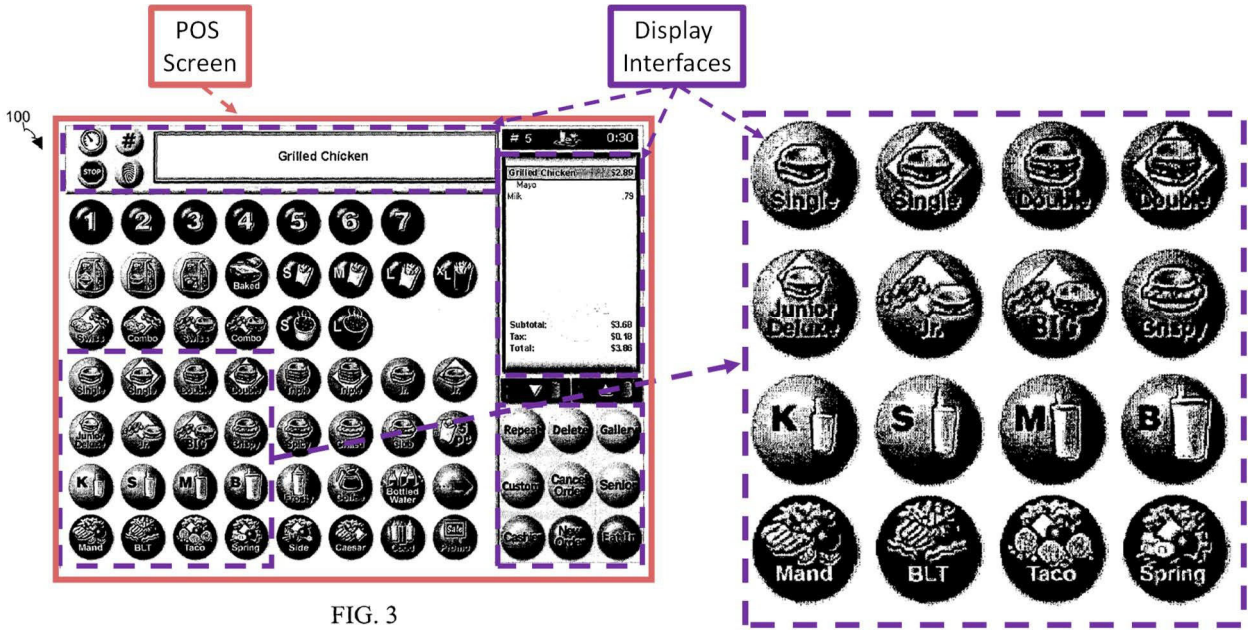


FIG. 3

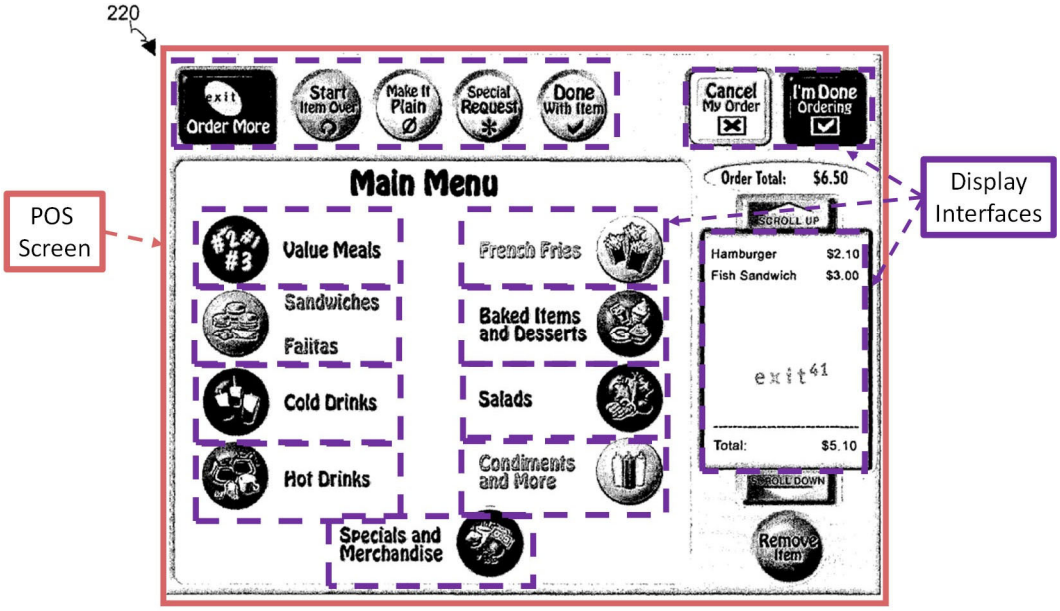


FIG. 9

Further, as explained for 1[f], Tengler discloses "[t]he self service ordering application 618" running on the kiosk is also "a platform for promotions and loyalty programs." *Id.*, [0103]; *see also id.*, Fig. 9 (button for "Specials and

Merchandise"). Tengler thus discloses and suggests allowing managers to create and modify buttons/keys (display interfaces) on POS screens (claim 9) related to such promotions/specials and loyalty point programs (and items associated with such promotions/programs) the same way managers do for food items, e.g., to allow customers to redeem discounts/promotions/loyalty points and check loyalty point balances. Tengler thus discloses and suggests the items comprise items for sale, promotions, and loyalty points programs (claim 4) and such items are associated with display interfaces comprising buttons and keys (claim 9).

211. Accordingly, it is my opinion that Tengler discloses and suggests the limitations of claim 4.

E. Claim 7

"The web-based point of sale (POS) builder system of claim 1, wherein the information regarding one or more POS transactions comprises one or more of the employee clock information, the customer add/update information, the item add/update information, or the promotion information"

212. As explained for 1[e]-1[f], Tengler's further information regarding POS transactions includes item add/update, customer add/update, employee clock, and promotion information (e.g., as part of a loyalty program).

213. Tengler discloses server 464 receiving information from POS terminals regarding orders (POS transactions) that includes item add/update

information based on customers adding/updating items in an order. EX1005, [0076], Figs. 9-10. Tengler discloses receiving "[i]nformation about the customer," including images of customers and order histories, saving the information in a database, associating it with orders, and displaying it on POS screens (customer add/update information). EX1005, [0017]-[0019], [0055], Claim 22. When an order is updated, the server also receives information regarding "the current time … and the worker using the station" (employee clock information). *Id.*, [0128]. Further, because Tengler's POS terminals/screens provide "a platform for promotions and loyalty programs" and include a "Specials and Merchandise" menu option (*id*., Fig. 9), it would be obvious that promotion information is received from POS terminals and stored on server 464 when participating customers purchase discounted items and/or items subject to a promotion. EX1005, [0103].

214. Accordingly, it is my opinion that Tengler discloses and suggests the limitations of claim 7.

F. Claim 8

"The web-based point of sale (POS) builder system of claim 1, wherein the information regarding one or more POS transactions are viewable via the POS builder interface"

215. As explained for 1[c], Tengler's POS builder interfaces includes manager GUIs 190, 192, and 196 accessible via main server 464, which also

includes database 602. EX1005, [0073], [0103], [0107], [0109], [0115]-[0116]; Figs. 7A-7B. Tengler's "database 602 … maintains a complete history of orders for a long period of time for later analysis and display." *Id.*, [0089]. Managers can access that information "remotely through a web interface," e.g., "using the interface 190," which includes a "Reports" tab. *Id.*, [0022], [0103], Fig. 7:

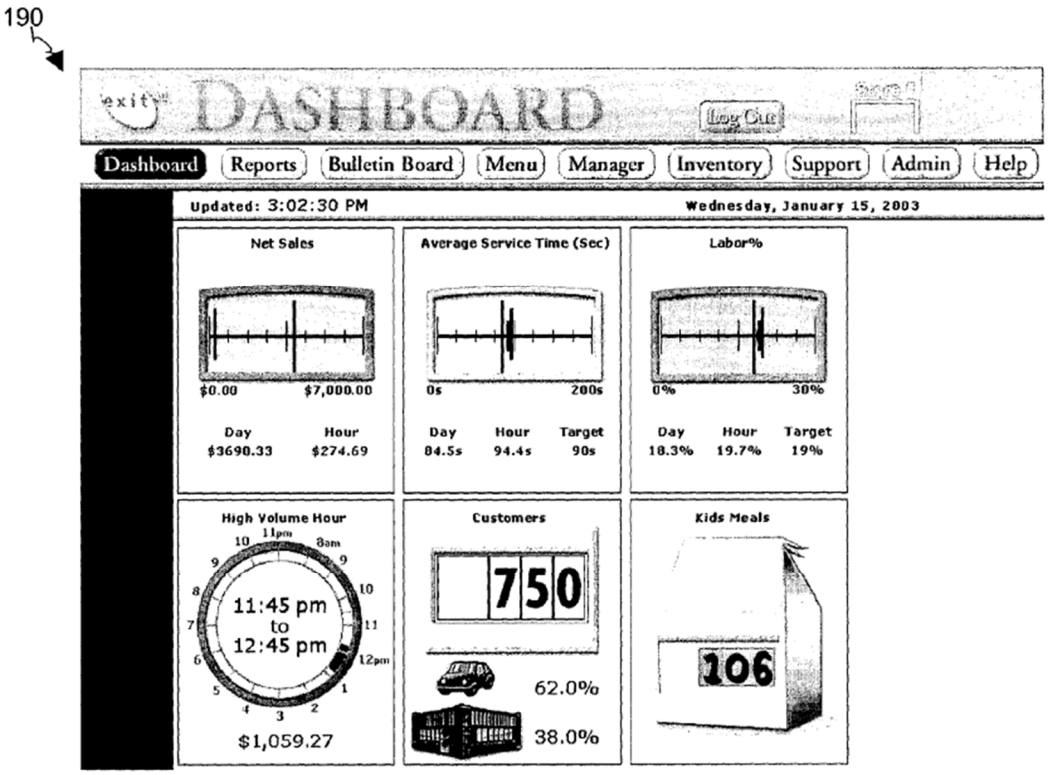


FIG. 7

Because interface 190 is a "Dashboard" interface to access reports, menus, etc., it is part of Tengler's POS builder interface. Tengler thus discloses and suggests that managers can view (via the POS builder interface) customer orders (POS

transactions) and "Reports" regarding customer orders, each containing information regarding POS transactions.

216. Accordingly, it is my opinion that Tengler discloses and suggests the limitations of claim 8.

G. Claim 10

"The web-based point of sale (POS) builder system of claim 1, wherein the received information comprises information indicative of at least one of a number, shape, or arrangement of the one or more display interfaces"

217. As explained for 1[c], Tengler's "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." EX1005, [0103]. Managers edit the user interfaces employees and customers use to place orders by creating/modifying menus, items, and thus their associated buttons displayed on POS screens. *Id.*, [0062], [0074]-[0076]. Tengler discloses various layouts/arrangements of circular, square, and rectangular buttons (display interfaces) and other interfaces displaying information regarding an order. *See id.*, Figs. 8-10. Figs. 3, 9:

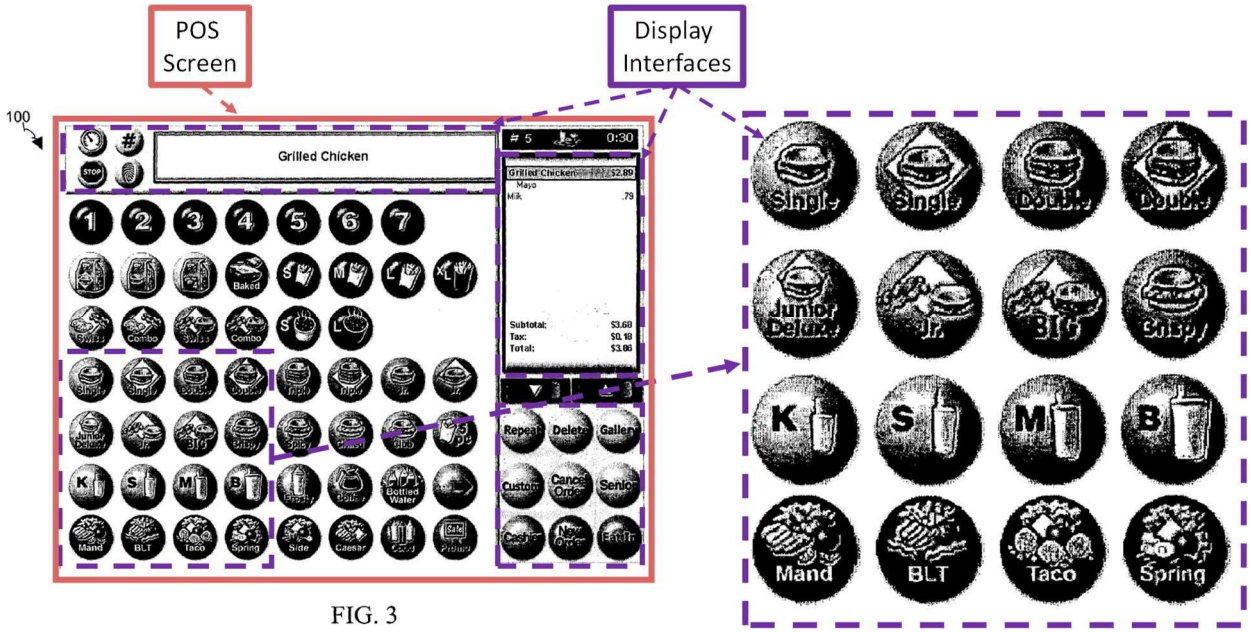


FIG. 3

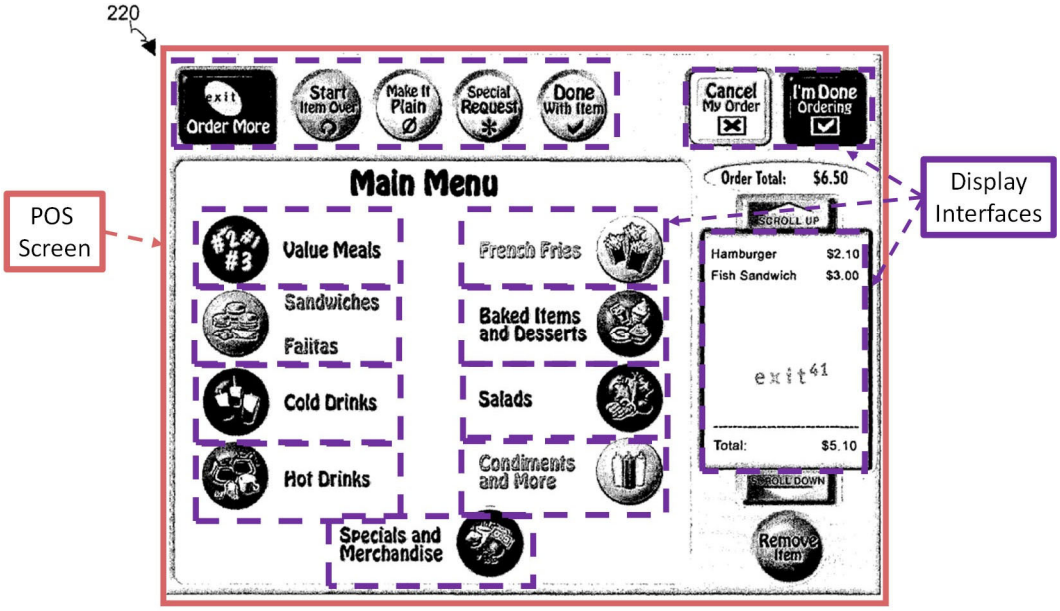


FIG. 9

A POSITA would have understood Tengler discloses and suggests manager edits to menus and items includes information indicative of at least one of a number, shape, and arrangement of display interfaces (buttons) appearing on POS screens

