Office de la Propriété Intellectuelle du Canada
Un organisme d'Industrie Canada

Canadian Intellectual Property Office
An agency of Industry Canada

CA 2486925 C 2010/07/13

(11)(21) 2 486 925

(12) BREVET CANADIEN
CANADIAN PATENT

(13) C

(86) Date de dépôt PCT/PCT Filing Date: 2003/04/29
(87) Date publication PCT/PCT Publication Date: 2003/11/13
(45) Date de délivrance/Issue Date: 2010/07/13
(85) Entrée phase nationale/National Entry: 2004/11/26
(86) N° demande PCT/PCT Application No.: IB 2003/001811
(87) N° publication PCT/PCT Publication No.: 2003/093989
(30) Priorité/Priority: 2002/04/30 (US10/135,328)

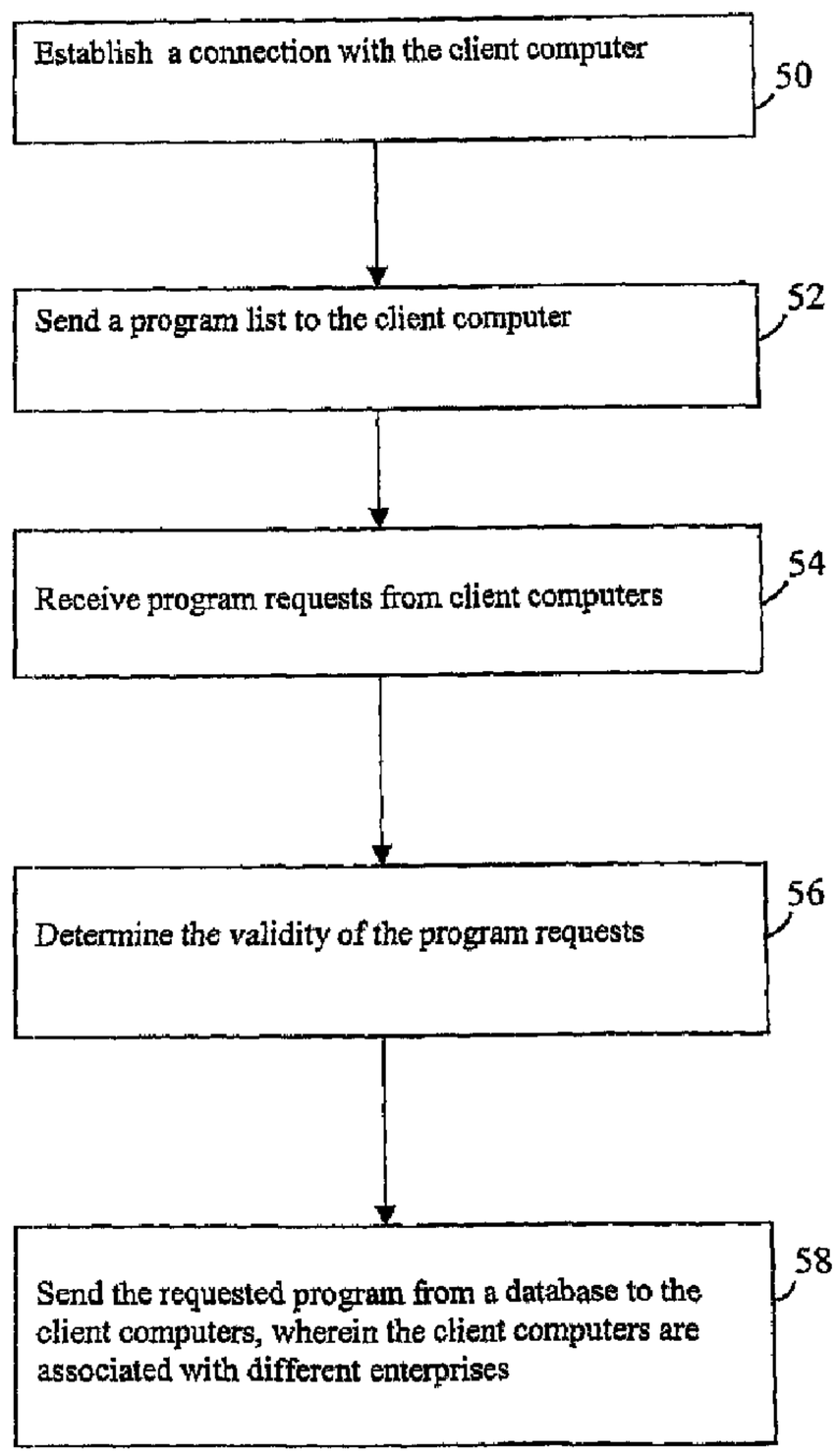
(51) Cl.Int./Int.Cl. *G06F 9/445* (2006.01)

