(12) INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(19) World Intellectual Property Organization
International Bureau

(43) International Publication Date
20 August 2009 (20.08.2009)

PCT

(10) International Publication Number
WO 2009/100542 A1

(51) International Patent Classification:
G06F 17/50 (2006.01)

(21) International Application Number:
PCT/CA2009/000190

(22) International Filing Date:
13 February 2009 (13.02.2009)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:
61/028,409 13 February 2008 (13.02.2008) US

(71) Applicant *(for all designated States except US)*: **DIRTT ENVIRONMENTAL SOLUTIONS, LTD.** [CA/CA]; 7303-30th Street S.E., Calgary, Alberta T2C 1N6 (CA).

(72) Inventors; and
(75) Inventors/Applicants *(for US only)*: **LOBERG, Barrie A.** [CA/CA]; Site 15, Box 10, RR1, Millarville, Alberta T0L 1K0 (CA). **BODEN, Dale** [CA/CA]; 231 Tuscany Dr. NW, Calgary, Alberta T3L 2H9 (CA).

(74) Agents: **BENNETT JONES LLP** et al.; 3400 - One First Canadian Place, Box 130, Toronto, Ontario M5X 1A4 (CA).

(81) Designated States *(unless otherwise indicated, for every kind of national protection available)*: AE, AG, AL, AM, AO, AT, AU, AZ, BA, BB, BG, BH, BR, BW, BY, BZ, CA, CH, CN, CO, CR, CU, CZ, DE, DK, DM, DO, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, GT, HN, HR, HU, ID, IL, IN, IS, JP, KE, KG, KM, KN, KP, KR, KZ, LA, LC, LK, LR, LS, LT, LU, LY, MA, MD, ME, MG, MK, MN, MW, MX, MY, MZ, NA, NG, NI, NO, NZ, OM, PG, PH, PL, PT, RO, RS, RU, SC, SD, SE, SG, SK, SL, SM, ST, SV, SY, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, ZA, ZM, ZW.

(84) Designated States *(unless otherwise indicated, for every kind of regional protection available)*: ARIPO (BW, GH, GM, KE, LS, MW, MZ, NA, SD, SL, SZ, TZ, UG, ZM, ZW), Eurasian (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), European (AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HR, HU, IE, IS, IT, LT, LU, LV, MC, MK, MT, NL, NO, PL, PT, RO, SE, SI, SK, TR), OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, ML, MR, NE, SN, TD, TG).

Published:

— *with international search report (Art. 21(3))*

— *before the expiration of the time limit for amending the claims and to be republished in the event of receipt of amendments (Rule 48.2(h))*

(54) **Title**: CREATION AND MODIFICATION OF VALID FUNCTIONAL DESIGN LAYOUTS

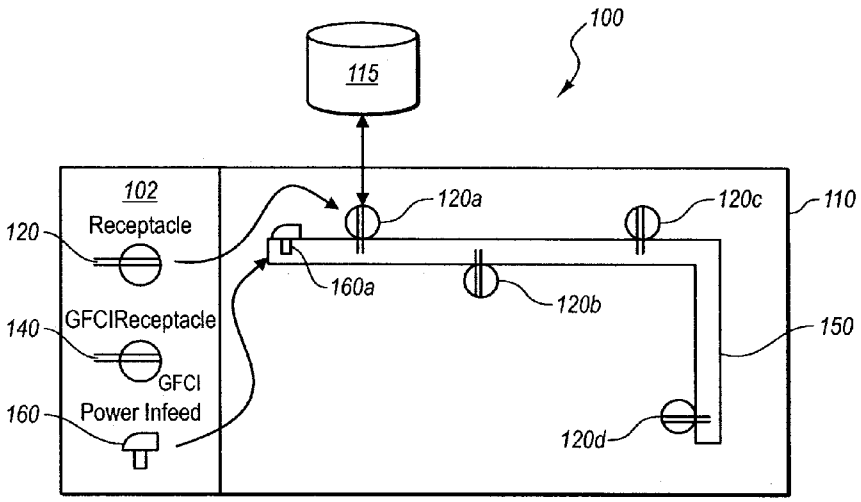


FIG. 1A

(57) **Abstract**: A software application can capture product parameters and attributes in order to allow a non-expert user to create a valid functional system layout in a design space. To enable accurate layouts, the software application can automatically apply all applicable components, connectors, rules, attributes, codes, and behaviors to each component/design element in the design space. The intelligent, object-oriented software can apply rules and behaviors automatically, not only providing an accurate, valid layout, but also optimizing the layout. The intelligent, object-oriented software can recognize required changes effected by a user-driven change, and automatically resolve the layout design.

WO 2009/100542 A1

CREATION AND MODIFICATION OF VALID FUNCTIONAL DESIGN LAYOUTS

CROSS-REFERENCE TO RELATED APPLICATIONS

[0001] The present invention claims the benefit of priority to U.S. Provisional Patent Application No. 61/028,409, filed on February 13, 2008, entitled "Automated Layout of Electrical Components and Wiring Intent in Design Software." The present invention is also a continuation-in-part of U.S. Patent Application No. 11/204,421, filed on August 16, 2005, entitled "Capturing a User's Intent in Design Software", which claims the benefit of priority to U.S. Patent Provisional Application 60/602,233, filed on August 17, 2004, now abandoned, entitled "Method and Apparatus for the Selection, Organization and Configuration of Products through Object Oriented Design Intent." The entire contents of each of the aforementioned patent applications and patent are incorporated herein by reference.

BACKGROUND OF THE INVENTION

1. Technical Field

[0002] This invention relates generally to systems, methods, and computer program products for modeling and design. More specifically, the present invention relates to software used to design space layouts and corresponding components.

2. Background and Relevant Art

[0003] As computerized systems have increased in popularity, so has the range of applications that incorporate computational technology. For example, architects and interior designers (or "designers") use a wide range of design software for designing the aesthetic as well as functional aspects of a given residential or commercial space. In some cases, the designer might use some software programs that are better suited for exterior design, and then use other software programs that are better suited for interior design.

[0004] For example, a designer might implement one software program to design an overall look of a building, and then use the software to design or position each of the structural components of the building, such as weight-bearing walls, trusses in a roof, and so on. The designer might then use another software program, whether separately, or as an add-on to the first software program, to design the functional components of a building, such as the electrical systems, plumbing systems, and heating, ventilating, and air conditioning (HVAC) systems. Finally, the designer might then use still a further software program to design the interior features of the building, such as functional walls, position of furniture, lighting fixtures, and so forth.

[0005] When designing the exterior and/or interior of a given residential or commercial space, the designer may need to take care that each of the elements in the design are structurally and functionally sound when built. This is because typical design software can allow spaces to be fairly configurable to suit the user's desires without specific regard in many cases to whether the design will actually

function or comply with applicable codes and regulations when built. For example, one typical software design program (e.g., conventional computer-aided design or CAD programs) might allow a designer to layout an electrical circuit that is ill-suited for the number or type of electrical components and loads that the designer has included in the circuit. If the circuit were actually constructed as laid out by the designer, the circuit may be hot and be prone to overload, may be a fire hazard, or may not satisfy the requirements of applicable building codes. In a situation such as this, however, the builder might indicate to the designer that the layout is physically impossible or impractical, and ask for a redesign. This, of course, can lead to any number of inefficiencies.

[0006] For example, conventional design software is often difficult to use, and heavily dependent on the skill of a user, such that conventional software for laying out functional systems can tend to be accessible primarily or uniquely to "product experts." In general, a product expert is one who understands the product components, rules, and behaviors in relation to other components (functional and non-functional) in the layout software. Nevertheless, even product experts can err when attempting to remember all aspects of any particular component, and can fail to remember or identify a component's relationships with respect to other parts of the layout.

[0007] One of the problems with many design software programs that can lead to the design of physically or functionally impractical structures, is that many such design problems require some drawing of a space in flat, two-dimensional space. For example, an electrical wiring layout is typically designed in a plan view

that emphasizes primarily only length and width from above. With views such as these, the designer will either need to independently visualize the three-dimensional spacing, or will need to perform a separate three-dimensional rendering of the design, if the software allows for it.

[0008] In addition, neither the three-dimensional rendering nor the two-dimensional drawing views are designed to accommodate necessary modifications to the objects or walls, based on real-world materials, or other important constraints. For example, a designer might use a first two-dimensional software package to place several electrical components in a work space. After which the designer may place furniture and work spaces within the same work space using another two-dimensional software package or an add-on to the first. The designer may then use another program to produce a three-dimensional rendering of the workspace. Upon reviewing the three-dimensional rendering of the workspace, the designer may find that some of the electrical components need to be repositioned relative to the furniture and work spaces or vise versa. Conventional design software typically requires changes to the electrical layout design or changes in furniture selection and placement to be made manually, which often results in a complete rework to ensure the end-user captures all of the needed changes.

[0009] For example, changing existing layouts can involve not only the manual placement of components, but also various considerations related to the changing of the walls, furniture, or other design components. Furthermore, the repositioning or replacement of functional components can require changes to the circuits, jumpers, breakers, pumps, vents, and machines supporting the components.

In addition, there are several mistakes that can be made with regard to estimating installation times, as well as time and cost for manual placement, rework, and so forth.

[0010] One way that conventional software lends itself to these difficulties is that there it often has inexact or error-prone software relationships between textual versions and graphical oriented versions of components. For example, conventional layout software applications will sometimes, upon completion of a design, utilize separate databases for the graphic components and textual components used to order the product. The linking between these different databases, however, introduces another area for potential error, and requires additional time for auditing the graphical results against the textual results.

[0011] Once a design has been finalized by a designer, the designer will need to generate one or more parts lists that reflect the various dimensions and parts placed in any of the design views. An estimator (or the like) may then use the parts list for any number of cost estimate or ordering ends. Unfortunately, there is generally not a convenient way for an accurate parts list to be generated automatically from one or more design views. For example, even though a designer might use a conventional design software program to design one or more views of a space, the designer might need to independently deduce a parts list based on each of the different views. In some cases, the designer might hire another person to identify each part needed.

[0012] Accordingly, an advantage in the art can be realized with systems, methods, and computer program products that provide a designer with the ability to

design spaces in a highly configurable and user-friendly manner. In particular, an advantage can be realized with systems that are configured to help ensure functional system layouts (electrical, plumbing, HVAC, etc.) are physically and functionally possible in at least one respect.