associated with such menus and items. For example, adding a new category to the main menu using the menu editor would modify the main menu screen to include an additional category button for selection. Likewise, adding a new beverage item to an existing category (e.g., "Cold Drinks") would modify the screen for such category to include an additional button associated with such item. Additionally, interface logic is implemented on the backend using GUI "design objects" in an object-oriented structure. EX1005, [0121]. A POSITA would have understood that object-oriented GUI design was common in the 2008 timeframe and discloses and suggests allowing a manager to customize the layout and functionality of design objects, e.g., buttons and other display interfaces.

218. Tengler thus discloses and suggests the server receives information from the POS builder interface, including information indicative of at least one of a number, shape, or arrangement of display interfaces on POS screens.

219. Accordingly, it is my opinion that Tengler discloses and suggests the limitations of claim 10.

H. Claim 12

"The web-based point of sale (POS) builder system of claim 1, wherein the one or more display interfaces are accessible on the POS builder interface"

220. As explained for claims 1[c] and 9-10, Tengler discloses buttons/keys on POS screens (display interfaces). EX1005, [0062], [0074]-[0076], Figs. 3, 8-10. Tengler's "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." EX1005, [0103]. For example, manager user interfaces such as 190, 192, and 196 (POS builder interface) allow managers to create/edit menus, items, and attributes that are displayed on POS screens associated with display interfaces (buttons/keys). Tengler thus suggests managers can access the buttons and other display interfaces on the POS builder interface to edit display interfaces on POS screens employees and customers use to place orders. Tengler further renders this claim obvious because a POSITA would have been motivated to enhance functionality and customization options for the user interface designer and would have had a reasonable expectation of success because Tengler includes GUI design objects and elements "for creating GUIs for client applications." *Id*., [0121]. For example, object orientation means that enhancing the UI is a simple matter of manipulating the views of the objects. In other words,

the POSITA would not have been tasked with defining objects and their relationships but would have had the simpler task of modifying the views of the extant objects.

221. Accordingly, it is my opinion that Tengler discloses and suggests the limitations of claim 12.

I. Claim 13

> ***"The web-based point of sale (POS) builder system of claim 1, wherein the at least one server is further configured to: receive, over the network from the POS builder interface, second information regarding a modification to at least one of the one or more POS screens; and update the at least one of the one or more POS screens on the one or more POS terminals based on the second information"***

222. As explained for 1[c], Tengler's server receives information over the network from managers using a POS builder interface to create/modify POS screens displayed on POS terminals, which is used to configure POS terminals to reflect those changes. EX1005, [0073], [0094], [0103], [0107], [0109], [0121]. "[M]anagers can … modify menus and change prices using interfaces 192 and 196" iteratively and/or as many times as desired. *Id.*, [0073]. Additional (second) information regarding a modification to a menu item or price is thus received when managers make such changes using the POS builder interface a second time after using it a first time. The system saves revised "specifications in the database 602"

and updates POS screens on POS terminals based on each set of revised specifications. EX1005, [0103], [0107]; *see also id.*, [0073], [0094]. Tengler thus discloses and suggests receiving second information to modify a POS screen, saving that information in the database at server 464, then updating the POS screens displayed on POS terminals to reflect those modifications.

223. Accordingly, it is my opinion that Tengler discloses and suggests the limitations of claim 12.

J. Claim 14

"The web-based point of sale (POS) builder system of claim 1, wherein the at least one server is further configured to store information regarding the one or more POS screens"

224. As explained for claim 13, Tengler's "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." EX1005, [0103]; *see also id.*, [0073]. Database 602 is part of main server 464. *Id.*, [0115]-[0116], Fig. 15:

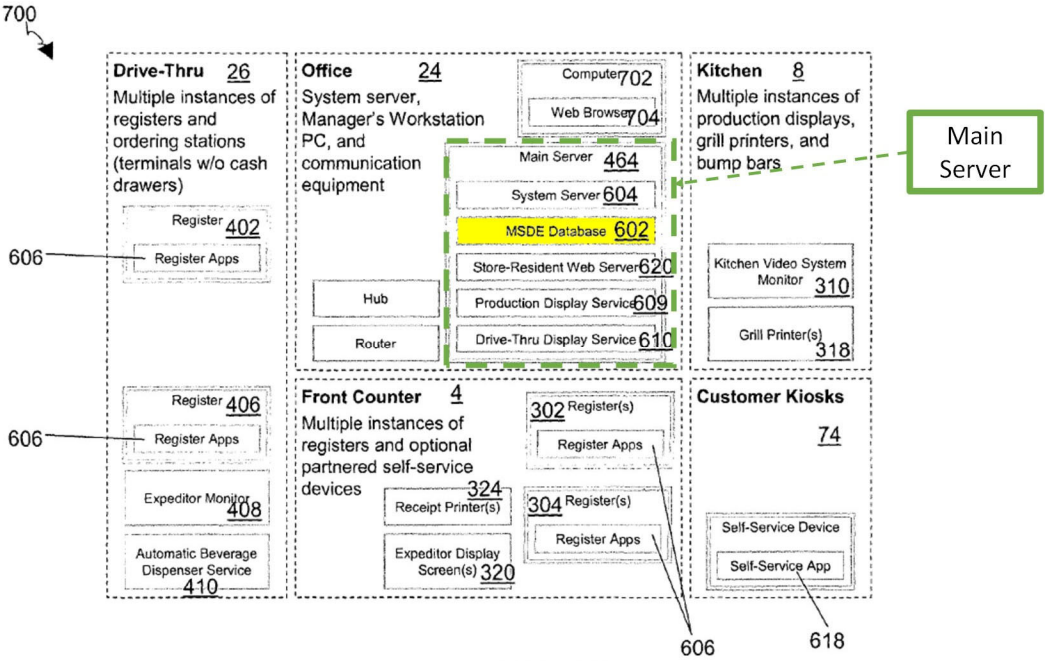


FIG. 15

Server 464 is thus configured to store information regarding POS screens.

225. Accordingly, it is my opinion that Tengler discloses and suggests the limitations of claim 14.

K. Claim 15

"The web-based point of sale (POS) builder system of claim 1, wherein the at least one server is located remotely from the one or more POS terminals"

226. As explained for 1[a]-1[b], Tengler's main server 464 is in "manager's office 24," which is in a different part of the restaurant than, and thus remote from, the order entry stations (POS terminals). EX1005, [0056], [0115]-[0116], Figs. 1, 15:

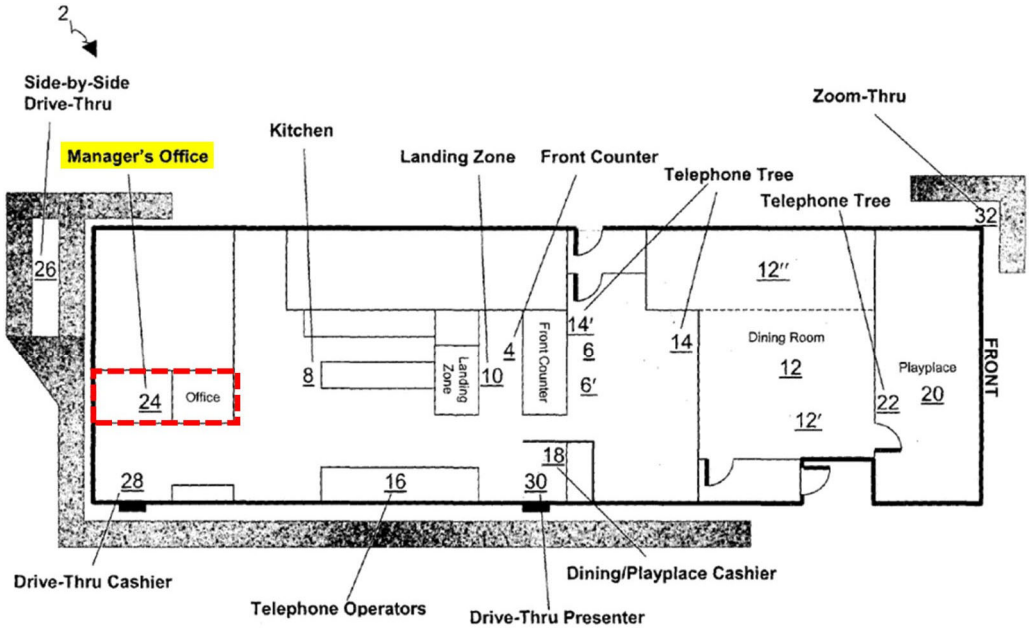


FIG. 1

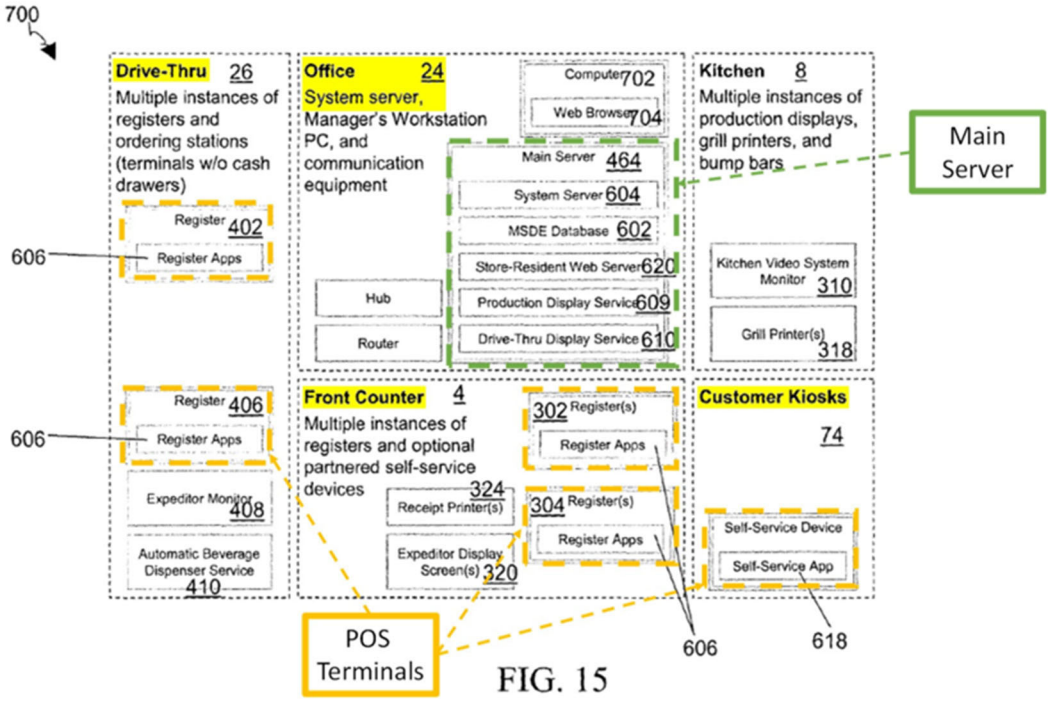


FIG. 15

227. Tengler also discloses a plurality of POS terminals at a remote call center running register applications 627 via proxies with RMS server 604 on main

server 464 where "telephone order takers at the call center use the same interface 100 as order takers in restaurant 2 to enter customer orders." *Id.*, [0114]; *see also id.*, [0016], [0025], [0112]-[0113]. Server 464 is located remotely from such POS terminals. *Id.*, [0028] (call center can "handl[e] calls from restaurants in different time zones").

228. Tengler also discloses and suggests that a central server can "provide[] features to manage a set of restaurants." EX1005, [0108]; *see also id.*, [0109]-[0111]. Tengler thus discloses and suggests that a server located remotely from the restaurant could be used to provide "enterprise management of multiple restaurants." *Id.*, [0108]. It would be obvious that such a server could be used to perform the same functions as server 464 regarding creating/modifying POS screens to allow management to efficiently make enterprise changes for multiple store locations on a single server. *Id.*; *see also* § XIII.A.2 (1[a]-1[b]).