(72) Inventeurs/Inventors:
CHENELLE, MICHAEL A., US;
GAUGHAN, TIMOTHY A., US;
GOWLAND, RUSSELL H., US;
GRATTO, STEVEN E., US;
HENRY, PETER J., US;
KAVLICK, JEFFREY J., US;
KLINE, PAUL A., US;
...

(73) Propriétaire/Owner:
ACCENTURE GLOBAL SERVICES GMBH, CH

(74) Agent: RIDOUT & MAYBEE LLP

(54) Titre : GESTION DE LOGICIEL
(54) Title: COMPUTER SOFTWARE MANAGEMENT



(57) Abrégé/Abstract:
Management of computer software in an enterprise includes receiving program requests from client computers associated with different enterprises. The program requests are validated and at one least requested program is sent from a database to at least one client computer.

CA 2486925 C 2010/07/13

(11)(21) **2 486 925**

(13) **C**

(72) **Inventeurs(suite)/Inventors(continued)**: NUGENT, DAN A., US; O'CONNOR, BRUCE L., US; SIVO, MICHAEL J., US

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

(19) World Intellectual Property Organization
International Bureau

(43) International Publication Date
13 November 2003 (13.11.2003)

PCT

(10) International Publication Number
WO 2003/093989 A3

(51) International Patent Classification[7]: G06F 9/445

(21) International Application Number:
PCT/IB2003/001811

(22) International Filing Date: 29 April 2003 (29.04.2003)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:
10/135,328 30 April 2002 (30.04.2002) US

(71) Applicant: **ACCENTURE GLOBAL SERVICES GmbH** [CH/CH]; Geschäftshaus Herrenacker 15, CH-8200 Schaffhausen (CH).