BRIEF SUMMARY OF THE INVENTION

[0013] Implementations of the present invention overcome one or more problems in the art with systems, methods, and computer program products configured to capture product information, such as component attributes and parameters, to allow a non-expert user to design a valid functional system layout within a design space. In particular, implementations of the present invention automatically associate product parameters and attributes to both functional and non-functional design components, including all applicable connectors, rules, features, characteristics, and behaviors. Based upon the components and their associated attributes and parameters, implementations of the present invention can automatically apply rules and behaviors to help create a functional system layout in a design space that is physically and functionally valid. Furthermore, implementations of the present invention can optimize the parameters of the functional system layout.

[0014] For example, in accordance with an implementation of the present invention a method of automatically configuring user input within a computerized environment to represent a design or layout of components within a design space can involve identifying one or more source components in a design space. The method

can further involve receiving user input regarding one or more functional components that are configured to receive flow from the one or more source components. Also, the method can involve automatically creating a valid functional system layout within the design space by optimizing the parameters of functional system lines connecting the one or more source components and the one or more functional components.

[0015] Additionally, a method in accordance with an implementation of the present invention of automatically configuring user input in a computerized environment to represent a design or layout of components within a design space can involve receiving user input regarding the placement of one or more functional components within a design space. The method can also involve receiving user input regarding the placement of one or more source components within the design space. The method can then involve populating functional system lines with optimized characteristics. The functional system lines can connect the one or more source components and the one or more functional components. The method can further involve receiving additional user input regarding the placement, relocation, or change of one or more of a non-functional design component, a functional component, or a source component within the design space. Additionally, the method can resolve the functional system lines in view of the additional user input.

[0016] Additional features and advantages of exemplary implementations of the invention will be set forth in the description which follows, and in part will be obvious from the description, or may be learned by the practice of such exemplary implementations. The features and advantages of such implementations may be

realized and obtained by means of the instruments and combinations particularly pointed out in the appended claims. These and other features will become more fully apparent from the following description and appended claims, or may be learned by the practice of such exemplary implementations as set forth hereinafter.

BRIEF DESCRIPTION OF THE DRAWINGS

[0017] In order to describe the manner in which the above-recited and other advantages and features of the invention can be obtained, a more particular description of the invention briefly described above will be rendered by reference to specific embodiments thereof which are illustrated in the appended drawings. One will appreciate that elements of similar structures or functions are generally represented by like reference numerals for illustrative purposes throughout the Figures. Understanding that these drawings depict only typical embodiments of the invention and are not therefore to be considered to be limiting of its scope, the invention will be described and explained with additional specificity and detail through the use of the accompanying drawings in which:

[0018] Figure 1A illustrates a conceptual diagram of a user interface and one or more components of a design software program when a user enters input into a design space in accordance with an implementation of the present invention;

[0019] Figure 1B illustrates a conceptual diagram of the user interface of Figure 1A in which the design software program automatically populates optimal functional system lines and flow intent lines between functional components in the design space in accordance with an implementation of the present invention;

[0020] Figure 1C illustrates a conceptual diagram of the user interface of Figure 1B when a user enters additional input into the design space in accordance with an implementation of the present invention;

[0021] Figure 1D illustrates a conceptual diagram of the user interface of Figure 1C in which the design software program automatically resolves the functional system lines in the design space in view of the additional user input in accordance with an implementation of the present invention;

[0022] Figure 1E illustrates a conceptual diagram of the user interface of Figure 1D when a user manually repositions one or more flow intent lines in accordance with an implementation of the present invention;

[0023] Figure 2A illustrates a conceptual diagram of a user interface and one or more functional components of a design software program when a user enters input into a design space in accordance with an implementation of the present invention;

[0024] Figure 2B illustrates a conceptual diagram of the user interface of Figure 2A in which the design software program automatically resolves any invalid functional components in accordance with an implementation of the present invention;

[0025] Figure 2C illustrates a conceptual diagram of the user interface of Figure 2A in which the design software program warns against placement of an invalid functional component in the design space in accordance with an implementation of the present invention;

[0026] Figure 3A illustrates a conceptual diagram of a user interface and one or more functional components of a design software program when a user enters input into a design space in accordance with an implementation of the present invention;

[0027] Figure 3B illustrates a conceptual diagram of the user interface of Figure 3A when a user requests the design software program to place functional components within the design space in accordance with an implementations of the present invention;

[0028] Figure 3C illustrates a conceptual diagram of the user interface of Figure 3B in which the design software program automatically and optimally places functional components and functional system lines in the design space in accordance with an implementation of the present invention;

[0029] Figure 4 illustrates a flowchart of a series of acts in a method in accordance with an implementation of the present invention of designing a valid functional system layout within the design space; and

[0030] Figure 5 illustrates a flowchart of a series of acts in a method in accordance with another implementation of the present invention of designing a valid functional system layout within the design space.

DETAILED DESCRIPTION OF THE PREFERRED EMBODIMENTS

[0031] The present invention extends to systems, methods, and computer program products configured to capture product information, such as component attributes and parameters, to allow a non-expert user to design a valid functional system layout within a design space. In particular, implementations of the present invention automatically associate product parameters and attributes to both functional and non-functional design components, including all applicable connectors, rules, features, characteristics, and behaviors. Based upon the components and their associated attributes and parameters, implementations of the present invention can automatically apply rules and behaviors to help create a functional system layout in a design space that is physically and functionally valid. Furthermore, implementations of the present invention can optimize the parameters of the functional system layout.

[0032] For example, based on the placement of source components (power sources, gas and water valves, etc.) and functional components (outlets, vents, sinks, showers, etc.) implementations of the present invention can optimize the routing of functional lines and conduits therebetween. Implementations of the present invention can also optimize the routing of functional lines and conduits by reducing the length of needed functional lines and conduits, reducing the space requirements needed for the functional system, reducing the number of joints and connectors, and adjusting the functional layout to correspond to product attributes, such as stock lengths etc.

[0033] One will thus appreciate that because the systems, methods, and computer program products of the present invention include design components (function and non-functional) that include associated product attributes and parameters, a non-expert can easily and efficiently design a valid functional design layout. Furthermore, implementations of the present invention can apply rules and behaviors automatically, not only providing an accurate, valid layout, but also allowing change at any time without rework. Implementations of the present invention can also recognize required changes effected by user driven changes, and automatically resolve the functional layout design, including required component substitutions and dimensional changes, to help maintain validity.

[0034] Furthermore, implementations of the present invention can represent the design space in two-dimensional graphics, three-dimensional graphics, and textual information. Thus, a user can easily and quickly toggle between different views or representations to gain a complete understanding of the design space and design components therein. Each of these separate views can be representations of the same dataset. Thus, one will appreciate that implementations of the present invention can eliminate the need to audit between information sources. Furthermore, because each representation can be based on the same dataset, all three representations can be resolved and modified instantly and simultaneously.

[0035] As mentioned above, implementations of the present invention can allow a non-expert user to design a valid functional system layout within a design space. As used herein, the term "functional system" refers to one or more of an electrical system, plumbing system, HVAC system, or the like. The Figures and

corresponding text included hereafter illustrate examples of creating valid electrical system layouts for ease of description. One will appreciate, however, that the systems, methods, and computer product programs can create valid layouts of other functional systems, such as for example, plumbing, and HVAC systems.

[0036] As used herein, the term "source component" refers to a source or "inlet" of a functional system within a design space (e.g., workspace, room, floor, or building). For example, a source component in an electrical system can include an electrical power infeed, main power line, etc. Similarly, a source component in a plumbing system can include a main water-line, water valve, etc. A source component in a HVAC system can include a main gas-valve, a furnace, or an air conditioning unit, etc. One will appreciate that a source component can vary depending upon the design space being used. For example in an electrical system, a source component for a building can include the main power line entering the building, while the source component for a workspace can include a power infeed.

[0037] Additionally, as used herein, the term "functional component" refers broadly to any component of a functional system. In some instances, the term "functional component" refers to an "outlet" of a functional system. For example, in an electrical system a functional component can comprise an electrical receptacle, a junction box, or an electrical fixture, etc. A functional component in a plumbing system can include a sink, a drinking fountain, or a toilet, etc. Along similar lines, a functional component in a HVAC system can include a ventilation register, etc.

[0038] Furthermore, as used herein the term "functional system line" refers to the connectors and conduits needed to interconnect the functional components and

source components of a functional system. Thus, in an electrical system, the functional system lines can include, for example, electrical wiring, jumpers, junction boxes, circuit breakers, etc. In a plumbing system, the functional system lines can include, for example, piping, venting, pumps, etc. Functional system lines in a HVAC system can include ducts, vents, intakes, etc.

[0039] In at least one implementation, the system, methods, and computer products of the present invention allow a user to select functional and source components from a product menu, which the user can then apply to design components within a design space, such as for example, a wall, modular furniture, a panel, or similar components. One will appreciate that a user can also select the design components from a product menu, or the user can import them into the software system from another software program. For example, Figure 1A illustrates an exemplary user interface 100 for a design software program in accordance with an implementation of the present invention.

[0040] As shown in Figure 1A, the user interface 100 can provide a user with a selection portion 102 with a list of images or icons, such as a receptacle icon 120, a ground fault circuit interrupter (GFCI) receptacle icon 140, a power infeed icon 160, and the like. The icons can represent items that the user can place in a design space 110 portion of the user interface 100. In at least one implementation, the icon is a symbol representing the item (e.g., a recognized functional component symbol). In alternative implementations, the icon can be an image associated with the item (e.g., image of receptacle 120). Thus, the icons 120, 140, 160 etc., provide

the user with some initial information about the components that the user can add to the design space.

[0041] Of course, the options provided to the user are not limited to the icons shown, necessarily. For example, the design software can provide the user with other options (not shown) as part of the design program for modifying the type of component 120, 140, 160 etc. In particular, the user may make choices to change the color, material, size, texture, thickness, and so forth of the component.