229. Accordingly, it is my opinion that Tengler discloses and suggests the limitations of claim 15.

L. Claim 16

> ***"The web-based point of sale (POS) builder system of claim 1, wherein the at least one server is further configured to receive the information for creating or modifying the one or more POS screens and create or modify the one or more POS***

screens in real time while the one or more POS terminals are in use performing one or more POS transactions"

230. As discussed in 1[c], "user interface designer 614," remotely accessible via web server 620 of main server 464, "allows management to edit the user interface" of client applications and save modified "specifications to database 602." EX1005, [0103]; *see also id.*, [0107], [0073]. Tengler thus suggests managers can make changes in real time during their shifts while POS terminals are in use for POS transactions. Tengler makes no mention of intervening processes or batch processing that might suggest something other than the ability to make changes in real time. Tengler also discloses real-time communication with server 464 and its components, e.g., RMS server 604 and database 602. *Id.*, [0055]-[0056], [0089], [0094]-[0095], [0107], [0112], [0115]-[0116], Fig. 15; *see also id.*, [0121], [0126]-[0127]. Tengler thus discloses and suggests server 464 receives information from the POS builder interface and modifies POS screens in real time while POS terminals are in use performing transactions, then provides modified specifications to POS terminals after the manager saves changes to database 602.

231. Accordingly, it is my opinion that Tengler discloses and suggests the limitations of claim 16.

M. Claim 17

"The web-based point of sale (POS) builder system of claim 1, wherein the one or more POS terminals use the one or more POS screens after completing a pending POS transaction"

232. As explained for 1[c], 1[e], and claim 16, Tengler discloses managers can edit/modify previously created POS screens. EX1005, [0073], [0103], [0107]. For example, "managers can also modify menus and change prices using interfaces 192 and 196." *Id.*, [0073]. A POSITA would have been motivated to ensure that manager changes are not immediately propagated to terminals in use for a pending transaction, e.g., to avoid customer and employee dissatisfaction with changing menu items, options, and prices during a pending transaction. A POSITA would also have been motivated to maintain data coherence and consistency by applying POS screen modifications after a pending transaction is complete. For example, if a restaurant changes its lunch menu to lower the price but remove the soup starter during a transaction, a user entering multiple meals on a single ticket might end up with some meals having soup at a higher price and others without. The user might have no clear way to conform those meals to either the new or old menu. To avoid this scenario and the possibility of other data coherence/consistency issues in the database, a POSITA would have been motivated to complete any open transaction

before implementing POS screen modifications. Such POS terminals would then display modified POS screens for subsequent transactions.

233. Accordingly, it is my opinion that Tengler discloses and suggests the limitations of claim 17.

N. Claims 18

> ***"The web-based point of sale (POS) builder system of claim 1, wherein the at least one server is further configured to maintain*** *information* ***regarding POS screens for separate sets of POS terminals separately"***

and 20

> ***"The web-based point of sale (POS) builder system of claim 1, wherein the at least one server is further configured to maintain information regarding the one or more POS screens"***

234. As explained for 1[b] and 16, Tengler discloses database 602 of server 464 maintains information regarding POS screens (claim 20). "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." EX1005, [0103]. The register application displays POS screens for employees at order terminals 52, and the self-service application displays POS screens for customers at self-service kiosks 74. EX1005, [0062], [0074]-[0076], Fig. 2; *see also id.*, [0026], [0032]-[0033]; Figs. 3-4 (register application at terminals 52):



FIG. 3



FIG. 4

Figs. 8-9 (self-service application at customer kiosks 74):

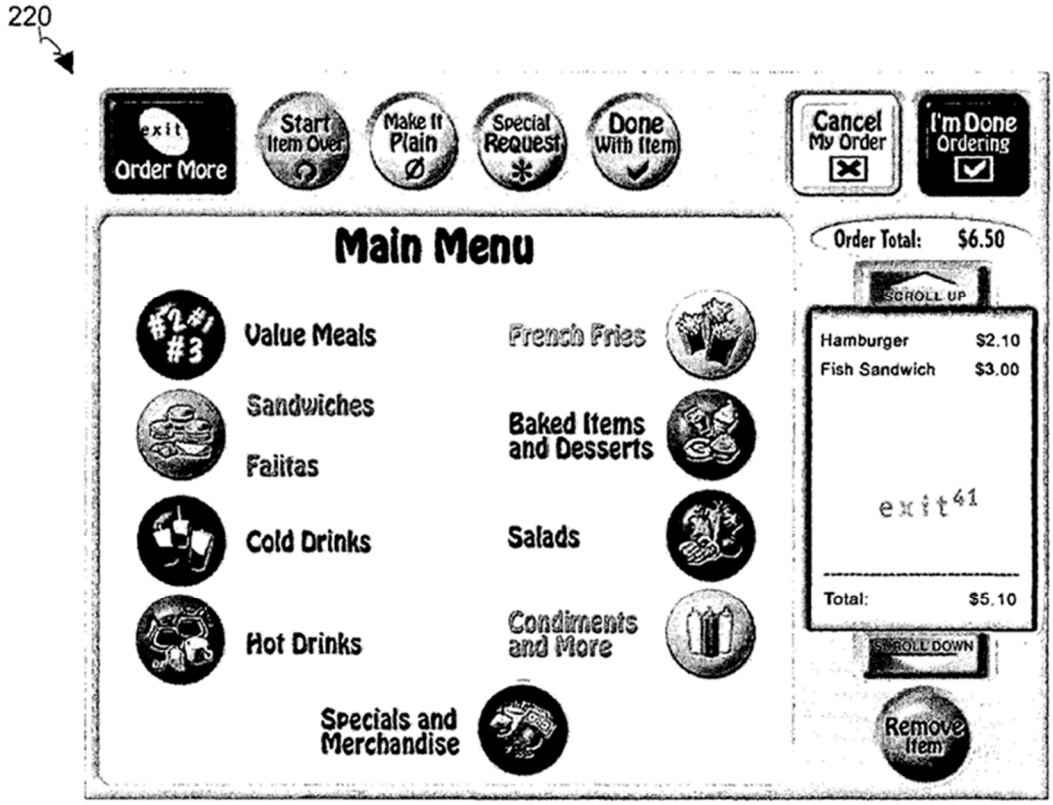


FIG. 9

235. A POSITA would have understood employee terminals and self-service terminals are separate sets of POS terminals and that database 602 would store information regarding POS screens for each set of POS terminals separately to ensure each respective terminal displays appropriate screens. *Id.*, [0073], [0103], [0107].

236. Accordingly, it is my opinion that Tengler discloses and suggests the limitations of claims 19 and 20.

O. Claim 19

"The web-based point of sale (POS) builder system of claim 1, wherein instructions to the POS builder interface for programmatic creation and modification of the POS terminals are not formatted in programming code"

237. As explained in 1[c], Tengler discloses a POS builder interface for accessing user interface designer for programmatic creation/modification of POS screens. Manager instructions to the POS builder interface are made using the GUI and are thus not formatted in programming code because the manager is not entering programming code into the interface. For example, managers "modify menus and change prices using interfaces 192 and 196" without access to programming code. EX1005, [0073]; *see also id.*, [0103], [0108]-[0109], Figs. 7A and 7B (excerpted):

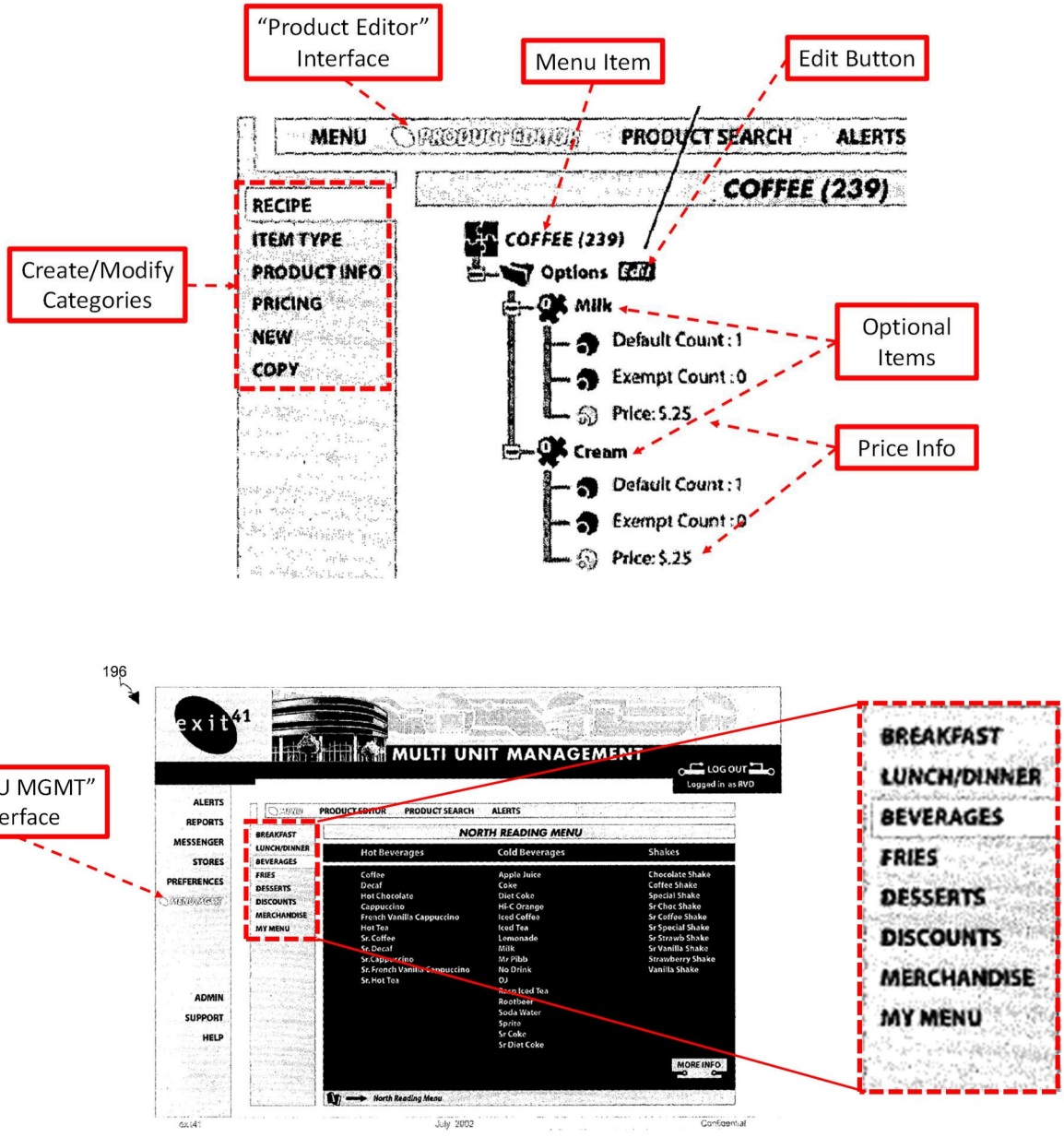


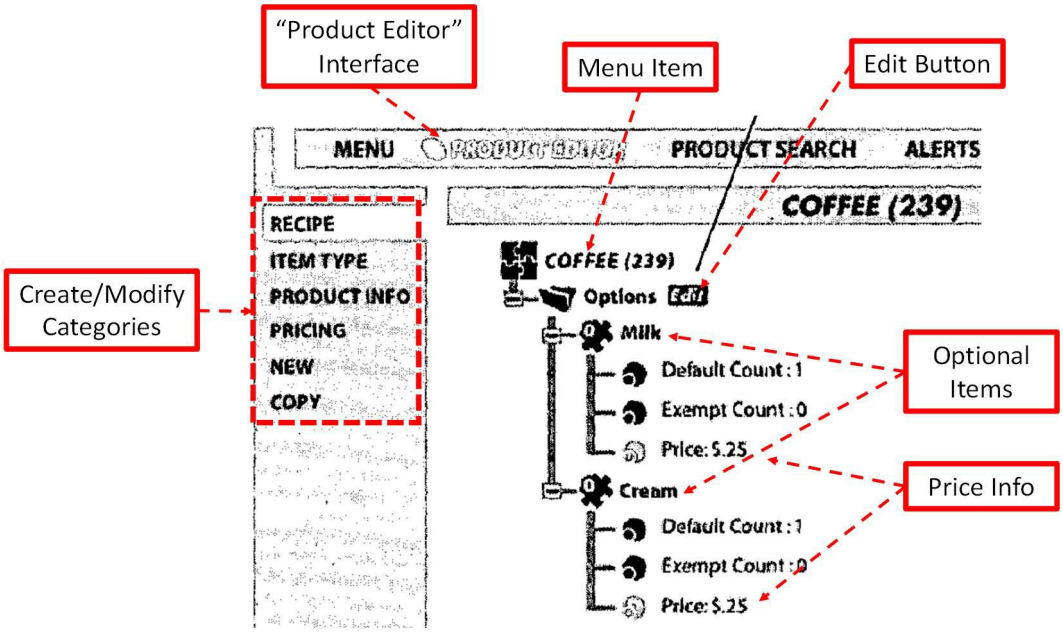
FIG. 7B

238. Accordingly, it is my opinion that Tengler discloses and suggests the limitations of claim 19.

P. Claim 21

"The web-based point of sale (POS) builder system of claim 1, wherein the received information comprises one or more attributes of the one or more items"

239. As explained in 1[c], "managers can … modify menus and change prices using interfaces 192 and 196." *Id.*, [0073]. The "product editor" screen of interface 192 lists various item attributes that can be modified/edited (including recipe, item type, product info, and pricing) and options for adding new products and copying existing products. *Id.*, [0073], Fig. 7A (excerpted):



Tengler thus discloses and suggests the server receives information comprising attributes (options, pricing) of menu items for display on POS screens of POS terminals.