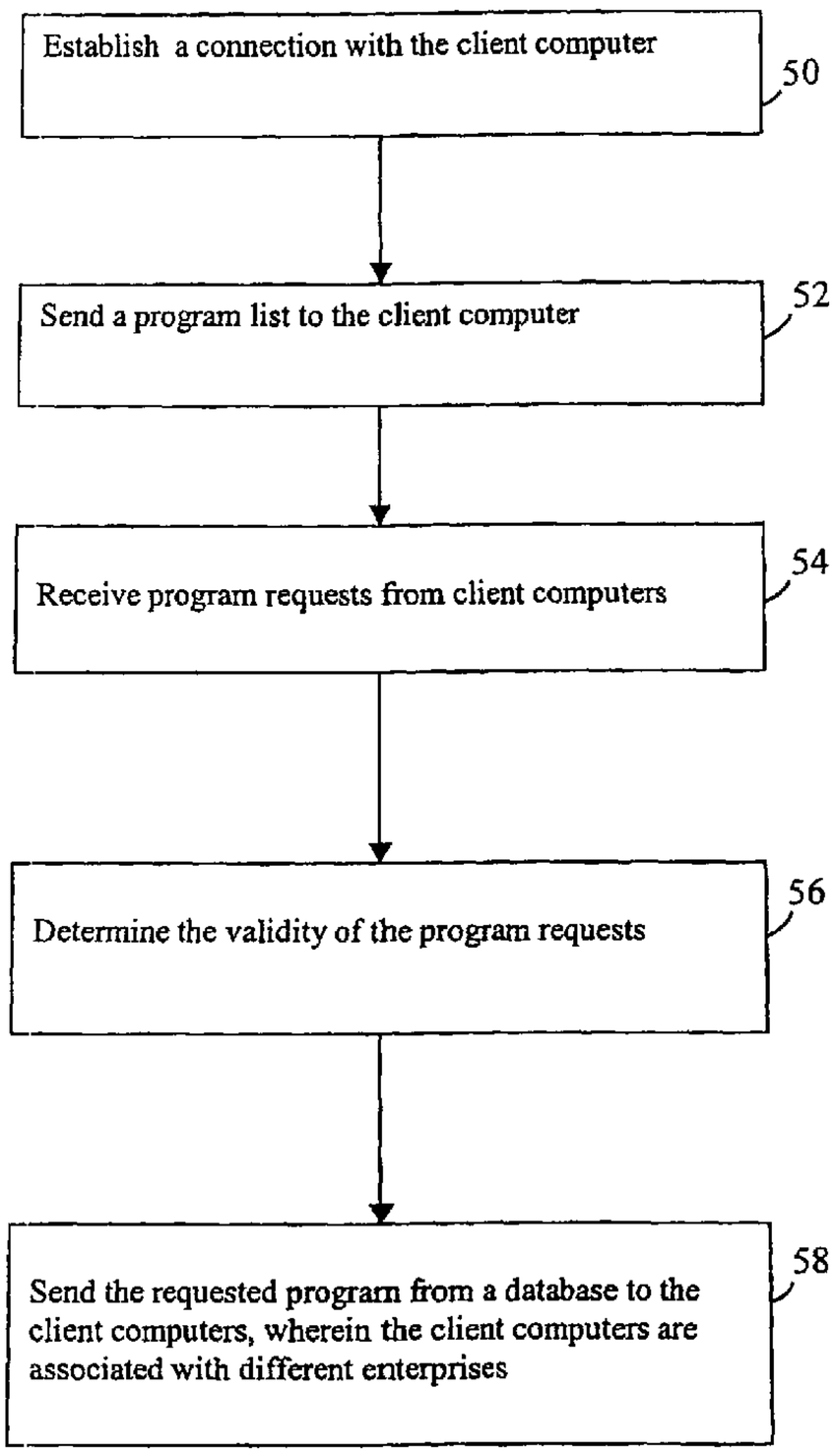
(72) Inventors: **CHENELLE, Michael, A.**; 6631 Vrooman Road, Painesville, OH 44077 (US). **GAUGHAN, Timothy, A.**; 3 Sentinel Road, Washington Crossing, PA 18977 (US). **GOWLAND, Russell, H.**; 6108 Courtside, Midland, MI 48642 (US). **GRATTO, Steven, E.**; 17330 Buckthorn Place, Chagrin Falls, OH 44023 (US). **HENRY, Peter, J.**; 9415 Woodchip Lane, Broadview Heights, OH 44147 (US). **KAVLICK, Jeffrey, J.**; 675 Deepwoods Drive, Aurora, OH 44202 (US). **KLINE, Paul, A.**; 6113 Partridge, Midland, MI 48640 (US). **NUGENT, Dan, A.**; 1777 Bridgewater Ct., Canton, MI 48188 (US). **O'CONNOR, Bruce, L.**; 8890 Vincent Avenue, Alto, MI 49302 (US). **SIVO, Michael, J.**; 4815 Lake Road, Sheffield Lake, OH 44054 (US).

(74) Agents: **MUSKER, David, Charles** et al.; R.G.C. Jenkins & Co., 26 Caxton Street, London SW1H 0RJ (GB).

(81) Designated States *(national)*: AE, AG, AL, AM, AT, AU, AZ, BA, BB, BG, BR, BY, BZ, CA, CH, CN, CO, CR, CU, CZ, DE, DK, DM, DZ, EC, EE, ES, FI, GB, GD, GE, GH, GM, HR, HU, ID, IL, IN, IS, JP, KE, KG, KP, KR, KZ, LC, LK, LR, LS, LT, LU, LV, MA, MD, MG, MK, MN, MW,

[Continued on next page]

(54) Title: COMPUTER SOFTWARE MANAGEMENT



(57) Abstract: Management of computer software in an enterprise includes receiving program requests from client computers associated with different enterprises. The program requests are validated and at one least requested program is sent from a database to at least one client computer.

WO 2003/093989 A3

WO 2003/093989 A3

MX, MZ, NI, NO, NZ, OM, PH, PL, PT, RO, RU, SC, SD, SE, SG, SK, SL, TJ, TM, TN, TR, TT, TZ, UA, UG, UZ, VN, YU, ZA, ZM, ZW.