[0042] In any event, Figure 1A shows that the user selects the receptacle icon 120, and draws a first receptacle 120a in the design space 110, as well as second, third, and fourth receptacles 120(b-d). The user can position the functional components (e.g., 120(a-d)) about or in relation to structural design components. For example, as shown in Figure 1A, the user can position the receptacles 120(a-d) about or in relation to a wall component 150. The user can create the structural design components by selecting them from a toolbar (or, "selection portion") 102, similar to the functional and source components 120, 140, 160. Alternatively, the user can import the structural design components into the design space from a separate software program, such as for example, a CAD program. According to some implementations of the present invention, the structural design components can be resolvable objects, such as those described in commonly-assigned U.S. Patent Application No. 11/204,420, filed August 16, 2005, now U.S. Patent No. 7,277,830, entitled "Capturing a User's Design Intent with Resolvable Objects," the entire content of which is incorporated by reference herein.

[0043] In either case, as the user enters the functional components (e.g., 120(a-d)) into the design space 110 about or in relation to the structural design components (e.g., 150), the software system can receive the input and identify the components. In particular, the software system can identify each component's location, type, size, relationships to other components, etc. Upon identification of the components, the software system can create an intelligent, automatically resolvable object for each component that includes one or more attributes and product parameters. For example, Figure 1A illustrates an object database 115 comprising objects for each design element (e.g., 120a, 120b, 120c, 160a) in design space 110.

[0044] The software system can create and compile such objects in virtually any object-oriented programming language, including, for example, an object-oriented programming language such as the JAVA programming language. However created, the software system can store the objects within a single data structure, and resolve each object (where applicable) in the design to conform to the given design intent. User input at any level of this dataset is thus appropriately propagated to all objects in the data structure consistent with the design intent, thereby resolving the complete design through the resolving of each object individually. These programming objects can include non-functional components as well as functional components, and thus design intents both for physical objects, as well as design intents for non-physical objects or schemes (e.g., wiring, electricity flow directions).

[0045] As mentioned above, each object in the object database 115 can include one or more attributes and product parameters, or reference to one or more databases comprising information of the same. Attributes can include product features and characteristics, such as for example, size, shape, color, type, texture, price, material composition, etc. Product parameters can include rules and behaviors associated with each component. For example, the rules and behaviors can include applicable laws, codes, manufacturing guidelines, and physical limitations. For instance, in the case of an electrical system, the product parameters can include the rules and regulations of applicable national, state, and local codes and regulations.

[0046] In addition to adding functional components (e.g., 120(a-d)), the user can add one or more source components to the design space 110. For example, Figure 1A shows that the user selects the power infeed icon 160, and draws a power infeed 160a in the design space 110. Similar to the functional components, the user can position the source components (e.g., 160a) about or in relation to structural design components (e.g., 150). Upon the user entering the source components into the design space 110, or prior thereto, the software system can receive the input and identify the components. In particular, the software system can identify each component's location, type, size, and relationship to other components using object-oriented software processing techniques. Upon identification of the source component, the software system can create an object (e.g., in object database 115) for each source component that includes one or more attributes and product parameters.

[0047] Upon receiving user input and identifying one or more source components, the software system can automatically populate functional system lines with optimized parameters to power-up the receptacles. The functional system lines can connect the functional components to the source components. For example, Figure 1B illustrates the interface 110 after the software system has populated a functional system line 130a that interconnects the functional components (e.g., receptacles 120(a-d)) and the source components (e.g., power infeed 160a) in the design space 110. In the electrical case, one will appreciate that the functional system line 130a can include any connecters and conduits, such as jumpers, connection boxes, circuit breakers, etc., needed to properly connect the receptacles 120(a-d) and the power infeed 160a.

[0048] One will appreciate that the software system can associate an object (similar to objects in object database 115 described above) to each connector and conduit of the functional system line 130a. Furthermore, the software system can populate the functional system line 130a in view of all of the components within the design space 110 and their associated objects. Thus, the software system can create the functional system line 130a to comply with applicable codes, rules, regulations, manufacturing guidelines, etc.

[0049] In addition to complying with applicable codes, rules, regulations, manufacturing guidelines, functional guidelines, etc., the software system can optimize the functional system line 130a. In other words, the software system can optimize the placement, location, number, dimension, type, run, length, cost, etc. of the functional system line 130a and its connectors and conduits. For example, the

software system can ensure that the functional system lines are populated in a manner that no one circuit is overloaded (e.g., based on the type of power source), and that each functional component is on a circuit that provides it with the proper amps and voltage, etc.

[0050] According to some implementations of the present invention, the wall component 150 can include modular wall components, or modular furniture system components. Furthermore, the connectors and conduits forming the functional system lines 130a can comprise pre-fabricated wires, jumpers, and other electrical connectors. In such implementations, the present invention can optimize the functional system lines to correspond to the pre-fabricated electrical components. Thus, the software system can calculate the run of the functional system lines to correspond with known parameters of products.

[0051] According to yet further implementations, the software system can automatically populate and optimize the functional system lines with respect to two- or three-dimensions. For example, the software system can optimize the run of the functional system lines in length, width, and height. Thus, the software system can reduce the run of a functional system line by slanting or otherwise angling it as it runs up a wall component or along a floor component where possible, such as within an open cavity within the wall or floor. Furthermore as explained in greater detail below, implementations of the present invention provide the ability to automatically render three-dimensional (3D) views of the functional system lines and other components with the design space, and provide compiled take-offs and pricing for all componentry.

[0052] In addition to the creation of functional system lines, the software system can also automatically create flow intent lines. As shown in Figure 1B, the flow intent lines 180(a-d) can visually indicate the connection between each functional component and each source component. For example, flow intent line 180a indicates that receptacle 120a is connected to power infeed 160a through an arched or curved line extending across the design space 110. Flow intent lines 180(b, c, and d) can similarly indicate the electrical connection of receptacles 120(b, c, and d).

[0053] As shown in Figure 1B, the flow intent lines 180(a-d) can provide a readily visible and easily understandable representation of the electrical connections of the components of the functional system in the design space 110. One will appreciate that the more complex the system, the larger the system, or the more components included in the system, the more difficult it can be for a user to identify the electrical connections between functional components and source components based solely upon the functional system lines.

[0054] After automatically populating the functional system lines and/or the flow intent lines, a user can place, remove, relocate, or change of one or more of a non-functional design component, a functional component, or a source component within the design space. For example, Figure 1C shows the user positioning an additional power infeed 160b into the design space 110. Upon receiving additional user input regarding the placement, removal, relocation, or change of one or more components in the design space, the software system can automatically resolve the functional system lines in view of the additional user input.

[0055] For example, Figure 1D illustrates that the software system can automatically modify the functional system layout by optimizing the functional system lines in view of the additional source component 160b. In particular, the software system can modify the functional system layout to connect each receptacle 120(a-d) to the nearest power infeed. Thus, the software system can modify functional system line 130a to connect receptacles 120a and 120b to the power infeed 160a. The software system can also create a new functional system line 130b to connect receptacles 120c and 120d to power infeed 160b. Thus, as illustrated by Figures 1C and 1D, the software system can optimize the functional system lines by, in at least one implementation, reducing the length and number of connectors and conduits connecting the components of the functional system.

[0056] One will appreciate in light of the disclosure herein that, if desired, a user can manually perform each step described herein as being automatically performed by the software system (e.g., placement, modification, addition, and removal of functional system lines and flow intent lines). Additionally, the user can manually change any action taken by the software system. For example, Figure 1E illustrates that the user can reposition one or more of the flow intent lines. This may be desirable for various reasons. For example, the user may want to ensure that all of the receptacles on one side of the wall 150 (or in one workspace, hall, or room etc.) are on one functional system line and all the receptacles on the other side of the wall 150 (or in another workspace, hall, or room etc.).

[0057] Thus, as shown in Figure 1E the user can modify flow intent line 180b so it connects receptacle 120b with power infeed 160b, and modify flow intent

line 180c so it connects receptacle 120c to power infeed 160a. Upon receiving user input regarding the repositioning of one or more flow intent lines, the software system can automatically resolve and re-optimize the functional system lines to correspond to the repositioning or modification of the flow intent lines. For example as shown in Figure 1E, the software system can route functional system line 130b to connect receptacles 120d and 120b to power infeed 160b. Similarly, the software system can route functional system line 130a to connect receptacles 120a and 120c to infeed 160a.

[0058] Figures 2A-2C illustrate additional or alternative features and aspects of the present invention. For example, as seen in Figure 2A a user can import or design a non-electrical component or set of components such as a wall component 150 or work-surface/panel system in a 2D interface 110 of the software. To create the design, the user can select a component of (e.g., Aluminum Framed Wall) from a product menu, and then draw a line or representative shape in the 2D interface. Upon completion of the line or placement of the shape, the software system can automatically apply default attributes, parameters, and options to all of the component(s). If the user introduces additional components to the existing design, each of the intelligent software objects (including objects for both original components and newly introduced components) automatically resolve themselves (or themselves in conjunction with resolution of other related software objects) in the design, providing a continuously resolved solution at all times.

[0059] Thus, when a functional component such as a receptacle 120e, or a non-functional design component, is introduced into the design—again through its

selection from a product menu—that component can resolve itself upon placement in accordance with the other components in the design space. For example, when placed at or near the desired location, the software system can automatically ensure that the receptacle 120e is positioned in a "valid" location.

[0060] For example, the software system can constrain the placement of the receptacle 120e placed by the user in Figure 2A. In particular, based on the attributes and parameters of the objects within the design space 110, the software system can ensure that no functional component is placed in an invalid location. For example, the software system can indentify that the user placed a receptacle 120e near a sink 200. Based on the parameters of the receptacle, the software system can recognize that only GFCI receptacles 140 can be place proximate a sink 200, and thus identify that the position of the receptacle 120e is invalid.

[0061] Upon recognizing or identifying an invalid component, the software system can automatically modify the location, type, connection, or number of the invalid component or other components within the design space 110 to correct the invalid functional component. For example, Figure 2B illustrates that the software system can change the receptacle 120e to a GFCI receptacle 140a to ensure that the component is valid. Figure 2B also illustrates that after resolving the design components, the software system can automatically populate functional system line(s) 130c to connect the receptacle 140a and to the power infeed 160c. Additionally, the software system can automatically generate and display a flow intent line 180 connecting the functional components (e.g., 140a) to the source components (e.g., 160c), as explained in greater detail above.

[0062] In additional or alternative implementations, however, the software system can also or alternatively warn the user about invalid placement of a component, or automatically move the component to the nearest valid location. For example, Figure 2C illustrates an interface 100 in accordance with an implementation of the present invention in which the software system generates a warning 155 against placement of a receptacle 120e in an invalid location in the design space 110. Alternatively or additionally, the software system could move the receptacle 120e to the nearest valid location (i.e., far enough away from the sink 200 to comply with the applicable codes/regulations).