240. Accordingly, it is my opinion that Tengler discloses and suggests the limitations of claim 21.

Q. Claim 22

"The web-based point of sale (POS) builder system of claim 1, wherein the POS terminals are configured to perform transactions independently of a connection with the network"

241. Tengler discloses POS terminals comprise client application 606 (such as register or self-service application) and database 607, such "that each of the stations operates using the cached copies of the data if the station is disconnected from the network." EX1005, [0011]; *see also id.*, [0090], Claim 6. This allows POS terminals to "run[] in a standalone mode by operating on stored or cached data if any of the cashier stations become disconnected from network 462." *Id.*, [0090]. POS terminals may thus perform transactions independently of a connection to the network.

242. Accordingly, it is my opinion that Tengler discloses and suggests the limitations of claim 22.

R. Claim 23

"The web-based point of sale (POS) builder system of claim 1, wherein the input interface element comprises a data

interface for inputting at least some of said further information"

243. As explained for 1[c] and claim 9, Tengler discloses numerous display interfaces for inputting further information (data interfaces) associated with ordering items in a restaurant (selecting/removing items, options, etc.). EX1005, [0003], [0062], [0076], Figs. 3-4, 8-10:

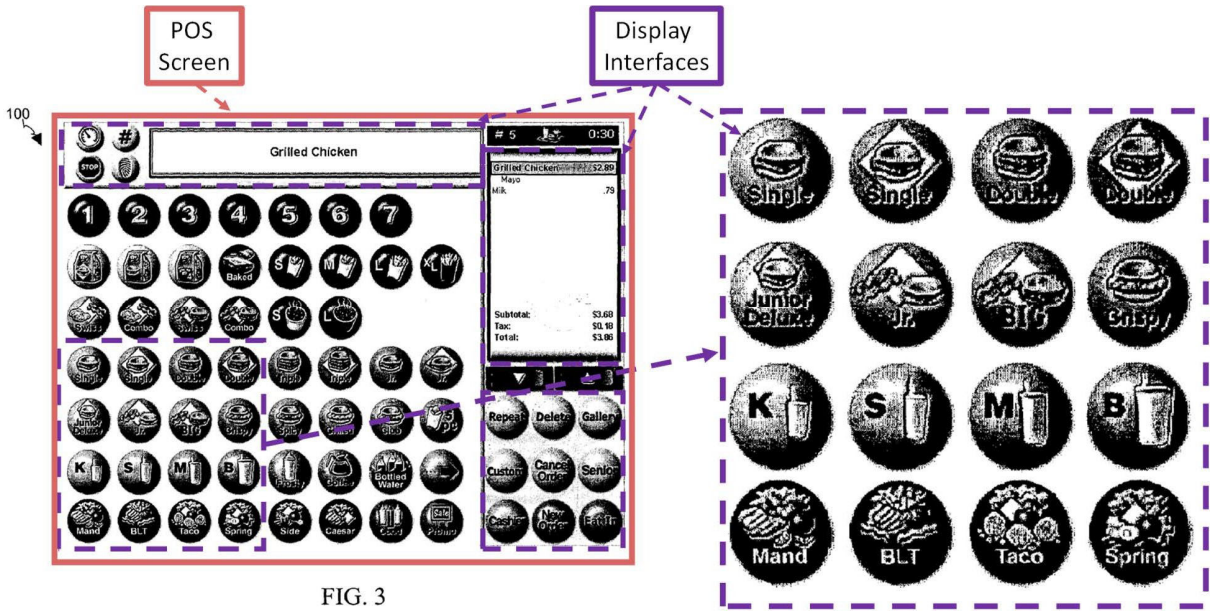


FIG. 3

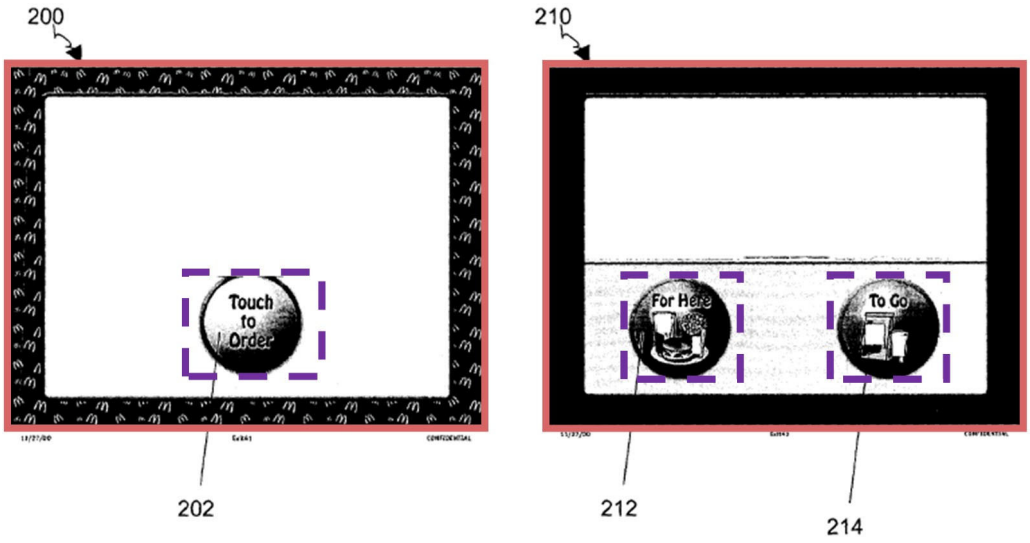


FIG. 8

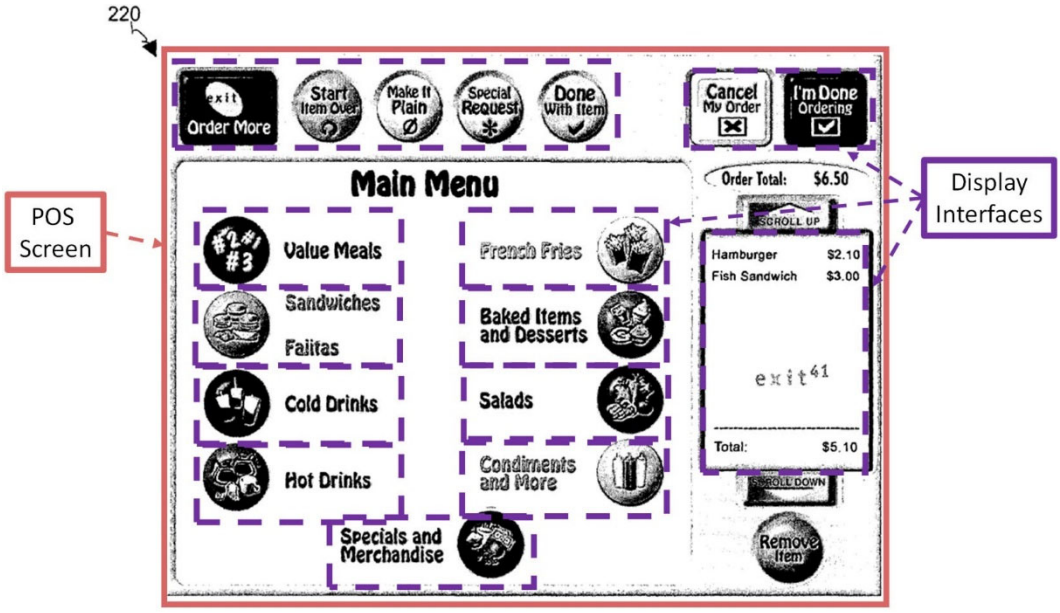


FIG. 9

A POSITA would have understood the ticket display interface (showing the current total and list of selected items) to be a running list that is updated as items are added, removed, or customized. In addition, the "repeat" button of Figure 3 would only be activated after an item were added to the running list. In another example,

the "senior" button of Figure 3 would be expected to apply a discounted pricing schedule to menu items. One or more of these inputs from the POS terminal represents further information input into the POS terminal, which is transmitted to server 464.

244. Additionally, a POSITA would be motivated to allow managers to specify a display interface element allowing text entry, e.g., to allow a customer to make special requests for further item customization.

245. Accordingly, it is my opinion that Tengler discloses and suggests the limitations of claim 23.

S. Claim 24

> ***"The web-based point of sale (POS) builder system of claim 1, wherein to configure the one or more POS terminals comprises dynamically configuring the one or more POS terminals specific for the corresponding customer based on the one or more transactions by the corresponding customer"***

246. As explained for 1[d] and 1[e], Tengler discloses "receiving an electronic image of a customer, and associating the electronic image with an order of a customer," which "is captured at the ordering location" and stored in the database. EX1005, [0017]-[0018]; *see also id.*, [0055]. Images can be displayed on the same terminals used to enter orders and take payment, e.g., when ordering from a drive- through cashier. *Id.*, [0058], [0062], [0065], [0077] ("drive-through cashier

payment area 28 includes … order entry"), [0088]. POS screens are modified based on the customer's image, so orders, including subsequent orders, can be delivered to the right person. *Id.* Tengler thus discloses and suggests dynamically configuring POS terminals specific to a corresponding customer based on information about the customer's transactions received from POS terminals and stored in the database at server 464.

247. Accordingly, it is my opinion that Tengler discloses and suggests the limitations of claim 24.

T. Claims 25

"The web-based point of sale (POS) builder system of claim 1, wherein the POS builder interface is configured to create or modify at least one of: a position or operation of a first display interface of the one or more display interfaces, wherein the first display interface comprises an input interface element"

and 26

"The web-based point of sale (POS) builder system of claim 25, wherein the input interface element comprises a touch screen input interface element"

248. Tengler's "user interface designer 614 allows management to edit the user interface of register and self-service applications and also saves the specifications in the database 602." EX1005, [0103]; *see also id.*, [0062], [0074]-[0076], Figs. 3-4, 8-10:

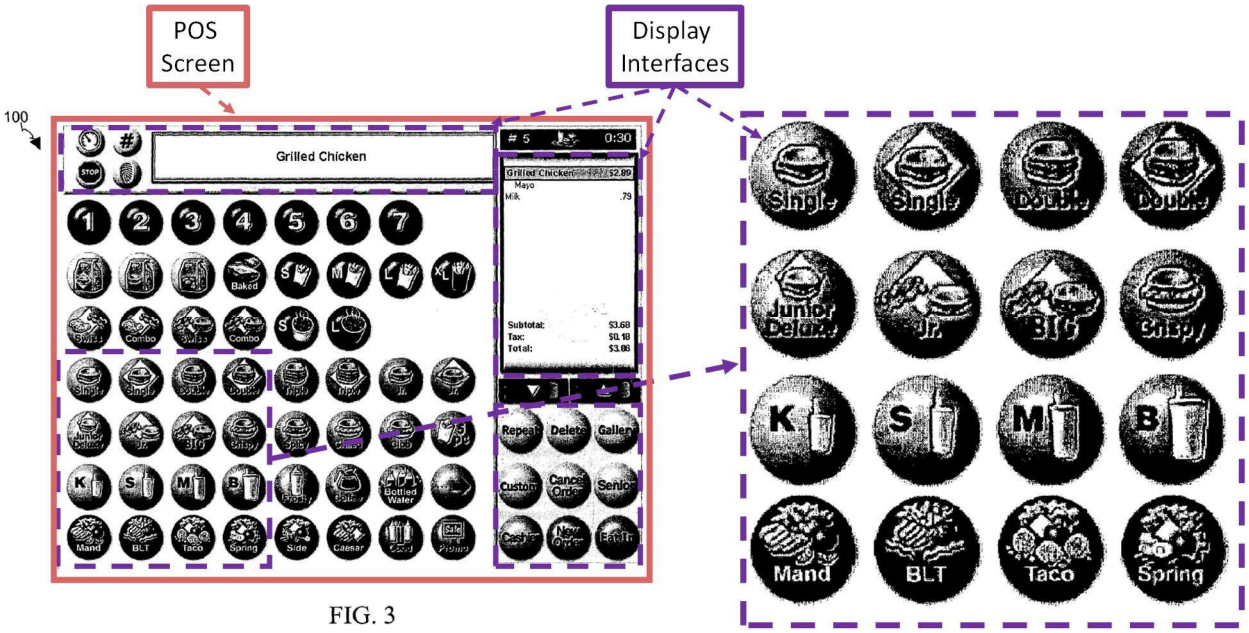


FIG. 3

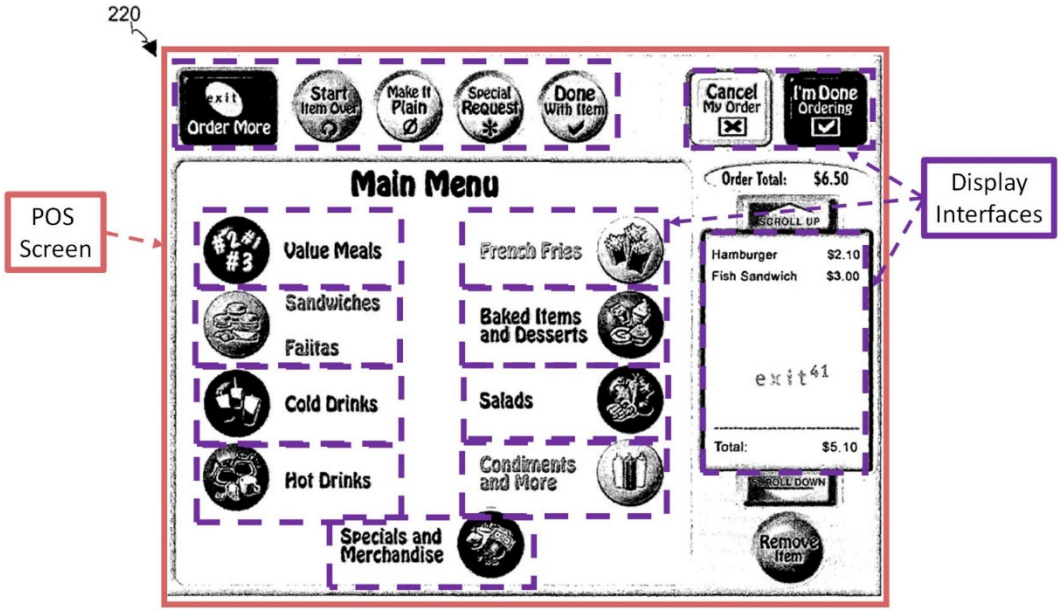


FIG. 9

249. As explained for 1[c], new and existing items, and their associated buttons (display interfaces), are created/modified using the POS builder interface. *Id.*, [0073], [0103], [0107], [0109]; Figs. 7, 7A-7B. As explained for claim 10, Tengler discloses and suggests the server receives specifications/information from

managers to create/modify POS screens displayed on POS terminals, including at least one of a number, shape, and arrangement of buttons (including their position). Further, because changing the item/option associated with a button on a POS screen (e.g., if a new item was added to replace an existing item) would modify the operation of a POS terminal by altering how it responds when that button (area) of the screen is selected, Tengler discloses and suggests creating/modifying the operation of buttons. *Id.* Because the display interfaces are buttons for inputting data/information on a touchscreen, they are themselves and/or include an input interface element. EX1005, [0075].