(84) **Designated States** *(regional)*: ARIPO patent (GH, GM, KE, LS, MW, MZ, SD, SL, SZ, TZ, UG, ZM, ZW), Eurasian patent (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), European patent (AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HU, IE, IT, LU, MC, NL, PT, RO, SE, SI, SK, TR), OAPI patent (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, ML, MR, NE, SN, TD, TG).

Published:

— *with international search report*

— *before the expiration of the time limit for amending the claims and to be republished in the event of receipt of amendments*

(88) **Date of publication of the international search report:**
16 September 2004

For two-letter codes and other abbreviations, refer to the "Guidance Notes on Codes and Abbreviations" appearing at the beginning of each regular issue of the PCT Gazette.

Computer Software Management

TECHNICAL FIELD

This invention relates to computer software management, and more particularly to a method and apparatus for deploying programs and computing platforms to computers associated with different enterprises.

BACKGROUND

A typical enterprise may have a large number of computers distributed over wide geographic areas spanning across different cities or countries. Each of the computers can be configured with a computing platform that may include different and often incompatible hardware components or software components such as different operating systems or application programs. The computing platform may have to be upgraded or replaced with a new computing platform because the current computing platform may have become obsolete.

To deploy a program or a new computing platform onto one or more client computers distributed across an enterprise often requires overcoming a large number of logistical problems. Factors such as the number and type of client computers in the enterprise must be considered. Moreover, it may be difficult to gather and analyze information related to the requirements of all the client computers in an enterprise.

SUMMARY

In one implementation, a first aspect of the invention includes receiving program requests from client computers associated with different enterprises. The validity of the program requests is determined and at least one requested program is sent from a database to at least one client computer.

The aforesaid method may include checking whether the client computers are authorized to access the programs indicated in the program requests. The client computers may be sent a list of programs that the computers are authorized to receive. The availability of the requested programs may be determined by comparing programs that the client computers have received with the programs that the client computers are authorized to receive. The database may be updated to indicate the client computer that has received a program. The database may be located in an enterprise different from the enterprise associated with the client computers.

In a second aspect, the invention provides an apparatus adapted to perform the methods disclosed in the first aspect.

In a third aspect, the invention provides an article comprising a computer-readable medium that stores computer executable instructions for causing a computer to perform the methods disclosed in the first aspect.

In a fourth aspect, the invention provides a method that includes accessing a system configured to perform the methods in disclosed in the first aspect.

In another implementation, a fifth aspect of the invention discloses a method that includes targeting a group of client computers in at least one enterprise to receive at least one program. A notification is sent to the client computers in the group that is to receive at least one program. At least one program is sent to the client computers in the group who have accepted receipt of the program. The program is sent from a database located in one enterprise to the group of client computers located in another enterprise.

The above method may include updating the database to indicate the client computer that has received a program. The notification may include information indicating a time limit within which the program is to be sent to the client computers. The database may be located in an enterprise different from the enterprise associated with the client computers.

In a sixth aspect, the invention provides an apparatus adapted to perform the methods disclosed in the fifth aspect.

In a seventh aspect, the invention provides an article comprising a computer-readable medium that stores computer executable instructions for causing a computer to perform the methods disclosed in the fifth aspect.

In an eighth aspect, the invention provides a method that includes accessing a system configured to perform the methods disclosed in the fifth aspect.

In yet another implementation, a ninth aspect of the invention discloses a method that includes receiving a request for a computing platform from a client computer over a network, wherein the request includes

configuration information of the client computer. In response to the request, the method provides the computing platform. The client computer is sent the computing platform which includes the received configuration information and a hardware independent software image of a computing platform.

The aforesaid method may include modifying the hardware independent software image based on client computer requirements. The hardware independent software image may include a hardware independent portion and a hardware dependent portion. The independent portion may include a software image of a test computer and the hardware dependent portion may include hardware dependent programs. The computing platform may include a program to determine the hardware configuration of the client computers. The hardware independent software image of the new computing platform may be created by installing an operating system and application programs onto a test computer.

In a tenth aspect, the invention provides an apparatus adapted to perform the methods disclosed in the ninth aspect.

In an eleventh aspect, the invention provides an article comprising a computer-readable medium that stores computer executable instructions for causing a computer to perform the methods disclosed in the ninth aspect.