[0063] In general, the term "valid" can have several different connotations in accordance with the present invention. In one implementation with regard to functional components, for example, the term valid can refer to component compatibility and/or compliance with applicable rules, regulation, and codes. In at least one implementation with regard to electrical receptacles, for example, the term valid can refer to relative component accessibility, such that a valid location would be a location for an electrical receptacle on a wall at minimum floor/ceiling/stud distances, and/or that the electrical outlet is not behind some permanent or semi-permanent fixture (e.g., a wall-mounted cabinet). In other cases, the term "valid" can refer to specific electrical configuration issues with regard to electrical wiring and electricity flow or load balance standards. For example, the software system could constrain (or otherwise warn against) placement of an electrical outlet where there is no intended wiring or electrical sources nearby. Similarly, the software

system can constrain or warn against electrical outlets that use the wrong voltage or amperage for a particular appliance, etc.

[0064] Additionally, the attributes and parameters of a functional component can include one or more "Option" objects, such as a "Powered/non-powered" Option. Upon placement of the receptacle on the non-electrical component in the design, the software system can recognize the existence (or lack thereof) of a power feed. If there are no power feeds present, the system can automatically set this option to non-powered and the system can display a warning flag (Outlet is not powered).

[0065] If a power feed type object does exist in the design, it will also have one or more options such as the "Auto-power Option (On/Off)." A user can drive this option upon placement of a component. When a user places a receptacle in a design with a valid power feed, an object for the receptacle can automatically recognize the status of the Auto-power Option, and automatically resolve itself accordingly, such as by automatically connecting (in the user interface) the receptacle to the power source via one or more functional system lines. If the Auto-power Option of the feed is Off, the receptacle Non-powered Option may not change; if Auto-power Option of the feed is "On" the receptacle Powered/Non-powered option can be automatically set to Powered. In the same manner, if an "Auto-power On" feed is introduced to a design that previously had no feed, all applicable receptacles switch their Option to Powered, resolving themselves. The placement of an "Auto-power On" feed (or the changing of an Auto-power Off feed to an Auto-power On feed) also can trigger the creation of one or more flow intent

lines, which according to some implementations are arc lines represented in the 2D graphic interface to indicate power from the source component to the receptacle.

[0066] Implementations of the present invention can also provide a solution level programming object, which recognizes behaviors and relationships between objects within the design, and which allows the software system to resolve itself with each user input. This solution level object recognizes the combined inclusion of the "Auto-power On" feed and receptacle, and understands that this calls for the addition of one or more additional components, such as harnesses or jumpers, to accomplish the power transfer from the power infeed to the receptacles in accordance with the electrical intent. The existence and attributes (dimensional or otherwise) of such harnesses and jumpers and other similar components can also be dependant on the attributes and parameters of the non-functional components within the design space, and they can resolve themselves accordingly.

[0067] Thus, when a user introduces an electrical component such as a receptacle into the design—again through its user selection from a product menu—that component can automatically resolve itself in the user interface upon placement in accordance with the electrical intent. Upon placement of the receptacle on a non-electrical component in the design, the receptacle can recognize the existence (or lack thereof) of a power feed. If there are no power feeds present, this option is automatically set to non-powered and a warning flag is displayed (Outlet is not powered).

[0068] As discussed above, according to some implementations of the present invention, a software system can automatically populate functional system

lines, automatically populate flow intent lines, and automatically resolve any invalid components with the design space based on object attributes and parameters associated with each component within the design space. Additionally, the software system can automatically place one or more functional components within a design space at valid placement locations. For example, Figure 3A illustrates a design space 110 in which a user places one or more source components 160c and 160d. One will appreciate at this point the design space does not include any functional components.

[0069] As shown in Figure 3B, the user can then request the software system to automatically place functional components within the design space. Upon receiving user input regarding one or more functional components (i.e., the request by the user to place functional components), the software system can automatically populate both functional components and functional system lines connecting them. For example, Figure 3C illustrates that the software system can place a plurality of receptacles 120(f-k) within the design space about a modular wall 105a defining a workspace, and connect them by functional system lines 130(d-e).

[0070] One will appreciate in light of the disclosure herein that the placement of the receptacles 120(f-k) can be based on the attributes and parameters of the functional and non-functional components within the design space 110. For example, Figure 3C illustrates that the software system can ensure that each wall of the workspace has at least one receptacle, that receptacles are placed near desk 203, that receptacles are not placed behind wall mounted furniture 202, 204, and 205, and that the placement of each receptacle is otherwise valid.

[0071] Additionally, the software system can automatically populate and optimize functional system lines 130d, 130e connecting the receptacles 120(f-k) with power infeeds 160d, 160e. Furthermore, the software system can automatically create and place flow intent lines 180(f-k) that indicate the flow intent of the functional system layout.

[0072] While the interface 100 shown in Figures 1A-3C is generally—but not without exception—a two dimensional plan view interface, the software system can also represent components in a three-dimensional interface, in which the a user or the software system can make placement and changes to the design space directly. Specifically, changes in a two-dimensional interface are automatically implemented when pulled up in a three-dimensional interface. This is possible at least in part since all interfaces of the software can utilize the same database, and further since the system has already previously rendered image information associated with each component that the user selects within the interface 110. As such, the software system can represent a single dataset as two-dimensional and three-dimensional graphical representations instantly and interchangeable, as well as represent textual representations.

[0073] As a result, auditing between the various outputs is not ordinarily required. Rather, validation of the design in one view confirms accuracy of the design in the other views as well. This means that pricing and product take-off lists are also accurate due to the elimination of any additional human adjustments or guesses that would otherwise need to be made based on the graphical views.

[0074] At any time before, during, or after design layout of any componentry, a user can open a separate 3D interface or textual interface (a Bill of Materials, Order Proposal, parts list, reports, etc.). For example, the present invention can include additional and alternative systems and methods for displaying automatically resolved user input in two or more dimensional views in a realistic manner. Such systems and methods are described in commonly-assigned U.S. Patent Application No. 11/204,419, filed August 16, 2005, now U.S. Patent No. 7,249,005, entitled "Design software incorporating efficient 3-D rendering." The entire contents of the U.S. Patent No. 7,249,005 are incorporated by reference herein.

[0075] Similarly, the present invention can include additional and alternative systems and methods for the continual generation of an accurate parts list along with the user input, which the software system can display in a parts list interface, and which does not need further review for correction or additional parts before ordering. Such systems and methods are described in commonly-assigned, and above referenced, U.S. Patent Application No. 11/204,421, filed on August 16, 2005, entitled "Capturing a User's Intent in Design Software."

[0076] Upon the opening of a 3D or textual interface (and dynamically in unison with additional user input) the system can automatically represent the design components by 3D or textual depiction in the appropriate interface. According to some implementations of the present invention, the 3D rendering or depiction of the design space can include flow intent lines that visually show the connection of each functional component to a source component. Alternatively, the flow intent lines

can be hidden or not created in the 3D rendering to allow the most accurate representation of how the design space will look when constructed. Additionally or alternatively, according to some implementations of the present invention, the user can activate or deactivate the flow intent lines. Thus, a user can decide whether the flow intent lines are created and displayed in either the 2D or 3D interface.

[0077] One will appreciate in light of the disclosure herein that the 2D, 3D, and textual interfaces can update accordingly with each user input, as each design component and/or functional system line resolves itself. Other non-physical component objects, such as, but not limited to, a product price, will also resolve themselves in accordance with related objects, and the system may or may not represent them in one or more of the graphical or textual interfaces.

[0078] Thus, a user could create a space only superficially in a three-dimensional view and, for example, place one or more functional components, such as electrical receptacles, in the design space. The software system can then automatically move the functional components into an appropriate or valid location (if necessary) based on the location of a source component or any other components in the design space. After ensuring the all components are in valid locations and configurations, the software system can connect only the appropriate conduits and connectors between the source component and the functional components.

[0079] The software system can further automatically maintain an ongoing, accurate parts list that lists each functional component, each conduit and connector, and each source component that would may be required to produce a valid, optimized functional system layout. For example, in the case of an electrical system

layout, the software system can maintain an ongoing, accurate parts list including each receptacle component, each wire (by specific dimension in terms of gauge, inches, feet, etc.), each power source, and any other jumpers, connectors, or other components needed. Thus, the user can have confidence that, should they submit an order for the three-dimensional design the user has just made, that all diagrams, dimensions, and components submitted to any entity with the order will be appropriate and up to applicable or chosen electrical code(s) and guidelines.

[0080] In addition to the foregoing, implementations of the present invention can also be described in terms of flowcharts comprising one or more acts in a method for accomplishing a particular result. For example, Figure 4 illustrates a method of automatically configuring user input in a computerized environment to create a valid functional system layout. Similarly, Figure 5 illustrates a method of automatically configuring user input in a computerized environment to populate functional system lines between functional components and source components. The acts of Figures 4 and 5 are discussed more fully below with respect to the components and diagrams of Figures 1A-3C.

[0081] For example, Figure 4 shows that a method in accordance with an implementation of the present invention can comprise an act 400 of identifying one or more source components in a design space. Act 400 can include identifying the location and/or object attributes and parameters of the source components in the design space. For example, as shown in Figure 1A, the interface 100 can receives one or more inputs from a user into the design space regarding a source component (e.g., power infeed 160a). Similarly, Figure 1C shows that the interface 100

receives user input regarding the location of an additional source component (e.g., power infeed 160b) placed within the design space 110.

[0082] Figure 4 also shows that a method in accordance with an implementation of the present invention can comprise an act 410 of receiving user input regarding one or more functional components configured to receive flow from the one or more source components. Act 410 can include receiving input from a user that corresponds to placing, removing, repositioning, or changing one or more functional components within the design space. For example, Figure 1A shows the interface 100 receives user input regarding positioning a functional component (e.g., electrical receptacle 120a) within the design space 110. Alternatively, act 410 can comprise receiving input from a user requesting placement of functional components within a design space. For example, Figure 3B shows the interface 100 receiving user input requesting the placement of functional component (e.g., electrical receptacles) within the design space 110.