250. Accordingly, it is my opinion that Tengler discloses and suggests the limitations of claim 25 and 26.

U. Claim 27

251. Claim 27 is similar to claim 1, but drafted from the perspective of a POS terminal rather than a server. Similar elements are obvious for the same reasons as in claim 1 and as explained below.

1. 27[pre]

"A web-based point of sale (POS) builder system comprising"

252. *See* 1[pre], § XIII.A.1.

253. Accordingly, it is my opinion that Tengler discloses and suggests the limitations of claim 27[pre].

2. 27[a]

"at least one POS terminal configured to: display one or more POS screens"

and 27[b]

"communicate with at least one server over a network comprising the Internet"

254. *See* 1[a]-1[b], § XIII.A.2.

255. Accordingly, it is my opinion that Tengler discloses and suggests the limitations of claim 27[a] and 27[b].

3. 27[c]

"receive, over the network from the at least one server, 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"

256. This limitation is similar to 1[c], it but recites a POS terminal receiving information from the server rather than a server receiving information from a POS builder interface. That difference is addressed below. The remaining limitations of 27[c] are obvious for the same reasons explained for 1[c].

257. As explained in 1[c], Tengler's POS builder resides on main server 464 and is accessible remotely over the Internet via a browser and web server 620.

EX1005, [0073], [0103], [0107], [0109], [0121]; Figs. 7, 7A-7B. Server 464 includes web server software 620 and database 602. *Id.*, [0115]-[0116], Fig. 15. "The store resident web server 620 also allows managers to edit the user interface of the register [606] and self-service applications [618] and also saves the specifications in the database 602." *Id.*, [0103]; *see also id.*, [0073], [0107]. POS terminals running register application 606 and self-service application 618 display their user interfaces (POS screens) to employees and customers. *Id.*, [0103], [0107], Fig. 15. POS terminals "contact the restaurant's RMS server 604 … to display the appropriate user interface, so that the order taker can enter the customer's order." *Id.*, [0112]. Further, as explained for claim 22, Tengler's POS terminals are configured to operate independent of a network connection. *Id.*, [0011], [0090]. Tengler thus discloses and suggests POS screen specifications, including information for creating/modifying buttons associated with items, are received by the POS terminals from database 602, which is part of server 464, over the network to display POS screens and process orders.

258. Accordingly, it is my opinion that Tengler discloses and suggests the limitations of claim 27[c].

4. 27[d]

"perform one or more transactions with respect to the one or more items"

259. As explained for 1[a]-1[b], Tengler's POS terminals "enabl[e] users to enter orders" and process payment. EX1005, [0011]; *see also id.*, [0055], [0062], [0074]-[0076], Figs. 3-4, 8-10. Tengler's POS terminals thus perform transactions regarding food items ordered in a restaurant.

260. Accordingly, it is my opinion that Tengler discloses and suggests the limitations of claim 27[d].

5. 27[e]

"transmit, 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"

and 27[h]

"wherein said one or more POS transactions relate to one or more transactions by corresponding customers respectively associated with at least one of said one or more POS terminals"

261. *See* 1[d], 1[g], § XIII.A.4.

262. As explained in 1[d], the further information regarding POS transactions relates to transactions by customers to order food items, either from cashier order terminals 52 or themselves via self-service kiosks 74. EX1005, [0011], [0055], [0062], [0074]-[0076]. The information is transmitted from the

POS terminals to main server 464 and stored in its database 602. *Id.*, [0049]-[0051], [0055], [0058], [0089], [0116], [0121]; *see also id.*, [0125], Fig. 17.

263. Accordingly, it is my opinion that Tengler discloses and suggests the limitations of claim 27[e] and 27[h].

6. 27[f]

"create or modify based on the received information or further information the one or more POS screens"

264. *See* 1[e], § XIII.A.5. Additionally, 27[f] allows a POS terminal to create or modify POS screens "based on the received information ***or*** further information." As explained for 1[e], Tengler discloses POS terminals configured to create and modify POS screens based on information provided from the POS builder interface ***and*** based on further information about POS transactions transmitted from POS terminals to the central server.

265. Accordingly, it is my opinion that Tengler discloses and suggests the limitations of claim 27[f].

7. 27[g]

"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"

266. *See* 1[f], § XIII.A.6.

267. Accordingly, it is my opinion that Tengler discloses and suggests the limitations of claim 27[g].

V. Claims 28, 31-38, 40-41

268. Claims 28, 31-38, and 40-41 recite nearly identical limitations to those depending from claim 1 and are obvious for the same reasons. The following chart identifies the corresponding claims with additional argument/support to address differences.

Limitation	Discussion Incorporated by Reference
Claim 28	Claim 3, § XIII.A.C
Claim 31	Claim 9, § XIII.A.D
Claim 32	Claim 10, § XIII.A.G
Claim 33	Claim 15, § XIII.A.K
Claim 34	Claim 16, § XIII.A.L Claim 34 is like claim 16 but from the perspective of the POS terminal rather than the server. As explained for claim 16, Tengler discloses and suggests managers create/modify POS screens on the server in real time while POS terminals are in use performing transactions in real time as the customer is placing the order.
Claim 35	Claim 17, § XIII.A.M

Limitation	Discussion Incorporated by Reference
Claim 36	Claim 21, § XIII.A.P
Claim 37	Claim 4, § XIII.A.D
Claim 38	Claim 22, § XIII.A.Q Claim 38 is like claim 22 but adds additional limitations regarding transmitting the information regarding POS transactions to the server after the transaction is complete and the POS terminal establishes a connection to the server. Tengler's main server 464 includes "web server software 620" that provides remote access to the POS builder via the Internet, and it is thus also a web server. EX1005, [0073], [0103], [0109], [0116]. As explained for claim 22, Tengler's POS terminals operate in standalone mode (disconnected from the network) and then later establish a connection to the network and upload the transaction data to database 602 that "maintains a complete history of orders." *Id.*, [0011], [0089]-[0090]. That discloses and suggests performing transactions without a connection to the server.
Claims 40-41	Claim 25-26, § XIII.A.T

269. Accordingly, it is my opinion that Tengler discloses and suggests the limitations of claims 28, 31-38, and 40-41.

W. Claim 39

"The web-based point of sale (POS) builder system of claim 38, wherein the at least one POS terminal is further configured to store locally the information regarding the one or more transactions"

270. As explained for claims 22 and 38, Tengler's POS terminals can perform transactions, store transaction information locally, and then establish a connection to main server 464 and upload the transaction data to database 602. EX1005, [0011], [0089]-[0090].

271. Accordingly, it is my opinion that Tengler discloses and suggests the limitations of claim 39.

X. Claim 42

272. Claim 42's method steps are similar to the functions of claim 1's system, but omit the server limitation (hence the numbering discrepancy). Claim 42 is obvious for the same reasons. The following chart identifies the corresponding limitations.

Limitation	Discussion Incorporated by Reference
42[pre]	1[pre], § XIII.A.1.
42[a]	1[a]-1[b], § XIII.A.2.
42[b]	1[c], § XIII.A.3.
42[c], 42[f]	1[d], 1[g], § XIII.A.4.
42[d]	1[e], § XIII.A.5.

Limitation	Discussion Incorporated by Reference
42[e]	1[f], § XIII.A.6.

273. Accordingly, it is my opinion that Tengler discloses and suggests the limitations of claim 42.

Y. Claim 43

274. Claim 43's method steps are nearly identical to the functions of claim 27's system. The following chart identifies the corresponding limitations of claim 27, which in turn reference claim 1 for several limitations (parallel references to claim 1 are provide for convenience).

Limitation	Discussion Incorporated by Reference
43[pre]	27[pre], § XIII.U.1 (1[pre], § XIII.A.1)
43[a]-43[b]	27[a]-27[b], § XIII.U.2 (1[a]-1[b], § XIII.A.2)
43[c]	27[c], § XIII.U.3 (1[c], § XIII.A.3)
43[d]	27[d], § XIII.U.4
43[e], 43[h]	27[e], 27[h], § XIII.U.5 (1[d], 1[g], § XIII.A.4)

Limitation	Discussion Incorporated by Reference
43[f]	27[f], § XIII.U.6 (1[e], § XIII.A.5) As explained for 1[e] and 27[f], Tengler discloses POS terminals configured to create and modify POS screens based on information provided by a manager using the POS builder interface ("received information" of 43[f]) ***and*** based on further information about POS transactions transmitted from POS terminals to the central server ("transmitted information" of 43[e]).
43[g]	27[g], § XIII.U.7 (1[f], § XIII.A.6)

275. Accordingly, it is my opinion that Tengler discloses and suggests the limitations of claim 43.

Z. Claim 44

276. Claim 44 recites similar limitations to claim 1, but more broadly claims "creating or modifying *functionality of the one or POS terminals*" (44[c]). Tengler's disclosure regarding creating and modifying a series of POS screens and items/buttons displayed on POS terminals/kiosks renders obvious the broader limitations regarding functionality for the same reasons as explained for claim 1. The following chart identifies the corresponding limitations.

Limitation	Discussion Incorporated by Reference
44[pre]	1[pre], § XIII.A.1.

Limitation	Discussion Incorporated by Reference
44[a]-44[b]	1[a]-1[b], § XIII.A.2.
44[c]	1[c], § XIII.A.3.
44[d], 44[g]	1[d], 1[g], § XIII.A.4.
44[e]	1[e], § XIII.A.5.
44[f]	1[f], § XIII.A.6.

277. Accordingly, it is my opinion that Tengler discloses and suggests the limitations of claim 44.

XIII. CONCLUSION

278. For at least the above reasons, it is my opinion that all claims of the '793 patent would have been obvious to a person of ordinary skill in the art.

XIV. AVAILABILITY FOR CROSS-EXAMINATION

279. In signing this declaration, I recognize that this declaration will be filed as evidence in a contested case before the Patent Trial and Appeal Board of the United States Patent and Trademark Office. I also recognize that I may be subject to cross-examination in the case and that cross-examination will take place within the United States. If cross-examination is required of me, I will appear for cross-examination within the United States during the time allotted for cross-examination.

XV. RIGHT TO SUPPLEMENT

280. I reserve the right to supplement my opinions in the future to respond to any arguments that the Patent Owner raises and to take into account new information as it becomes available to me.

XVI. ACKNOWLEDGMENT

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

Dated:



Dr. Todd Mowry

TABLE OF EXHIBITS

Exhibit Number	Description
1001	U.S. Patent No. 11,226,793 to Baratta et al. ("the '793 patent")
1002	Expert Declaration of Todd Mowry
1003	Prosecution History of the '793 patent
1004	U.S. Patent Publication No. 2007/0265935 ("Woycik")
1005	U.S. Patent Publication No. 2005/0049921 ("Tengler")
1006	U.S. Patent Publication No. 2008/0208696 ("Olson")
1007	U.S. Patent Publication No. 2004/0181454 ("Manno")
1008	U.S. Patent Publication No. 2006/0235755 ("Mueller")
1009	U.S. Patent Publication No. 2004/0128199 ("Cusack")
1010	United Kingdom Patent Application GB2369201 ("Tomlinson")
1011	U.S. Pat. No. 5,818,428 ("Eisenbrandt")
1012	Plaintiff's Response in Opposition to Defendant's Motion for Summary Judgment, *CloudofChange, LLC v. NCR Corporation*, 6-19-cv-00513 (W.D. Tex.)
1013	WO 02/39226 A2 ("Thompson")
1014	Plaintiff's Motion in *Limine*, *CloudofChange, LLC v. NCR Corporation*, 6-19-cv-00513 (W.D. Tex.)
1015	U.S. Patent Publication No. 2004/0103037 ("Wetmore")
1016	RESERVED

Exhibit Number	Description
1017	Plaintiff's First Amended Preliminary Infringement Contentions Exhibit A3, *CloudofChange, LLC v. Lightspeed POS Inc.*, 6-21-cv-01102 (W.D. Tex.)
1018	First Amended Scheduling Order, *CloudofChange, LLC v. Lightspeed POS Inc.*, 6-21-cv-01102, D.I. 31 (W.D. Tex.)
1019	U.S. Patent Publication No. 2002/0194074 ("Jacobs")
1020	U.S. Patent No. 7,051,091 ("Cohen")
1021	U.S. Patent No. 6,384,850 ("McNally")
1022	U.S. Patent No. 6,002,395 ("Wagner")
1023	RESERVED
1024	Final Claim Constructions of the Court, *CloudofChange, LLC v. NCR Corporation*, 6-19-cv-00513 (W.D. Tex.)
1025	Supplemental Claim Construction Order, *CloudofChange, LLC v. NCR Corporation*, 6-19-cv-00513 (W.D. Tex.)
1026	U.S. Patent No. 6,247,032 ("Bernardo")
1027	Plaintiff's Preliminary Infringement Contentions for the '793 patent, *CloudofChange, LLC v. Clover Network, Inc.*, 6-22-cv-00634 (W.D. Tex.)
1028	Scheduling Order, *CloudofChange, LLC v. Clover Network, Inc.*, 6-22-cv-00634 (W.D. Tex., Oct. 24, 2022)
1029	Final Claim Constructions of the Court, *CloudofChange, LLC v. Lightspeed POS Inc.*, 6-21-cv-01102 (W.D. Tex., Sep. 16, 2022)

APPENDIX A

Todd C. Mowry

Curriculum Vitae

November, 2022

Contact Information

Computer Science Department
Carnegie Mellon University
5000 Forbes Avenue
Pittsburgh, PA 15213

Phone: (412) 268-3725
Fax: (412) 268-5576
Email: tcm@cs.cmu.edu
URL: http://www.cs.cmu.edu/~tcm

Education

Ph.D. in Electrical Engineering, Stanford University, March 1994.