In a twelfth aspect, the invention provides a method that includes accessing a system configured to perform the methods disclosed in the ninth aspect.

In various implementations, the invention may provide one or more of the following advantages. The program deployment process can automate the process of deploying programs to client computers in an enterprise. Such a process may obviate the need for manually distributing programs or for creating distribution applications for each enterprise. The invention also may reduce the burden of software distribution on an enterprise by providing an infrastructure for distributing programs among client computers in different enterprise. These techniques may improve the efficiency of distributing programs because it can leverage a single infrastructure among one or more different enterprises. The process also may exploit the use of the Internet infrastructure by using standard Internet based protocols to perform its task. As a result, the overall cost of updating programs in an enterprise may be reduced.

The program deployment process also can track the usage of programs among the client computers which may important for asset management purposes such a licensing. The process can be implemented as standalone process or combined with a current deployment structure of an enterprise. The process can "pull" application programs to a client computer in response to requests from the computers. Moreover, the process also can "push" application programs to the client and provide several types of time frames in which the client can receive the programs. Consequently, network traffic due to programs distribution from a centralized database to client computers in an enterprise may be reduced.

The invention also may provide one or more of the following advantages. The computing platform process can create a computing platform that may be independent of the specific hardware requirements of client computers in an enterprise. The process can gather in a central location a repository of hardware dependent programs such as device drivers from manufacturers of hardware components of a client computer. As a result, a new computing platform can be created that can reduce the need for information technology (IT) personnel to physically check each of the client computers in an enterprise.

The details of one or more embodiments of the invention are set forth in the accompanying drawings and the description below. Other features, objects, and advantages of the invention will be apparent from the description and drawings, and from the claims.

DESCRIPTION OF DRAWINGS

FIG. 1 is a simplified block diagram of a program management system according to an implementation of the invention.

FIG. 2 is a detailed block diagram of a program management computer according to an implementation of the invention.

FIG. 3 is a flow chart of a program deployment process according to an implementation of the invention.

FIG. 4 is a flow chart of a program deployment process according to another implementation of the invention.

FIG. 5 is a flow chart of a computing platform deployment process according to an implementation of the invention.

Like reference symbols in the various drawings indicate like elements.

DETAILED DESCRIPTION

FIG. 1 is a simplified block diagram of a program management system 10 according to an implementation of the invention. The management system 10 includes a program management computer 12 that communicates with one or more enterprise computer systems 22 over a system network 20. An example of a system network 20 can include a wired or wireless network such as the Internet, the World Wide Web, or other network. The management computer 12 includes a program management module 14 having a program deployment process 16 and a computing platform deployment process 18. The program process 16 is responsible for managing the deployment of programs onto the enterprise systems 22. Similarly, the platform process 18 handles the deployment of a computing platform onto the enterprise systems 22.

Each of the enterprise systems 22 include a server computer 24 that handles one or more client computers 28 over an enterprise network 26. The server computer 24 facilitates communication between the client computers 28 and the management computer 12 over the enterprise network 26 and the system network 20. Each enterprise system 22 belongs to a separate domain protected by a firewall from the other enterprise systems and from the system network 20. Similarly, the management computer 12 is associated with an

enterprise different from the enterprise systems 22. For example, the management computer 12 can be associated with an enterprise having a corporate identity different from that of the client computers 28 and the enterprise systems 22.

In each enterprise system 22, the client computers 28 can issue requests for programs, such as application programs, from the management computer 12. A client computer 28 can be a personal computer (PC) such as a desktop or laptop computer executing an operating system such as Microsoft Windows XP™ and using a Web browser such as Microsoft Internet Explorer™ to connect to the management computer 12 over the system network 20.

The program process 16 can satisfy the requests from the client computers 28 by determining whether the request is valid and by sending the requested programs to the client computers. The program process 16 also is capable of initiating the deployment process by targeting a group of client computers 28 that are to receive programs. Likewise, the client computers 28 can issue requests for a new computing platform, such as a new operating system, from the management computer 12. The platform process 18 handles these requests by sending to the client computer 28 the computing platform along with configuration information associated with the client computer.