[0083] In addition, Figure 2 shows that a method in accordance with an implementation of the present invention can comprise an act 420 of automatically creating a valid functional system layout within the design space by optimizing the parameters of functional system lines connecting the one or more source components and the one or more functional components. Act 420 can include populating and optimizing the placement, location, number, dimension, type, and/or length of functional system lines connecting the one or more functional components and one or more source components in a design space. For example, Figure 1B shows the interface 100 automatically placing functional conduits and connectors

130a between a source component (e.g., power infeed 160a) and the functional components (e.g., electrical receptacles 120(a-d)). Act 420 can also include adding, removing, or modifying functional conduits and connectors between functional components. For example, Figure 1D, shows the interface 100 modifying the functional conduits and connectors 130a, and adding functional conduits and connectors 130b.

[0084] By contrast, Figure 5 illustrates that an additional or alternative method in accordance with an implementation of the present invention can comprise an act 500 of receiving user input regarding the placement of one or more functional components within a design space. Act 500 can include receiving user input with regard to placing, removing, or changing the placement of functional components within a design space. For example, Figure 1A shows the interface 100 receiving user input indicating the placement of functional components (e.g., electrical receptacles 120(a-d)) within the design space 110.

[0085] Figure 5 also shows that a method in accordance with an implementation of the present invention can comprise an act 510 of receiving user input regarding the placement of one or more source components within the design space. Act 510 can include receiving user input with regard to placing, removing, or changing the placement of source components within a design space. For example, Figure 1A shows the interface 100 receiving input from a user positioning a source component (e.g., power infeed 160a) within the design space 110. Alternatively, as shown in Figure 1C, the interface 100 can receive user input regarding the placement of an additional source component (e.g., power infeed 160b) within the design space 110.

[0086] In addition, Figure 5 shows that a method in accordance with an implementation of the present invention can comprise an act 520 of populating functional system lines with optimized characteristics, the functional system lines connecting the one or more source components and the one or more functional components. Act 520 can include populating and optimizing the placement, location, number, dimension, type, and/or length of functional system lines connecting the one or more functional components in a design space. For example, Figure 1B shows that interface 100 automatically places functional conduits and connectors 130a between a source component (e.g., power infeed 160a) and functional components (e.g., electrical receptacles 120(a-d)).

[0087] Furthermore, Figure 5 shows that the method in accordance with an implementation of the present invention can comprise an act 530 of receiving additional user input regarding the placement, relocation, or change of one or more of a non-functional design component, a functional component, or a source component within the design space. Act 530 can include receiving additional user input that can change or modify any of the design components within the design space, or add additional ones. For example, Figure 1C shows the interface 100 can receive user input regarding the placement of an additional source component (e.g., power infeed 160b) within the design space 110.

[0088] Still further, Figure 5 shows a method in accordance with an implementation of the present invention can comprise an act 540 of resolving the functional system lines in view of the additional user input. Act 540 can include adding, removing, or modifying functional system lines in view of the additional

user input to maintain validity and optimization of the functional system layout. For example, Figure 1D shows the interface 100 modifying the functional conduits and connectors 130a, and adding functional conduits and connectors 130b.

[0089] Accordingly, the diagrams and methods provided herein illustrate a number of ways and configurations in which design software can automatically adjust prior, present, and/or future user inputs to create an accurate depiction of a design space. In particular, the design software in accordance with the present invention can continually resolve user input, as well as to continually resolve appropriate positioning of user input in real-time. Furthermore, the design software in accordance with the present invention can accomplish these ends while maintaining an accurate parts list for each of the items placed in a given design space. Thus, implementations of the present invention can represent an effective and efficient means for designing any interior and/or exterior space, and ultimately for constructing the same.

[0090] The embodiments of the present invention may comprise a special purpose or general-purpose computer including various computer hardware components, as discussed in greater detail below. Embodiments within the scope of the present invention also include computer-readable media for carrying or having computer-executable instructions or data structures stored thereon. Such computer-readable media can be any available media that can be accessed by a general purpose or special purpose computer.

[0091] By way of example, and not limitation, such computer-readable media can comprise RAM, DRAM, ROM, EEPROM, CD-ROM or other optical disk

storage, magnetic disk storage or other magnetic storage devices, or any other medium which can be used to carry or store desired program code means in the form of computer-executable instructions or data structures and which can be accessed by a general purpose or special purpose computer. When information is transferred or provided over a network or another communications connection (either hardwired, wireless, or a combination of hardwired or wireless) to a computer, the computer properly views the connection as a computer-readable medium. Thus, any such connection is properly termed a computer-readable medium. Combinations of the above should also be included within the scope of computer-readable media.

[0092] Computer-executable instructions comprise, for example, instructions and data which cause a general purpose computer, special purpose computer, or special purpose processing device to perform a certain function or group of functions. Although the subject matter has been described in language specific to structural features and/or methodological acts, it is to be understood that the subject matter defined in the appended claims is not necessarily limited to the specific features or acts described above. Rather, the specific features and acts described above are disclosed as example forms of implementing the claims.

[0093] The present invention may be embodied in other specific forms without departing from its spirit or essential characteristics. The described embodiments are to be considered in all respects only as illustrative and not restrictive. The scope of the invention is, therefore, indicated by the appended claims rather than by the foregoing description. All changes which come within the meaning and range of equivalency of the claims are to be embraced within their scope.

CLAIMS

We claim:

1. A method of automatically configuring user input in a computerized environment to represent a design or layout of components within a design space, comprising:

identifying one or more source components in a design space;

receiving user input regarding one or more functional components configured to receive flow from the one or more source components; and

automatically creating a valid functional system layout within the design space by optimizing the characteristics of functional system lines connecting the one or more source components and the one or more functional components.

2. The method as recited in claim 1, further comprising:

receiving additional user input regarding the placement of one or more additional source components in the design space; and

automatically modifying the functional system layout by optimizing the characteristics of the functional system lines in view of the additional user input.

3. The method as recited in claim 1, wherein automatically creating a valid functional system layout comprises populating all required functional connectors and conduits needed to validly connect the one or more source components and the one or more functional components in compliance with applicable codes, rules, and manufacturing guidelines.

4. The method as recited in claim 1, wherein optimizing the characteristics of functional system lines comprises optimizing the placement of the functional system lines.

5. The method as recited in claim 1, wherein optimizing the parameters of functional system lines comprises calculating an optimal run of connectors and conduits between the one or more functional source components and the one or more functional components.

6. The method as recited in claim 5, further comprising calculating the optimal run based on attributes of the functional connectors and the one or more functional components.

7. The method as recited in claim 1, further comprising automatically populating flow intent lines in the design space indicating the connections between the one or more functional components and the one or more source components.

8. The method as recited in claim 7, further comprising:

receiving user input regarding the repositioning of one or more flow intent lines; and

resolving the optimized functional system lines to correspond with the repositioning of the one or more flow intent lines.

9. The method as recited in claim 1, wherein automatically creating a valid functional system layout further comprises automatically resolving any invalid functional components.

10. The method as recited in claim 9, further comprising modifying the location, type, connection, or number of the invalid functional components.

11. The method as recited in claim 1, wherein receiving user input regarding one or more functional components comprises receiving user input regarding the placement of a functional component within the design space.

12. The method as recited in claim 11, further comprising constraining the placement of the functional component to a valid location.

13. The method as recited in claim 12, further comprising determining the valid location based upon the attributes and parameters of the functional component and other components within the design space.

14. The method as recited in claim 1, further comprising automatically placing one or more functional components within the design space at valid placement locations in the design space upon receiving the user input regarding the one or more functional components.

15. The method as recited in claim 1, wherein the functional system comprises an electrical system.

16. The method as recited in claim 15, wherein the one or more functional components comprise one or more of an electrical receptacle, a junction box, circuit breaker, or electrical wiring.

17. The method as recited in claim 1, further comprising generating a parts list in real-time including all the components of the valid functional system layout, wherein the parts list updates in real-time when any changes are made to the valid functional system layout.

18. The method as recited in claim 1, further comprising rendering the valid functional system layout in 2D graphics, 3D graphics, and textual information, each of which are separate representations of the same dataset.

19. A method of automatically configuring user input in a computerized environment to represent a design or layout of components within a design space, comprising:

receiving user input regarding the placement of one or more functional components within a design space;

receiving user input regarding the placement of one or more source components within the design space;

populating functional system lines with optimized characteristics, the functional system lines connecting the one or more source components and the one or more functional components;

receiving additional user input regarding the placement, relocation, or change of one or more of a non-functional design component, a functional component, or a source component within the design space; and

resolving the functional system lines in view of the additional user input.

20. The method as recited in claim 19, further comprising resolving any invalid functional components generated by the user input or additional user input.

21. The method as recited in claim 19, further comprising generating a warning upon receiving user input that positions the one or more functional components at an invalid location.

22. The method as recited in claim 19, wherein populating functional system lines with optimized characteristics comprises calculating optimal placement of the functional system lines between the one or more functional components in three-dimensions.

23. The method as recited in claim 19, further comprising automatically populating flow intent lines in the design space indicating the connections between the one or more functional components and the one or more source components.

24. The method as recited in claim 19, wherein the one or more functional components comprise components of a heating, ventilating, and air conditioning system.

25. In a computerized environment in which a user interface of a design program is used to design one or more interior or exterior spaces, a computer program product having computer-executable instructions stored thereon that, when executed, cause one or more processors in the computerized environment to perform a method of automatically configuring user input in a computerized environment to represent a design or layout of components within a design space, comprising the following:

identifying one or more source components in a design space;

receiving user input regarding one or more functional components configured to receive flow from the one or more source components; and

automatically creating a valid functional system layout within the design space by optimizing the characteristics of functional system lines connecting the one or more source components and the one or more functional components.