- Thesis: "Tolerating Latency Through Software-Controlled Data Prefetching."
- Supervisors: Anoop Gupta and Monica Lam.

M.S. in Electrical Engineering, Stanford University, June 1989.

B.S. in Electrical Engineering with Highest Distinction, University of Virginia, May 1988.

Academic Appointments

7/08–Present: Professor, Computer Science Department (with a courtesy appointment in the Department of Electrical and Computer Engineering), Carnegie Mellon University.

8/09–6/10: Associate Department Head for Faculty, Computer Science Department

6/15–Present: Director of the Ph.D. program, Computer Science Department

7/97–6/08: Associate Professor, Computer Science Department (with a courtesy appointment in the Department of Electrical and Computer Engineering), Carnegie Mellon University.

12/93–6/97: Assistant Professor, Department of Electrical and Computer Engineering (with a courtesy appointment in the Department of Computer Science), University of Toronto.

9/89–11/93: Graduate Research Assistant, Stanford University.

Honors and Awards

- ASPLOS Influential Paper Award, 2022 (for "Design and Evaluation of a Compiler Algorithm for Prefetching," from ASPLOS 1992).
- Best Paper Runner-Up Award, VLDB conference, 2018.
- ACM Fellow, for contributions to software prefetching and thread-level speculation, 2016.
- SCS Doctoral Dissertation Award co-won by Angela Demke Brown (Ph.D. advisee), 2005.
- Most Thought-Provoking Idea Award (for Claytronics), the Wild and Crazy Idea Session IV held at ASPLOS-XI, 2004.
- Best Paper Award, the 20th International Conference on Data Engineering (ICDE), 2004.
- SCS Doctoral Dissertation Award co-won by J. Gregory Steffan (Ph.D. advisee), 2003.
- Runner-Up for Best Paper Award, the ACM SIGMOD Conference, 2001.
- Alfred P. Sloan Research Fellow, 1999-2001.
- TR100 Award (awarded by MIT's Technology Review magazine to the top 100 most promising young innovators in science and technology), 1999.
- Best Paper Award, the Second Symposium on Operating Systems Design and Implementation, 1996.

Doctoral Thesis Supervision

Student	Thesis Title (or Topic)	Graduation Date
Chi-Keung Luk	Optimizing the Cache Performance of Non-Numeric Applications. *(Nominated for the ACM Thesis Award by the University of Toronto Department of Computer Science.)*	January, 2000
J. Gregory Steffan	Hardware Support for Thread-Level Speculation. *(Co-winner of the SCS Doctoral Dissertation Award.)*	April, 2003
Antonia Zhai	Compiler Optimization of Value Communication for Thread-Level Speculation.	January, 2005
Angela Demke Brown	Explicit Compiler-based Memory Management for Out-of-core Applications. *(Co-winner of the SCS Doctoral Dissertation Award.)*	May, 2005
Christopher Colohan	Applying Thread-Level Speculation to Database Transactions.	November, 2005
Shimin Chen	Redesigning Database Systems in Light of CPU Cache Prefetching.	December, 2005
Amit Manjhi	Increasing the Scalability of Dynamic Web Applications	March, 2008
Olatunji Ruwase	Improving Device Driver Reliability Through Decoupled Dynamic Binary Analysis	May, 2013
Michelle Goodstein	Dataflow Analysis-Based Dynamic Parallel Monitoring	August, 2014
Vivek Seshadri	Simple DRAM and Virtual Memory Abstractions to Enable Highly Efficient Memory Subsystems	May, 2016
Gennady Pekhimenko	Practical Data Compression for Modern Memory Hierarchies	July, 2016
Chris Fallin	Finding and Exploiting Parallelism with Data-Structure-Aware Static and Dynamic Analysis	February, 2019
Prashanth Menon	On Building Robustness into Compilation-Based Main-Memory Database Query Engines	May, 2021
Pratik Fegade	A New Framework for Code Optimization	May, 2022 *(expected)*
Ziqi Wang	Improving Transactional Memory through Page Overlays	May, 2023 *(expected)*

Publications

Book Chapters

1. Amir Yazdanbakhsh, Gennady Pekhimenko, Hadi Esmaeilzadeh, Onur Mutlu, and Todd C. Mowry. Towards Breaking the Memory Bandwidth Wall Using Approximate Value Prediction. In *Approximate Circuits: Methodologies and CAD*, Springer International Publishing, 2019, pages 417–441.

Refereed Journal Articles

1. Prashanth Menon, Amadou Ngom, Lin Ma, Todd C. Mowry, and Andrew Pavlo. Permutable Compiled Queries: Dynamically Adapting Compiled Queries without Recompiling. In *Proceedings of the VLDB Endowment*, 14(2):101-113, October 2020.
2. Amir Yazdanbakhsh, Gennady Pekhimenko, Bradley Thwaites, Hadi Esmaeilzadeh, Onur Mutlu, and Todd C. Mowry. RFVP: Rollback-Free Value Prediction with Safe-to-Approximate Loads. In *ACM Transactions on Architecture and Code Optimization (TACO)*, 12(4):62, February 2016.
3. Amir Yazdanbakhsh, Gennady Pekhimenko, Bradley Thwaites, Hadi Esmaeilzadeh, Onur Mutlu, and Todd C. Mowry. Mitigating the Memory Bottleneck with Approximate Load Value Prediction. In *IEEE Design and Test*, 33(1), January 2016.
4. Vivek Seshadri, Samihan Yedkar, Hongyi Xin, Onur Mutlu, Phillip B. Gibbons, Michael A. Kozuch, and Todd C. Mowry. Mitigating Prefetcher-Caused Pollution Using Informed Caching Policies for Prefetched Blocks. In *ACM Transactions on Architecture and Code Optimization (TACO)*, 11(4), January 2015.
5. Shimin Chen, Phillip B. Gibbons, Michael Kozuch, and Todd C. Mowry. Log-Based Architectures: Using Multicore to Help Software Behave Correctly. In *Operating Systems Review*, 45(1), January 2011.
6. Shimin Chen, Michael Kozuch, Phillip B. Gibbons, Michael Ryan, Theodoros Strigkos, Todd C. Mowry, Olatunji Ruwase, Evangelos Vlachos, Babak Falsafi, and Vijaya Ramachandran. Flexible Hardware Acceleration for Instruction-Grain Lifeguards. In *IEEE Micro (Top Picks from 2008 Computer Architecture Conferences)*, 29(1), January 2009.
7. Seth C. Goldstein, Todd C. Mowry, Jason D. Campbell, Michael P. Ashley-Rollman, Michael De Rosa, Stanislav Funiak, James F. Hoburg, Mustafa E. Karagozler, Brian Kirby, Peter Lee, Padmananabhan Pillai, J. Robert Reid, Daniel D. Stancil, and Michael P. Weller. Beyond Audio and Video: Using Claytronics to Enable Pario. In *AI Magazine*, 30(2), March 2009.
8. Antonia Zhai, J. Gregory Steffan, Christopher B. Colohan, and Todd C. Mowry. Compiler and Hardware Support for Reducing the Synchronization of Speculative Threads. In *ACM Transactions on Architecture and Code Optimization (TACO)*, 5(1), May 2008.
9. Christopher B. Colohan, Anastassia Ailamaki, J. Gregory Steffan, and Todd C. Mowry. Incrementally parallelizing database transactions with thread-level speculation. In ACM Transactions on Computer Systems (TOCS), 26(1), February 2008.
10. Shimin Chen, Anastassia Ailamaki, Phillip B. Gibbons, and Todd C. Mowry. Improving Hash Join Performance through Prefetching. In *ACM Transactions on Database Systems*, 32(3):1-32, September 2007.
11. Christopher B. Colohan, Anastasia Ailamaki, J. Gregory Steffan, and Todd C. Mowry. CMP Support for Large and Dependent Speculative Threads. In *IEEE Transactions on Parallel and Distributed Systems*, 18(8):1041-1054, August 2007.
12. J. Gregory Steffan, Christopher B. Colohan, Antonia Zhai and Todd C. Mowry. The STAMPede Approach to Thread-Level Speculation. In *ACM Transactions on Computer Systems*, 23(3):253-300, August 2005.

13. Seth Copen Goldstein, Jason Campbell and Todd C. Mowry. Programmable Matter. In *IEEE Computer*, 38(6):99-101, June 2005.
14. Angela Demke Brown, Todd C. Mowry and Orran Krieger. Compiler-Based I/O Prefetching for Out-of-Core Applications. In *ACM Transactions on Computer Systems*, 19(2):111-170, May 2001.
15. Chi-Keung Luk and Todd C. Mowry. Architectural and Compiler Support for Effective Instruction Prefetching: A Cooperative Approach. In *ACM Transactions on Computer Systems*, 19(1):71-109, February 2001.
16. Todd C. Mowry and Chi-Keung Luk. Understanding Why Correlation Profiling Improves the Predictability of Data Cache Misses in Nonnumeric Applications. In *IEEE Transactions on Computers*, 49(4), April 2000.
17. Chi-Keung Luk and Todd C. Mowry. Automatic Compiler-Inserted Prefetching for Pointer-Based Applications. In *IEEE Transactions on Computers*, 48(2), February 1999.
18. Mark Horowitz, Margaret Martonosi, Todd C. Mowry, and Michael D. Smith. Informing Memory Operations: Memory Performance Feedback Mechanisms and their Applications. In *ACM Transactions on Computer Systems*, 16(2):170-205, May 1998.
19. Todd C. Mowry. Tolerating Latency in Multiprocessors through Compiler-Inserted Prefetching. In *ACM Transactions on Computer Systems*, 16(1):55-92, February 1998.
20. Todd Mowry and Anoop Gupta. Tolerating Latency Through Software-Controlled Prefetching in Shared-Memory Multiprocessors. In *Journal of Parallel and Distributed Computing*, 12(2):87-106, 1991.

Refereed Conference Papers

1. Pratik Fegade, Tianqi Chen, Phillip Gibbons, and Todd C. Mowry. The CoRa Tensor Compiler: Compilation for Ragged Tensors with Minimal Padding. To appear in *Proceedings of the Fifth Conference on Machine Learning and Systems (MLSys)*, August 2022.
2. Amadou Ngom, Prashanth Menon, Matthew Butrovich, Lin Ma, Wan Shen Lim, Todd C. Mowry, and Andrew Pavlo. Filter Representation in Vectorized Query Execution. In *Proceedings of the 17th International Workshop on Data Management on New Hardware (DaMoN)*, June 2021.
3. Ziqi Wang, Chul-Hwan Choo, Michael A. Kozuch, Todd C. Mowry, Gennady Pekhimenko, Vivek Seshadri, and Dimitrios Skarlatos. NVOverlay: Enabling Efficient and Scalable High-Frequency Snapshotting to NVM. In *Proceedings of the 48th Annual International Symposium on Computer Architecture (ISCA)*, June 2021.
4. Pratik Fegade, Tianqi Chen, Phillip Gibbons, and Todd C. Mowry. Cortex: A Compiler for Recursive Deep Learning Models. In *Proceedings of the Fourth Conference on Machine Learning and Systems (MLSys)*, April 2021.
5. Daming D. Chen, Phillip B. Gibbons, and Todd C. Mowry. TardisTM: Incremental Repair for Transactional Memory. In *Proceedings of the Eleventh International Workshop on Programming Models and Applications for Multicores and Manycores (PMAM '20)*, February 2020.
6. Ziqi Wang, Michael A. Kozuch, Todd C. Mowry, and Vivek Seshadri. Multiversioned Page Overlays: Enabling Faster Serializable Hardware Transactional Memory. In *Proceedings of the 28th International Conference on Parallel Architectures and Compilation Techniques (PACT)*, September 2019.
7. Vivek Seshadri, Donghyuk Lee, Thomas Mullins, Hasan Hassan, Amirali Boroumand, Jeremie Kim, Michael A. Kozuch, Onur Mutlu, Phillip B. Gibbons, and Todd C. Mowry. Ambit: in-memory accelerator for bulk bitwise operations using commodity DRAM technology. In *Proceedings of the 50th Annual International Symposium on Microarchitecture (MICRO-50)*, October 2017.

8. Prashanth Menon, Andrew Pavlo, and Todd C. Mowry. Relaxed Operator Fusion for In-Memory Databases: Making Compilation, Vectorization, and Prefetching Work Together At Last. In *Proceedings of VLDB 11(1): 1-13*, September 2017.

9. Andrew Pavlo, Gustavo Angulo, Joy Arulraj, Haibin Lin, Jiexi Lin, Lin Ma, Prashanth Menon, Todd Mowry, Matthew Perron, Ian Quah, Siddharth Santurkar, Anthony Tomasic, Skye Toor, Dana Van Aken, Ziqi Wang, Yingjun Wu, Ran Xian, and Tieying Zhang. Self-Driving Database Management Systems" In *Proceedings of the 2017 Conference on Innovative Data Systems Research (CIDR)*, January 2017.