FIG. 2 is a detailed block diagram of a program management computer 12 according to an implementation of the invention. The management computer 12 can be implemented as a single server computer that includes a computer bus 40 connecting a central processing unit (CPU) 42 to memory

30. The CPU 42 is capable of executing computer programs 38 residing in memory 30 and processing data stored in memory. The computer programs 38 can include an operating system, such as Microsoft Windows 2000 server, device drivers for handling low-level aspects of hardware components of the computer, application programs such as Microsoft Office, or other programs. The CPU 42 can include an Intel Pentium processor or other processor. The memory 30 can be a combination of read only memory (ROM), random access memory (RAM), static RAM (SRAM), dynamic RAM (DRAM), or other memory.

The memory 30 contains the program management module 14 which includes the program deployment process 16 and the computing platform process 18. A database 31 can be used to store data related to the execution software in the management computer 12. In one implementation, the database 31 is used to store program process data 32 which is data related to the execution of the program process 16. Likewise, the database 31 can be used to store platform process data 34 which is data associated with the execution of the computing platform process 18. The database 31 can be implemented using a database management program such as Microsoft SQL 2000™, Oracle™, Sybase™, or other database management program.

The program process 16 is capable of handling program client requests 16a for programs from client computers 28, and sending client notifications 16b to a targeted group of client computers 28. The program process 16 uses the program process data 32 to perform its various tasks. The program

process data 32 includes programs 32a, program lists 32b, program installed lists 32c, and group lists 32d. The programs 32a represent the programs that the program process 16 can send to the client computers 28 in response to client requests 16a for programs. The program list 32b is list of programs that is sent to the client computers 28 wherein each entry in the list indicates a program that the computer is authorized to receive. The program installed list 32c is a list of programs that the client computer 28 has received. Each client computer 28 is associated with a program list 32b and a program installed list 32c. The group list 32d is data representing a group of client computers 28 that have been targeted to receive one or more programs 32a.

The computing platform deployment process 18 handles client requests 18a for computing platforms 36 from the client computers 28 and uses the platform process data 34 in the database 31 to accomplish this task. The platform process 18 provides a computing platform 36 that includes a hardware independent portion 36a and a hardware dependent portion 36b. The hardware independent portion 36a can include a software image of a test computer that is loaded with an operating system and application programs. The hardware dependent portion 36b includes hardware dependent programs such as device drivers. There can be one or more computing platforms 36 depending on the requirements of the enterprise systems 22. Each client computer 28 is associated with configuration information 35 which represents the hardware and software settings of the client computer.

An input/output (I/O) controller 44 includes hardware elements, such as an interface card, and software elements, such as device drivers, connected to the computer bus 40. The I/O controller 44 is used to couple the management computer 12 to computer components such as the system network 20, a display device 46, an input device 48, or other components.

The display device 46 can include a computer display capable of displaying data related to the execution of programs in the management computer 12. For example, the display device 46 can display data associated with the execution of the computer programs 38 and the program management module 14. The input device 48 can include a pointing device such as a mouse, track ball, light pen, touch sensitive display, or other type of pointing device and may be used in combination with a keyboard to input data into the management module 14 executing on the management computer 12.

The I/O controller 44 connects the management computer 12 to the system network 20. The computer 12 is capable of communicating over a wired or wireless network such as, for example, a local area network (LAN), wide area network (WAN), Ethernet, the Internet, or other network. The operation of the computer 12 can be controlled through a local terminal connected directly to the computer 12 or a remote computer coupled to the computer 12 over the network 20.

FIG. 3 is a flow chart of a program deployment process 16 according to an implementation of the invention that includes having the client computers 12 "pull" programs 32a from the database 32. The client

computers 28, which are associated with different enterprises 22, connect 50 to the management computer 12 through the system network 20. In one embodiment, the connection can include a portal mechanism and a World Wide Web site provided by the management computer 12. The client computer 28 can use a communications program, such as a Web browser, to connect to the portal mechanism of the management computer 12 over the system network 20. The connection process can include a login process which can verify whether the client computer 28 is authorized to access the management computer 12. For example, information related to the client computer 28 attempting to access the management computer 12 can be compared to authorization information stored in the database 31 to determine the identity of the client computer and whether it is authorized to access the management computer 12.