5

10

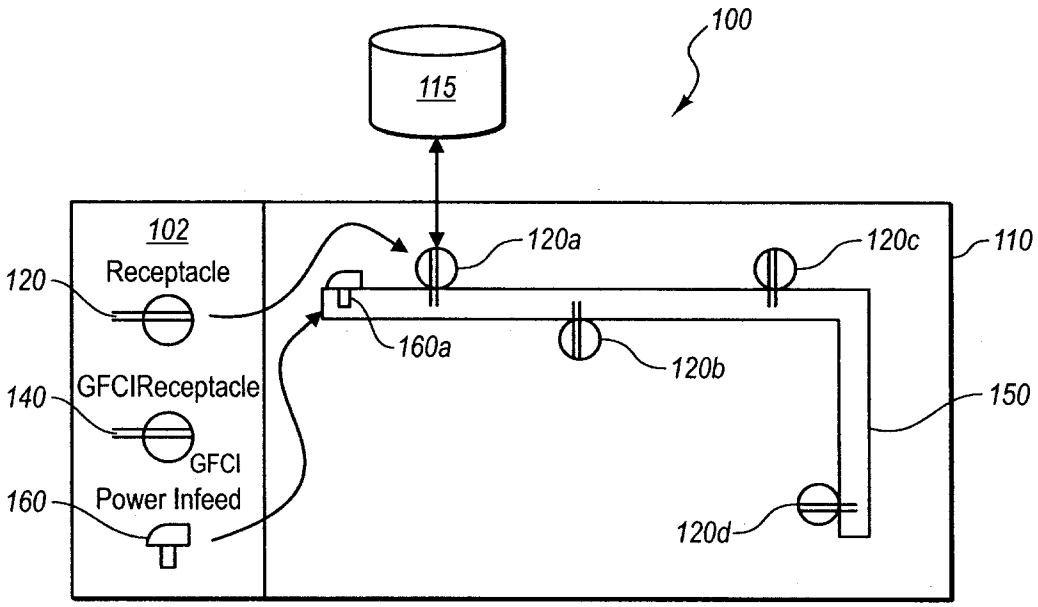

100
115
102
Receptacle
120
GFCIReceptacle
140
GFCI
Power Infeed
160
120a
120c
110
160a
120b
150
120d

FIG. 1A

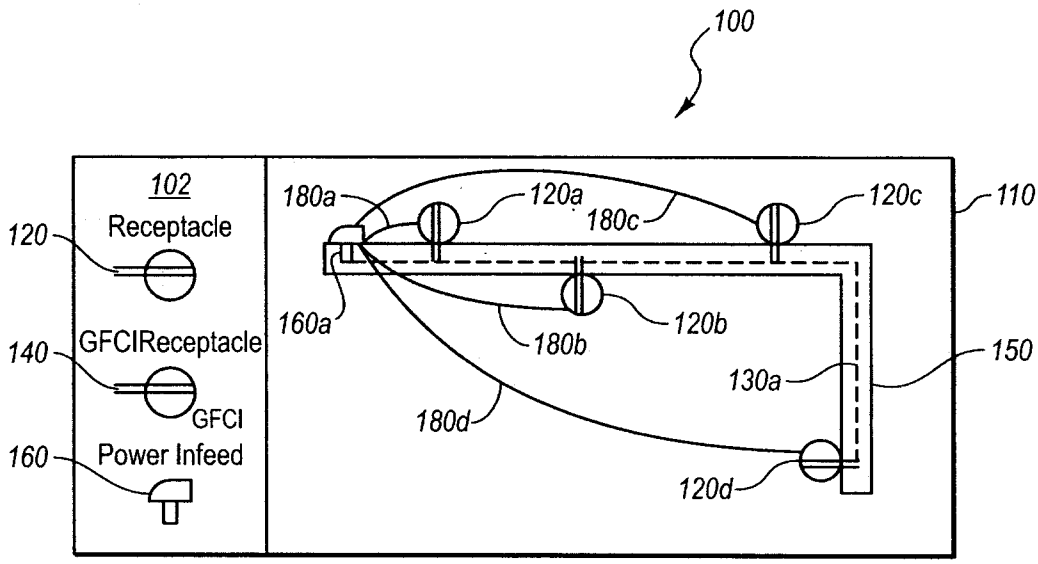

100
102
Receptacle
120
GFCIReceptacle
140
GFCI
Power Infeed
160
180a
120a
180c
120c
110
160a
120b
180b
150
130a
180d
120d

FIG. 1B

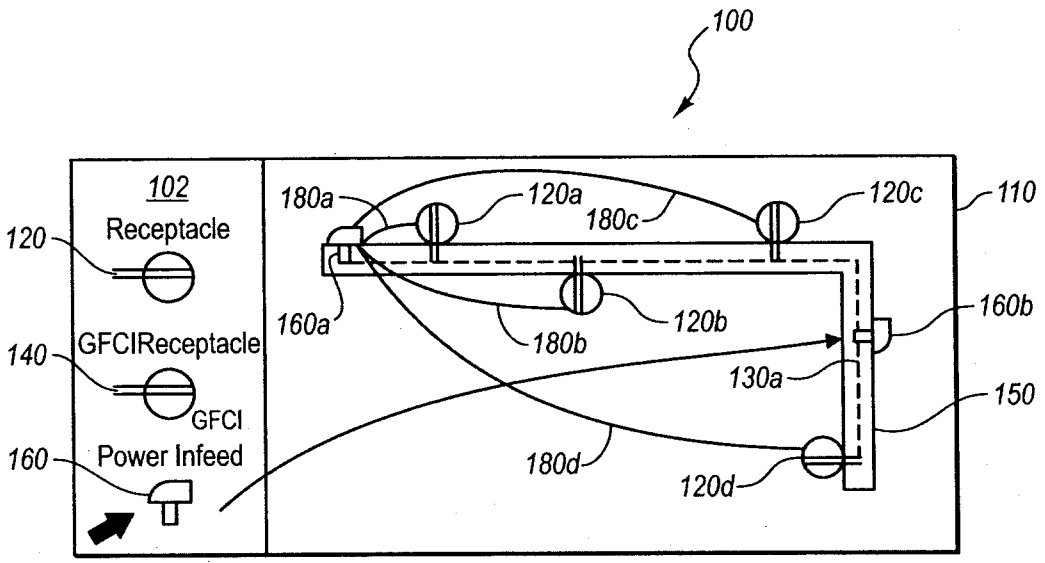

100
102
Receptacle
120
GFCIReceptacle
140
GFCI
160
Power Infeed
180a
120a
180c
120c
110
160a
180b
120b
160b
130a
150
180d
120d

FIG. 1C

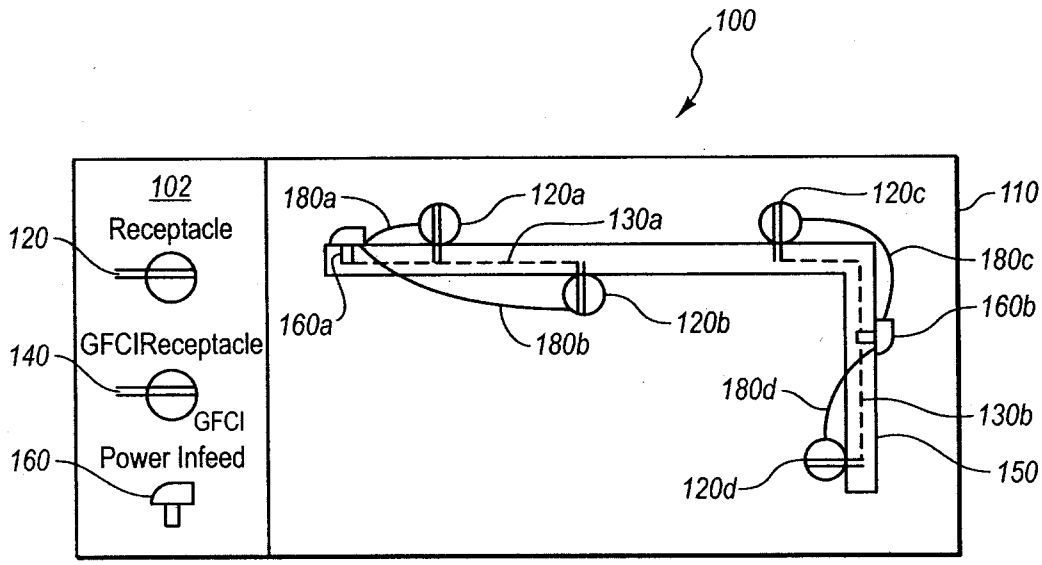

100
102
Receptacle
120
GFCIReceptacle
140
GFCI
160
Power Infeed
180a
120a
130a
120c
110
180c
160b
160a
180b
120b
180d
130b
150
120d

FIG. 1D

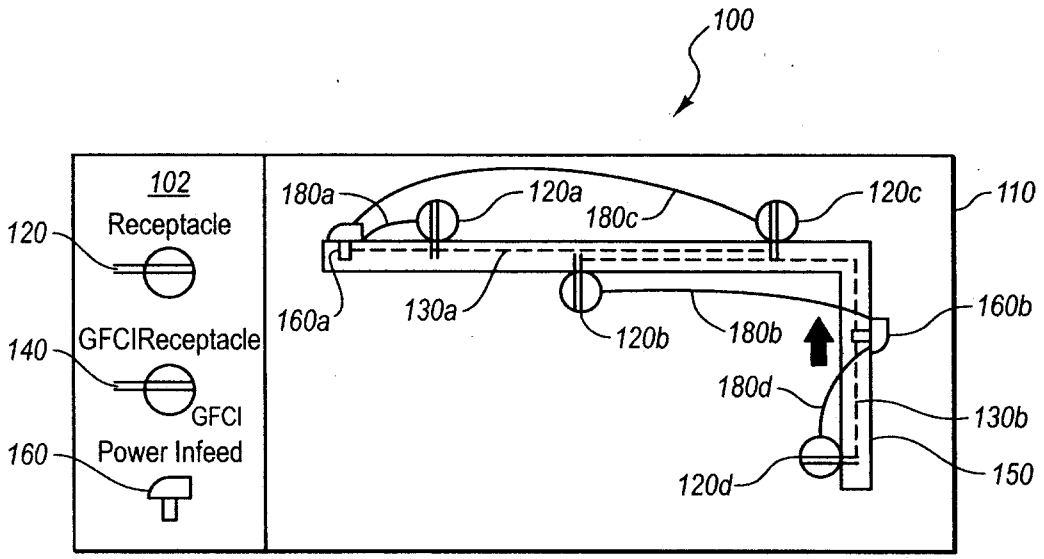

100
102
Receptacle
120
GFCIReceptacle
140
GFCI
160
Power Infeed
180a
120a
180c
120c
110
160a
130a
120b
180b
160b
180d
130b
120d
150