10. Gennady Pekhimenko, Evgeny Bolotin, Nandita Vijaykumar, Onur Mutlu, Todd C. Mowry, and Stephen W. Keckler. A Case for Toggle-Aware Compression for GPU Systems. In *Proceedings of the 22nd International Symposium on High-Performance Computer Architecture (HPCA)*, March 2016.

11. Michelle L. Goodstein, Phillip B. Gibbons, Michael A. Kozuch, and Todd C. Mowry. Tracking and Reducing Uncertainty in Dataflow Analysis-Based Dynamic Parallel Monitoring. In *Proceedings of the 2015 International Conference on Parallel Architectures and Compilation Techniques (PACT '15)*, October 2015.

12. Nandita Vijaykumar, Gennady Pekhimenko, Adwait Jog, Abhishek Bhowmick, Rachata Ausavarungnirun, Chita Das, Mahmut Kandemir, Todd C. Mowry, and Onur Mutlu. A Case for Core-Assisted Bottleneck Acceleration in GPUs: Enabling Efficient Data Compression. In *Proceedings of the 42nd Annual International Symposium on Computer Architecture (ISCA)*, June 2015.

13. Vivek Seshadri, Gennady Pekhimenko, Olatunji Ruwase, Onur Mutlu, Phillip B. Gibbons, Michael A. Kozuch, Todd C. Mowry, and Trishul Chilimbi. Page Overlays: An Enhanced Virtual Memory Framework to Enable Fine-grained Memory Management. In *Proceedings of the 42nd Annual International Symposium on Computer Architecuture (ISCA)*, June 2015.

14. Gennady Pekhimenko, Tyler Huberty, Rui Cai, Onur Mutlu, Phillip P. Gibbons, Michael A. Kozuch, and Todd C. Mowry. Exploiting Compressed Block Size as an Indicator of Future Reuse. In *Proceedings of the 21st International Symposium on High-Performance Computer Architecture (HPCA)*, February 2015.

15. Vivek Seshadri, Abhishek Bhowmick, Onur Mutlu, Phillip B. Gibbons, Michael A. Kozuch, and Todd C. Mowry. The Dirty-Block Index. In *Proceedings of the 41st Annual International Symposium on Computer Architecuture (ISCA)*, June 2014.

16. Olatunji Ruwase, Michael A. Kozuch, Phillip B. Gibbons, and Todd C. Mowry. Guardrail: High Fidelity Correctness Checking of Device Drivers for Safeguarding I/O Operations. In *Proceedings of the Nineteenth International Conference on Architectural Support for Programming Languages and Operating Systems (ASPLOS 2014)*, March 2014.

17. Gennady Pekhimenko, Vivek Seshadri, Yoongu Kim, Hongyi Xin, Onur Mutlu, Phillip B. Gibbons, Michael A. Kozuch, and Todd C. Mowry. Linearly Compressed Pages: A Low-Complexity, Low-Latency Main Memory Compression Framework. In *Proceedings of the 46th Annual International Symposium on Microarchitecture (MICRO-46)*, December 2013.

18. Vivek Seshadri, Yoongu Kim, Chris Fallin, Donghyuk Lee, Rachata Ausavarungnirun, Gennady Pekhimenko, Yixin Luo, Onur Mutlu, Phillip B. Gibbons, Michael A. Kozuch, and Todd C. Mowry. RowClone: Fast and Energy-Efficient In-DRAM Bulk Data Copy and Initialization. In *Proceedings of the 46th Annual International Symposium on Microarchitecture (MICRO-46)*, December 2013.

19. Michelle L. Goodstein, Shimin Chen, Phillip B. Gibbons, Michael A. Kozuch, and Todd C. Mowry. Chrysalis Analysis: Incorporating Synchronization Arcs in Dataflow-Analysis-based Parallel Monitoring. In *Proceedings of the 21st International Conference on Parallel Architectures and Compilation Techniques (PACT-2012)*, September 2012.

20. Vivek Seshadri, Onur Mutlu, Todd C. Mowry, and Michael A. Kozuch. The Evicted-Address Filter: A Unified Mechanism to Address Both Cache Pollution and Thrashing. In *Proceedings of the 21st International Conference on Parallel Architectures and Compilation Techniques (PACT-2012)*, September 2012.

21. Gennady Pekhimenko, Vivek Seshadri, Onur Mutlu, Todd C. Mowry, Phillip B. Gibbons, and Michael A. Kozuch. Base-Delta-Immediate Compression: A Practical Data Compression Mechanism for On-Chip Caches. In *Proceedings of the 21st International Conference on Parallel Architectures and Compilation Techniques (PACT-2012)*, September 2012.

22. Olatunji Ruwase, Shimin Chen, Phillip Gibbons, and Todd C. Mowry. Decoupled Lifeguards: Enabling Path Optimizations for Dynamic Correctness Checking Tools. In *Proceedings of the ACM SIGPLAN 2010 Conference on Programming Language Design and Implementation (PLDI)*, June 2010.

23. F. Ryan Johnson, Radu Stoica, Anastasia Ailamaki, and Todd C. Mowry. Decoupling Contention Management from Scheduling. In *Proceedings of the Fifteenth International Conference on Architectural Support for Programming Languages and Operating Systems (ASPLOS 2010)*, March 2010.

24. Michelle Goodstein, Evangelos Vlachos, Shimin Chen, Phillip B. Gibbons, Michael Kozuch, and Todd C. Mowry. Butterfly Analysis: Adapting Dataflow Analysis to Dynamic Parallel Monitoring. In *Proceedings of the Fifteenth International Conference on Architectural Support for Programming Languages and Operating Systems (ASPLOS 2010)*, March 2010.

25. Evangelos Vlachos, Michelle Goodstein, Michael Kozuch, Shimin Chen, Babak Falsafi, Phillip B. Gibbons, and Todd C. Mowry. ParaLog: Enabling and Accelerating Online Parallel Monitoring of Multithreaded Applications. In *Proceedings of the Fifteenth International Conference on Architectural Support for Programming Languages and Operating Systems (ASPLOS 2010)*, March 2010.

26. Amit Manjhi, Charles Garrod, Bruce M. Maggs, Todd C. Mowry, Anthony Tomasic. Holistic Query Transformations for Dynamic Web Applications. In *Proceedings of the 2009 IEEE 25th International Conference on Data Engineering (ICDE)*, March-April 2009.

27. Daniel J. Dewey, Michael P. Ashley-Rollman, Michael DeRosa, Seth Copen Goldstein, Todd C. Mowry, Siddhartha S. Srinivasa, Padmanabhan Pillai, and Jason Campbell. Generalizing metamodules to simplify planning in modular robotic systems. In *Proceedings of the IEEE/RSJ 2008 International Conference on Intelligent Robots and Systems (IROS)*, September 2008.

28. Charles Garrod, Amit Manjhi, Anastasia Ailamaki, Bruce Maggs, Todd Mowry, Christopher Olston, and Anthony Tomasic. Scalable Query Result Caching for Web Applications. In *Proceedings of the 34th International Conference on Very Large Databases (VLDB)*, August 2008.

29. Lei Li, Wenjie Fu, Fan Guo, Todd C. Mowry, and Christos Faloutsos. Cut-and-stitch: efficient parallel learning of linear dynamical systems on SMPs. In *Proceedings of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD)*, August 2008.

30. Shimin Chen, Michael Kozuch, Theodoros Strigkos, Babak Falsafi, Phillip B. Gibbons, Todd C. Mowry, Michael Ryan, Olatunji Ruwase, and Evangelos Vlachos. Flexible Hardware Acceleration for Instruction-Grain Program Monitoring. In *Proceedings of the 35th Annual International Symposium on Computer Architecture (ISCA)*, June 2008.

31. Olatunji Ruwase, Phillip B. Gibbons, Todd C. Mowry, Vijaya Ramachandran, Shimin Chen, Michael Kozuch, and Michael Ryan. Parallelizing dynamic information flow tracking. In *Proceedings of the 20th Annual ACM Symposium on Parallel Algorithms and Architectures (SPAA)*, June 2008.

32. Brian Kirby, Burak Aksak, James Hoburg, Todd C. Mowry, and Padmanabhan Pillai. A Modular Robotic System Using Magnetic Force Effectors. In *Proceedings of the IEEE/RSJ 2007 International Conference on Intelligent Robots and Systems (IROS)*, October 2007.

33. Michael Ashley-Rollman, Seth Goldstein, Peter Lee, Todd C. Mowry, and Padmanabhan Pillai. Meld: A Declarative Approach to Programming Ensembles. In *Proceedings of the IEEE/RSJ 2007 International Conference on Intelligent Robots and Systems (IROS)*, October 2007.

34. Amit Manjhi, Phillip B. Gibbons, Anastassia Ailamaki, Charles Garrod, Bruce M. Maggs, Todd C. Mowry, Christopher Olston, Anthony Tomasic, and Haifeng Yu. Invalidation Clues for Database Scalability Services. In *Proceedings of the 2007 IEEE 23rd International Conference on Data Engineering (ICDE)*, pages 316-325, April 2007.

35. Michael De Rosa, Peter Lee, Seth Goldstein, Jason Campbell, Padmanabhan Pillai, and Todd C. Mowry. Distributed Watchpoints: Debugging Large Multi-Robot Systems. In *Proceedings of the 2007 IEEE International Conference on Robotics and Automation (ICRA)*, pages 3723-3729, April 2007.

36. Benjamin D. Rister, Jason Campbell, Padmanabhan Pillai, and Todd C. Mowry. Integrated Debugging of Large Modular Robot Ensembles. In *Proceedings of the 2007 IEEE International Conference on Robotics and Automation (ICRA)*, pages 2227-2234, April 2007.

37. Shimin Chen, Phillip Gibbons, Michael Kozuch, Vasileios Liaskovitis, Anastassia Ailamaki, Guy Blelloch, Babak Falsafi, Limor Fix, Nikos Hardavellas, Todd C. Mowry and Chris Wilkerson. Scheduling Threads for Constructive Cache Sharing on CMPs. In *Proceedings of 18th ACM Symposium on Parallelism in Algorithms and Architectures (SPAA)*, pages 105-115, June 2007.

38. Shimin Chen, Babak Falsafi, Phillip B. Gibbons, Michael Kozuch, Todd C. Mowry, Radu Teodorescu, Anastassia Ailamaki, Limor Fix, Gregory R. Ganger, Bin Lin, and Steven W. Schlosser. Log-Based Architectures for General-Purpose Monitoring of Deployed Code. In *Proceedings of the Workshop on Architectural and System Support for Improving Software Dependability (ASID), held with ASPLOS XII*, October 2006.

39. Christopher B. Colohan, Anastassia Ailamaki, J. Gregory Steffan, and Todd C. Mowry. Tolerating Dependences Between Large Speculative Threads Via Sub-Threads. In *Proceedings of the 33rd Annual International Symposium on Computer Architecture (ISCA)*, pages 216-226, June 2006.

40. Amit Manjhi, Anastassia Ailamaki, Bruce M. Maggs, Todd C. Mowry, Christopher Olston, and Anthony Tomasic. Simultaneous Scalability and Security for Data-Intensive Web Applications. In *Proceedings of the 2006 ACM SIGMOD International Conference on Management of Data*, pages 241-252, June 2006.

41. Vasileios Liaskovitis, Shimin Chen, Phillip B. Gibbons, Anastassia Ailamaki, Guy E. Blelloch, Babak Falsafi, Limor Fix, Nikos Hardavellas, Michael Kozuch, Todd C. Mowry, and Chris Wilkerson. Parallel Depth First vs. Work Stealing Schedulers on CMP Architectures. In *Proceedings of 18th ACM Symposium on Parallelism in Algorithms and Architectures (SPAA)*, August 2006.

42. Brian Kirby, Jason Campbell, Burak Aksak, Padmanabhan Pillai, James F. Hoburg, Todd C. Mowry, and Seth Copen Goldstein. Catoms: Moving Robots without Moving Parts. In *Proceedings of the Twentieth National Conference qon Artificial Intelligence (AAAI)*, pages 1730-1731, July 2005.

43. Christopher Olston, Amit Manjhi, Charles Garrod, Anastassia Ailamaki, Bruce M. Maggs, and Todd C. Mowry. A Scalability Service for Dynamic Web Applications. In *Proceedings of the Second Biennial Conference on Innovative Data Systems Research (CIDR)*, pages 56-69, January 2005.

44. Christopher B. Colohan, Anastassia Ailamaki, J. Gregory Steffan, and Todd C. Mowry. Optimistic Intra-Transaction Parallelism on Chip Multiprocessors. In *Proceedings of the 31st International Conference on Very Large Data Bases (VLDB)*, pages 73-84, September 2005.

45. Shimin Chen, Anastassia Ailamaki, Phillip B. Gibbons, and Todd C. Mowry. Inspector Joins. In *Proceedings of the 31st International Conference on Very Large Data Bases (VLDB)*, pages 817-828, September 2005.

46. Antonia Zhai, Christopher B. Colohan, J. Gregory Steffan, and Todd C. Mowry. Compiler Optimization of Memory-Resident Value Communication Between Speculative Threads. In *Proceedings of the 2007 International Symposium on Code Generation and Optimization (CGO)*, pages 39-52, March 2004.

47. Shimin Chen, Anastassia Ailamaki, Phillip B. Gibbons and Todd C. Mowry. Improving Hash Join Performance through Prefetching. In *Proceedings of the 20th International Conference on Data Engineering (ICDE)*, pages 116-127, April 2004. *(This paper received the Best Paper Award.)*

48. Antonia Zhai, Christopher B. Colohan, J. Gregory Steffan and Todd C. Mowry. Compiler Optimization of Scalar Value Communication Between Speculative Threads. In *Proceedings of the Tenth International Conference on Architectural Support for Programming Languages and Operating Systems (ASPLOS-X)*, October 2002.