Once a client computer 28 is authorized to access the management computer 12, the program process 16 sends 52 the client computer a program list 32b containing a list of the programs 32a in the database 32 that the computer is authorized to receive. In one embodiment, the program process 16 can send the program list 32b in the form of a Web page containing the list of the authorized programs. The Web page can be constructed using standard Web language and protocols such as hypertext markup language (HTML). The contents of the program list 32b can be based on the programs that the client computer 28 is authorized to access. For example, a client computer in the marketing department of an enterprise may be authorized to access

programs that may be different from the programs a client computer in the manufacturing department is authorized to access.

The program process 16 receives 54 client requests 16a for programs from the client computer 28. The program requests 16a are generated by the client computer 28 when it selects one or more programs it would like to receive from the program list 32b. In one embodiment, the client computer 28 can make this selection from a Web page that contains one or more entries representing the programs 32a from the program list 32b.

Once the program process 16 receives a client request 16a for programs, the process determines 56 the validity of the program requests. This determination may include checking the availability of the programs indicated in the program request 16a. This checking procedure can include comparing programs that the client computer 28 has received in the past, as indicated by the data in the program installed 32c, with the programs that the client computer 28 is authorized to receive as indicated by the data in the program list 32b.

If the client computer 28 is authorized to receive one or more programs, the program process 16 sends 58 the programs 32a from the database 32 to the client computer. The programs 32a that are sent to the client computer can be packaged with an program module for installing the programs on the client computer 28. An example of an installation module can include a Wice interface program.

Once the programs requested by the client computer 28 have been received and installed on the client computer, the program process 16 can update a data structure in the database to indicate the client computer that has received a program. This data structure also can be a part of an asset management database used to track assets (programs) for licensing purposes. In addition, data in the programs installed 32c can be updated to indicate which programs the client computer has requested and received. As mentioned above, this data can be used to determine which programs the client computer has received.

Moreover, the client computer 28 can be sent a notification indicating the status of the program request. For example, the client computer 18 can be sent an alert in the form of an electronic mail (Email) message indicating status information including whether the request was a success or failure.

FIG. 4 is a flow chart of a program deployment process 16 according to another implementation of the invention in which programs 32a are "pushed" from the database 32 to the client computers 28. One or more client computers 28 connect 60 to the management computer 12 through the system network 20 using techniques similar to those described above. The program process 16 keeps track of the client computers that are connected to the management computer 12 using standard techniques. For example, a login process similar to the process discussed above can be used to verify whether the client computer is authorized to access the management computer 12. In addition, the program process 16 can use the login process to gather

information related to the client computer and to store this information in a data structure in the database 31. The data structure can be organized to include entries for the client computers 28 in each of the enterprises 22. As a result, the identities of the client computers that are connected to the management computer 12 are obtained.

The program process 16 targets 62 a group of client computers 28 in at least one enterprise 22 to receive at least one program 32a. The group list 32d is a data structure that can be used to store the data identifying the client computers in a group. There can be one or more group list 32d for each enterprise 22. The members in each group list 32d can be selected based on criteria such as the department associated with the client computers, the software or hardware requirements of the client computers, business consideration of the enterprise, or other criteria. For example, the marketing department in an enterprise may decide that the employees in the marketing department should receive either a new program or an upgrade to a current program. As a result, the marketing department can use the deployment process 16 to create a group list 32d that includes the client computers associated with the employees in the department and the programs 32a they are to receive.

Once a group list 32d has been created for the targeted group, the program process 16 sends 64 a notification 16b to the client computers in the group that is to receive at least one program. The notification 16b can include a message, such as an Email or a Web page, which includes time limit

information such as a time period indicating when a program is to be sent to the client computers. The notification 16b can indicate that the receipt of programs is mandatory. In this case, the notification 16b provides the client computer with one or more time periods to select from and a time limit in which the program is to be sent to the client computer. Alternatively, the notification 16b can indicate an optional time limit in which case the client computer is given an option to receive the program immediately or at a later time.

The program process 16 sends 66 at least one program 32a to the client computers 28 in the group who have accepted receipt of the program. If program process 16 sends the programs 32a from the database 32 to the client computers 28 that have been authorized to receive the programs. The database 32 is located in an enterprise different from the enterprise associated with the group. As mentioned above, the programs 32a that are sent to the client computer can be packaged with an installation module that allows the programs 32a to be easily installed on the client computer 28.