FIG. 1E

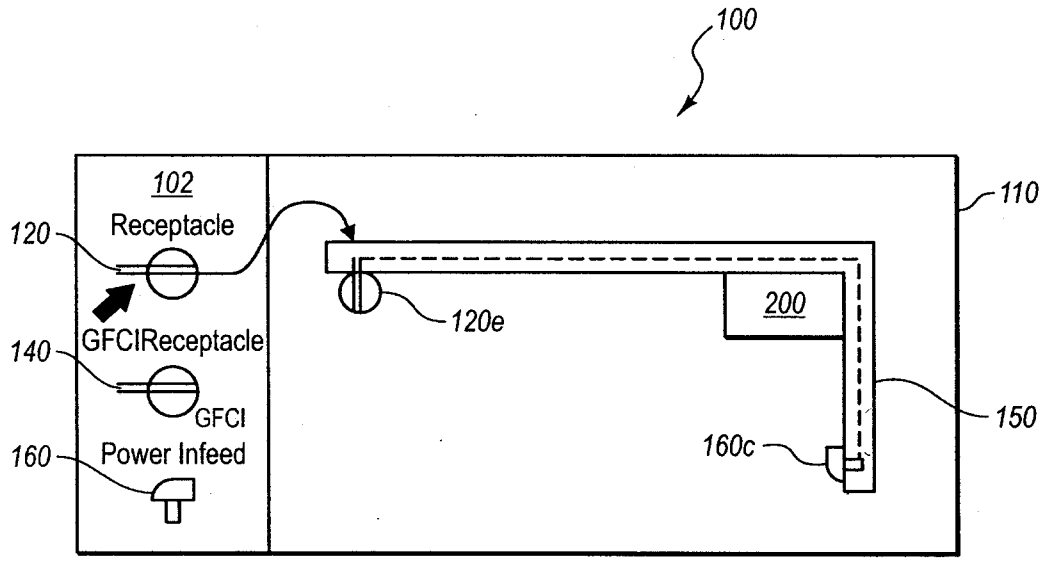

100
102
Receptacle
120
GFCIReceptacle
140
GFCI
160
Power Infeed
110
120e
200
150
160c

FIG. 2A

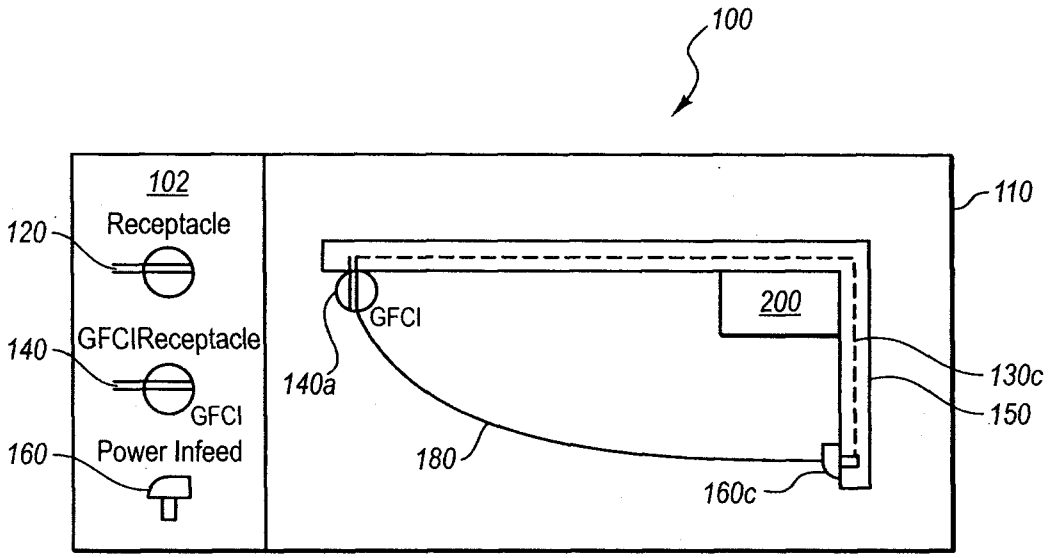

100
102
Receptacle
120
GFCIReceptacle
140
GFCI
160
Power Infeed
110
GFCI
140a
200
130c
150
180
160c

FIG. 2B

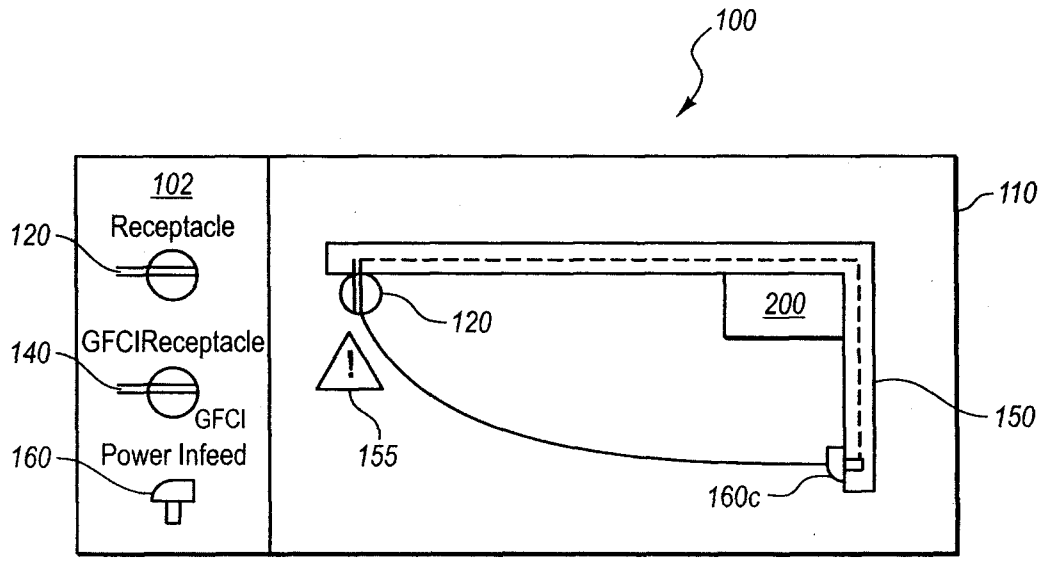

100
102
Receptacle
120
GFCIReceptacle
140
GFCI
160
Power Infeed
110
120
200
150
155
160c

FIG. 2C

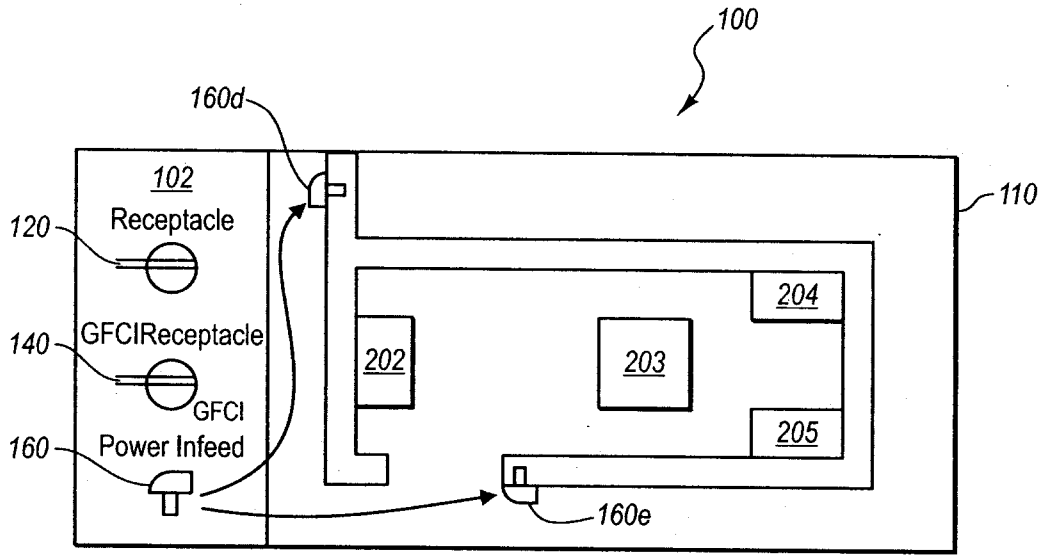

100
160d
102
Receptacle
120
GFCIReceptacle
140
GFCI
160
Power Infeed
110
204
202
203
205
160e

FIG. 3A

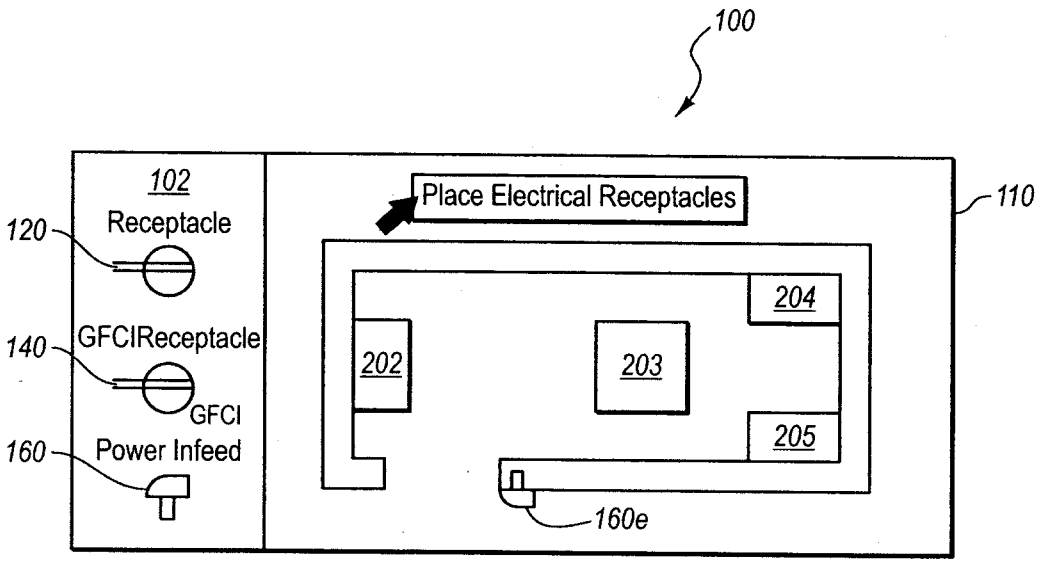

100
102
Receptacle
120
GFCIReceptacle
140
GFCI
160
Power Infeed
Place Electrical Receptacles
110
204
202
203
205
160e