49. Shimin Chen, Philip B. Gibbons, Todd C. Mowry and Gary Valentin. Fractal Prefetching B+-Trees: Optimizing Both Cache and Disk Performance. In *Proceedings of the 2002 ACM SIGMOD International Conference on Management of Data*, June 2002.

50. J. Gregory Steffan, Christopher B. Colohan, Antonia Zhai and Todd C. Mowry. Improving Value Communication for Thread-Level Speculation. In *Proceedings of the Eighth International Symposium on High-Performance Computer Architecture (HPCA)*, February 2002.

51. Shimin Chen, Philip B. Gibbons and Todd C. Mowry. Improving Index Performance through Prefetching. In *Proceedings of the 2001 ACM SIGMOD International Conference on Management of Data*, pages 235-246, May 2001. *(This paper received the Runner-Up for Best Paper Award.)*

52. Angela Demke Brown and Todd C. Mowry. Taming the Memory Hogs: Using Compiler-Inserted Releases to Manage Physical Memory Intelligently. In *Proceedings of the Fourth Symposium on Operating Systems Design and Implementation (OSDI 2000)*, pages 31-44, October 2000.

53. J. Gregory Steffan, Christopher B. Colohan, Antonia Zhai and Todd C. Mowry. A Scalable Approach to Thread- Level Speculation. In *Proceedings of the 27th Annual International Symposium on Computer Architecture*, pages 1-12, June 2000.

54. Todd C. Mowry and Sherwyn R. Ramkissoon. Software-Controlled Multithreading Using Informing Memory Operations. In *Proceedings of the Sixth International Symposium on High-Performance Computer Architecture*, January, 2000.

55. Chi-Keung Luk and Todd C. Mowry. Memory Forwarding: Enabling Aggressive Layout Optimizations by Guaranteeing the Safety of Data Relocation. In *Proceedings of the 26th Annual International Symposium on Computer Architecture (ISCA)*, pages 88-99, May 1999.

56. Chi-Keung Luk and Todd C. Mowry. Cooperative Prefetching: Compiler and Hardware Support for Effective Instruction Prefetching in Modern Microprocessors. In *Proceedings of the 31st Annual International Symposium on Microarchitecture*, pages 182-193, December 1998.

57. J. Gregory Steffan and Todd C. Mowry. The Potential for Using Thread-Level Data Speculation to Facilitate Automatic Parallelization. In *Proceedings of the Fourth International Symposium on High-Performance Computer Architecture*, pages 2-13, February, 1998.

58. Charles Chan, Adley Lo, and Todd C. Mowry. Comparative Evaluation of Latency Tolerance Techniques for Software Distributed Shared Memory. In *Proceedings of the Fourth International Symposium on High-Performance Computer Architecture*, pages 300-311, February, 1998.

59. Todd C. Mowry and Chi-Keung Luk. Predicting Data Cache Misses in Non-Numeric Applications Through Correlation Profiling. In *Proceedings of the 30th Annual International Symposium on Microarchitecture*, pages 314-320, December 1997.

60. Todd C. Mowry, Angela K. Demke and Orran Krieger. Automatic Compiler-Inserted I/O Prefetching for Out-of-Core Applications. In *Proceedings of the Second Symposium on Operating Systems Design and Implementation (OSDI '96)*, pages 3-17, October 1996. *(This paper received the Best Paper Award.)*

61. Chi-Keung Luk and Todd C. Mowry. Compiler-Based Prefetching for Recursive Data Structures. In *Proceedings of the Seventh International Conference on Architectural Support for Programming Languages and Operating Systems*, pages 222-233, October 1996.

62. Edouard Bugnion, Jennifer M. Anderson, Todd C. Mowry, Mendel Rosenblum and Monica S. Lam. Compiler- Directed Page Coloring for Multiprocessors. In *Proceedings of the Seventh International Conference on Architectural Support for Programming Languages and Operating Systems*, pages 244-255, October 1996.

63. Mark Horowitz, Margaret Martonosi, Todd C. Mowry, and Michael D. Smith. Informing Memory Operations: Providing Memory Performance Feedback in Modern Processors. In *Proceedings of the 23rd Annual International Symposium on Computer Architecture*, pages 260-270, May 1996.

64. Todd C. Mowry, Monica S. Lam and Anoop Gupta. Design and Evaluation of a Compiler Algorithm for Prefetching. In *Proceedings of the Fifth International Conference on Architectural Support for Programming Languages and Operating Systems*, pages 62-73, October 1992.

65. Anoop Gupta, John Hennessy, Kourosh Gharachorloo, Todd Mowry, and Wolf-Dietrich Weber. Comparative Evaluation of Latency Reducing and Tolerating Techniques. In *Proceedings of the 18th Annual International Symposium on Computer Architecture*, pages 254-263, May 1991.

66. Anoop Gupta, Wolf-Dietrich Weber, and Todd Mowry. Reducing Memory and Traffic Requirements for Scalable Directory-Based Cache Coherence Schemes. In *Proceedings of International Conference on Parallel Processing*, pages 312-321, August 1990.

Technical Reports

- Evangelos Vlachos, Michelle Goodstein, Michael Kozuch, Shimin Chen, Babak Falsafi, Phillip B. Gibbons, Todd C. Mowry, and Olatunji Ruwase. Parallel LBA: Conherence-based Parallel Monitoring of Multithreaded Applications. Carnegie Mellon University Technical Report CMU-CS-09-108, March 2009.
- Michelle Goodstein, Evangelos Vlachos, Shimin Chen, Phillip Gibbons, Michael Kozuch, and Todd C. Mowry. The Butterfly Model: Theoretical Foundations. Carnegie Mellon University Technical Report CMU-CS-08-170, February 2009.
- Amit Manjhi, Anastassia Ailamaki, Bruce M. Maggs, Todd C. Mowry, Christopher Olston, and Anthony Tomasic. Simultaneous Scalability and Security for Data-Intensive Web Applications. Carnegie Mellon University Technical Report CMU-CS-06-116, March 2006.
- Charles Garrod, Amit Manjhi, Anastassia Ailamaki, Phillip Gibbons, Bruce Maggs, Todd Mowry, Christopher Olston, and Anthony Tomasic. Scalable Consistency Management for Web Database Caches. Carnegie Mellon University Technical Report CMU-CS-06-128, July 2006.
- Amit Manjhi, Phillip B. Gibbons, Anastassia Ailamaki, Charles Garrod, Bruce M. Maggs, Todd C. Mowry, Christopher Olston, Anthony Tomasic, and Haifeng Yu. Invalidation Clues

for Database Scalability Services. Carnegie Mellon University Technical Report CMU-CS-06-139, July 2006.

- Christopher B. Colohan, Anastassia Ailamaki, J. Gregory Steffan, and Todd C. Mowry. Supporting Large Speculative Threads for Databases and Beyond. Carnegie Mellon University Technical Report CMU-CS-05-109, July 2005.
- Christopher B. Colohan, Anastassia Ailamaki, J. Gregory Steffan, and Todd C. Mowry. Optimistic Intra-Transaction Parallelism on Chip Multiprocessors. Carnegie Mellon University Technical Report CMU-CS-05-118, March 2005.
- Shimin Chen, Anastassia Ailamaki, Phillip B. Gibbons, and Todd C. Mowry. Improving Hash Join Performance through Prefetching. Carnegie Mellon University Technical Report CMU-CS-03-157, October 2003.
- Shimin Chen, Phillip B. Gibbons, Todd C. Mowry, and Gary Valentin. Fractal Prefetching B+-Trees: Optimizing Both Cache and Disk Performance. Carnegie Mellon University Technical Report CMU-CS-02-115, March 2002.
- Spiros Papadimitrious and Todd C. Mowry. Exploring Thread-Level Speculation in Software: The Effects of Memory Access Tracking Granularity. Carnegie Mellon University Technical Report CMU-CS-01-145, July 2001.
- Shimin Chen, Phillip B. Gibbons and Todd C. Mowry. Improving Index Performance through Prefetching. Carnegie Mellon University Technical Report CMU-CS-00-177, December 2000.
- J. Gregory Steffan, Christopher B. Colohan and Todd C. Mowry. Extending Cache Coherence to Support Thread-Level Data Speculation on a Single Chip and Beyond. Carnegie Mellon University Technical Report CMU-CS-98-171, December 1998.
- Todd C. Mowry and Sherwyn R. Ramkissoon. Software-Controlled Multithreading Using Informing Memory Operations. Carnegie Mellon University Technical Report CMU-CS-98-169, October 1998.
- Chi-Keung Luk and Todd C. Mowry. Compiler and Hardware Support for Automatic Instruction Prefetching: A Cooperative Approach. Carnegie Mellon University Technical Report CMU-CS-98-140, June 1998.
- J. Gregory Steffan, Christopher B. Colohan and Todd C. Mowry. Architectural Support for Thread-Level Data Speculation. Carnegie Mellon University Technical Report CMU-CS-97-188, November 1997.
- Todd C. Mowry and Chi-Keung Luk. Predicting Data Cache Misses in Non-Numeric Applications Through Correlation Profiling. Carnegie Mellon University Technical Report CMU-CS-97-175, September 1997.
- J. Gregory Steffan and Todd C. Mowry. The Potential for Thread-Level Data Speculation in Tightly-Coupled Multiprocessors. University of Toronto Technical Report CSRI-TR-350, February 1997.
- Chi-Keung Luk and Todd C. Mowry. Predicting Data Cache Misses in Non-Numeric Applications Through Correlation Profiling. University of Toronto Technical Report CSRI-TR-359, February, 1997.
- Mark Horowitz, Margaret Martonosi, Todd C. Mowry, and Michael D. Smith. Informing Loads: Enabling Software to Observe and React to Memory Behavior. Stanford University Technical Report CSL-TR-95-673, July 1995.
- Todd C. Mowry. Tolerating Latency Through Software-Controlled Data Prefetching. Technical Report CSL-TR-94-628, Stanford University, June 1994.

Patents Held

- Todd C. Mowry. U.S. Patent 7,127,586: Prefetching hints. Issued October, 2006.

- Shimin Chen, Phillip B. Gibbons, and Todd C. Mowry. U.S. Patent 6,772,179: System and method for improving index performance through prefetching. Issued August, 2004.
- Todd C. Mowry. U.S. Patent 6,240,488: Prefetching hints. Issued May, 2001.
- Todd C. Mowry. U.S. Patent 5,732,242: Consistently specifying way destinations through prefetching hints. Issued March, 1998.
- Todd C. Mowry and Earl A. Killian. U.S. Patent 5,696,958: Method and apparatus for reducing delays following the execution of a branch instruction in an instruction pipeline. Issued December, 1997.

Distinguished Lectures

9/25/13: "The Log-Based Architectures Project: Exploiting Multicore to Help Parallel Software Behave Correctly", University of Wisconsin-Madison, Department of Computer Science, Distinguished Lecture Series.

2/7/08: "Pario: the Next Step Beyond Audio and Video", University of Toronto, Department of Electrical and Computer Engineering Distinguished Lecture Series.

Professional Activities

Professional Societies

- Member of the Institute of Electrical and Electronics Engineers (IEEE)
- Member of the Association of Computing Machinery (ACM)

Industrial Employment

5/04–6/07: Intel Corporation, Pittsburgh, Pennsylvania: Director of the Intel Research Pittsburgh Lab.

6/89–11/93: Silicon Graphics, Inc., Mountain View, California (formerly MIPS Computer Systems, Sunnyvale California): Computer Architect (part-time).

Consulting

7/07–2/11: Intel Corporation, Pittsburgh, Pennsylvania: Research Advisor.

8/96–4/04: SandCraft, Inc., Santa Clara, California: Member of the Technical Advisory Board.

1/96–4/04: IBM, Toronto: Visiting Scientist.

12/93–12/96: Silicon Graphics, Inc., Mountain View, California: Computer Architecture Consultant.

University Committee Work and Other Service Activities

Committee Work

- Faculty Hiring Committees:
  - Computer Science Department: 2008, 2010.
  - Computer Systems Area: 2000, 2001, 2002, 2003, 2004.
- Chair of the Computer Science Department Head Selection Committee, 2007.
- CSD Ph.D. Admissions Committee, 2001, 2002, 2003.
- School of Computer Science Council, 2001-2004.
- Carnegie Mellon University Faculty Senate (Presidential Appointeee), 2000-2001.
- Computer Systems Area Advocate, 2000-2004.
- Doctoral Review Committee, 1998-2004.
- ACM Thesis Award Nomination Committee, 1999.

Journal, Conference and Workshop Organization

- Editor-in-Chief, ACM Transactions on Computer Systems, 2013-2018.
- Associate Editor, ACM Transactions on Computer Systems, 2001-2013.
- Program Chair of the International Conference on Architectural Support for Programming Languages and Operating Systems (ASPLOS), 2011.
- Sponsorship Chair, International Conference on Architectural Support for Programming Languages and Operating Systems (ASPLOS), 2010.
- Member of the ISCA program committee, 1998, 2000, and 2011.
- Member of the ASPLOS program committee, 2000, 2004.
- Member of the ASPLOS external review committee, 2012.
- Co-Program Chair of the International Conference on Parallel Architectures and Compilation Techniques (PACT), 2001.
- Member of the International Symposuim on Microarchitecture program committee, 1998.
- Member of the Workshop on Architectural and System Support for Improving Software Dependability (ASID) program committee, 2005.
- Member of the Workshop on Memory System Performance (MSP) program committee (in conjunction with PLDI), 2002 and 2004.
- Member of the IBM CASCON program committee, 1995, 1996, 1997.
- Member of the First SUIF Workshop Program Committee, 1996.
- Organized the CASCON 97 workshop on Compiler Optimization, 1997.
- Initiated and organized the 1st Toronto / Rochester Workshop on Shared-Memory Multiprocessors, 1994.