FIG. 3B

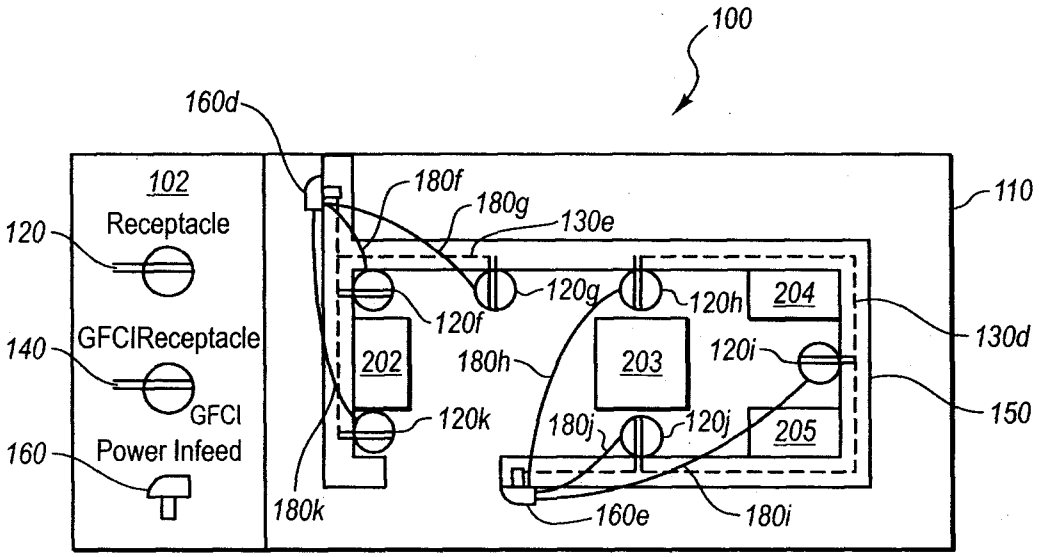

100
160d
102
Receptacle
120
GFCIReceptacle
140
GFCI
Power Infeed
160
180k
180f
180g
130e
120f
120g
120h
204
110
130d
202
180h
203
120i
120k
180j
120j
205
150
160e
180i

FIG. 3C

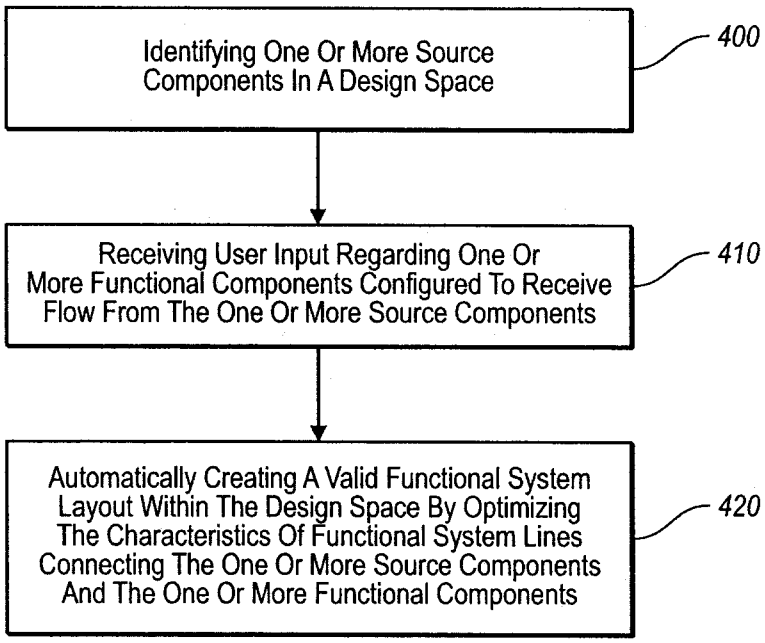

Identifying One Or More Source Components In A Design Space
400
Receiving User Input Regarding One Or More Functional Components Configured To Receive Flow From The One Or More Source Components
410
Automatically Creating A Valid Functional System Layout Within The Design Space By Optimizing The Characteristics Of Functional System Lines Connecting The One Or More Source Components And The One Or More Functional Components
420

FIG. 4

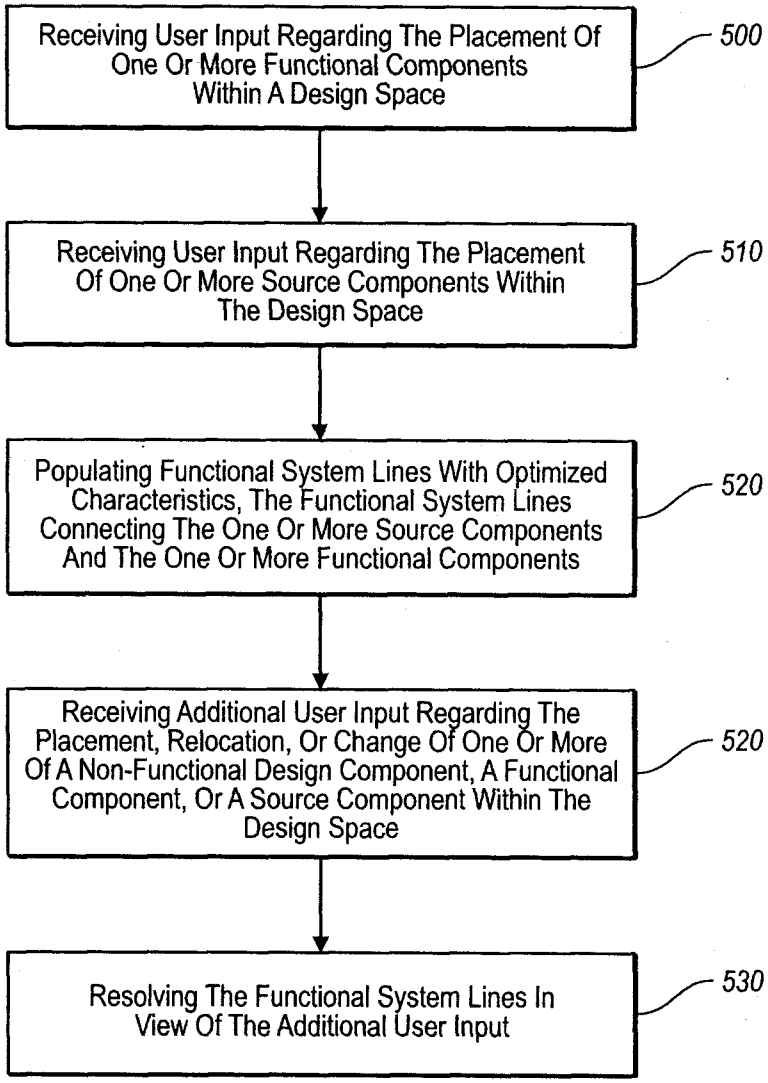

Receiving User Input Regarding The Placement Of One Or More Functional Components Within A Design Space
500
Receiving User Input Regarding The Placement Of One Or More Source Components Within The Design Space
510
Populating Functional System Lines With Optimized Characteristics, The Functional System Lines Connecting The One Or More Source Components And The One Or More Functional Components
520
Receiving Additional User Input Regarding The Placement, Relocation, Or Change Of One Or More Of A Non-Functional Design Component, A Functional Component, Or A Source Component Within The Design Space
520
Resolving The Functional System Lines In View Of The Additional User Input
530

FIG. 5

INTERNATIONAL SEARCH REPORT

International application No.
PCT/CA2009/000190

A. CLASSIFICATION OF SUBJECT MATTER
IPC: ***G06F 17/50*** (2006.01)
According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)
IPC: ***G06F 17/50*** (2006.01)

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic database(s) consulted during the international search (name of database(s) and, where practicable, search terms used)
Databases: Delphion, Google Patents, Canadian Patent Database, IEEE Xplore
Keywords: building, layout, design space, automatic, optimize, valid, source/functional components, distribution system

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	WO 90/03618 (Madden et al.) 5 April 1990 (05-04-1990) * abstract; claims; page 12, lines 7-19 *	1-25
X	WO 93/22741 (Normann et al.) 11 November 1993 (11-11-1993) * entire document *	1-25
X	DeJesus, E. et al. "PEARL: an expert system for power supply layout" 23rd ACM/IEEE Design Automation Conference, 29 June - 2 July 1986, Proceedings, IEEE, (US), paper 34.4, pages 615-621 * page 615, col. 2, lines 18-26; page 617, col. 2, lines 6-14 * * page 618, col. 2 et seq.; section "Placement User Interface"	1-25

[ ] Further documents are listed in the continuation of Box C.

[X] See patent family annex.

* Special categories of cited documents :

"A" document defining the general state of the art which is not considered to be of particular relevance

"E" earlier application or patent but published on or after the international filing date

"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)

"O" document referring to an oral disclosure, use, exhibition or other means

"P" document published prior to the international filing date but later than the priority date claimed

"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention

"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone

"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art

"&" document member of the same patent family

Date of the actual completion of the international search	Date of mailing of the international search report
31 March 2009 (31-03-2009)	5 June 2009 (05-06-2009)

Name and mailing address of the ISA/CA	Authorized officer
Canadian Intellectual Property Office Place du Portage I, C114 - 1st Floor, Box PCT 50 Victoria Street Gatineau, Quebec K1A 0C9 Facsimile No.: 001-819-953-2476	Nicholas Tsagaris 819- 953-0790

INTERNATIONAL SEARCH REPORT
Information on patent family members

International application No.
PCT/CA2009/000190

Patent Document Cited in Search Report	Publication Date	Patent Family Member(s)	Publication Date
WO 9003618A1	05-04-1990	AU 4330489A	18-04-1990
		GB 8822739D0	23-11-1988
WO 9322741A2	11-11-1993	AU 8395491A	04-02-1992
		CA 2087103A1	13-01-1992
		CA 2087103C	12-10-1999
		CA 2134645A1	11-11-1993
		CA 2134645C	18-04-2000
		US 5227983A	13-07-1993
		US 5557537A	17-09-1996
		US 5808905A	15-09-1998
		US 6131077A	10-10-2000
		US 6662144B1	09-12-2003
		US 6701288B1	02-03-2004
		US 2004128116A1	01-07-2004
		WO 9201262A1	23-01-1992
		WO 9322741A3	09-12-1993