The client computer 28 can be sent a notification indicating the status of the program request. For example, the client computer 28 can be sent an alert in the form of an electronic mail (Email) message indicating whether the request was a success or failure. Such a notification prevents the occurrence of multiple installations.

Once the programs 32a have been received and installed on the client computer 28, the program process 16 updates 68 a data structure in the

database to indicate the client computers in the group that have received a program. As mentioned above, this data structure also can be a part of an asset management database which can be used to track assets (programs) for licensing purposes.

FIG. 5 is a flow chart of a computing platform process 18 according to an implementation of the invention. The client computer 28 establishes 70 a connection to the management computer in a manner similar to the manner described above. For the following description it is assumed that the client computer 28 has established a connection to the management computer and is authorized to send requests 18a to receive a new computing platform 36.

The platform process 18 receives 72 a client request 18a for a computing platform 36 from a client computer 28 over the system network 20.

The request 18a can be accompanied with configuration information 35 associated with the client computer 28. The configuration information 35 can include hardware settings of the client computer 28 such as printer port assignments or other settings. The configuration information 35 also can include software parameters such as language type that the client computer has selected or other parameters. In one embodiment, the configuration information 35 can be captured by a program embedded in a Web page sent by the process 18 to the client computer 28.

The platform process 18 provides 74 the computing platform 36. In one implementation, a hardware independent software image of the computing platform 36 can be created by installing an operating system, one

or more application programs, and device drivers onto a test computer such as a PC. As mentioned above, the computing platform 36 is a hardware independent software image of a computing platform that includes a hardware independent portion 36a and a hardware dependent portion 36b.

The independent portion 36a includes a software image of a test computer. An image creating program, such as Sysprep from Microsoft, can be used to create the hardware independent portion 36a. This program can detect the hardware aspects of the test computer and abstract these aspects by replacing specific identifiers associated with the test computer with a generic identifier. The software image can be captured and stored in storage resource as a compact disk (CD) or in the database 32. The platform process 18 can handle specific requirements of one or more enterprise systems 22 and of client computers 28. For example, a client file can be created for each enterprise or for a group of client computers in an enterprise. The client file can include information related to the enterprise such as enterprise wide settings including language settings, the graphic display resolution, printer mappings, or other settings.

The hardware dependent portion 36b includes hardware dependent programs such as device drivers for the hardware components of the test computer such as video cards, printers, network cards, or other components. The hardware dependent 36b portion can be continuously updated with device drivers provided by manufacturers of hardware components. In this way, the

computing platform 36 is hardware independent and, therefore can be installed on a large number of different client computers 28.

The platform process 18 sends 76 to the client computer 28 the configuration information 35 associated with the client computer and the computing platform 36. The computing platform 36 can be packaged with software that can interface with a download manager interface of an operating system such as Microsoft Windows XP™. When the computing platform 36 is received by the client computer, the download manager executes and examines the hardware dependent portion 36b of the computing platform 36. The download manager evaluates the hardware aspects of the client computer 28 and determines which device drivers from the hardware dependent portion 36b are necessary for proper operation. The appropriate device drivers are then installed in the client computer 28.

Moreover, the computing process 18 is capable of handling post installation procedures that may not be handled by the creation of the client file. For example, the procedure can be handle custom settings such as the network drive mappings on a client computer 28. This procedure can be automated by executing a program that gathers and organizes the custom setting information from the client computers 28 in an enterprise 22. The program can then use this information to configure the settings of the client computers that require such configuration.

A number of embodiments of the invention have been described. Nevertheless, it will be understood that various modifications may be made

without departing from the spirit and scope of the invention. For example, the techniques described above can be easily adapted to operate with other operating systems such as the Linux operating system. Accordingly, other embodiments are within the scope of the following claims.

5

CLAIMS:

1. A method comprising:

for each of a plurality of client computers, storing a unique identifier and at least one characteristic in a first database;

storing a plurality of programs in a second database;

a management server receiving a program request said program request including a desired value for a specified characteristic and a desired program, wherein said specified characteristic is stored in the first database and said desired program is stored in the second database;

the management server querying the first database to identify one or more of said plurality of client computers having a stored characteristic value that matches said desired characteristic value, and storing the unique identifiers of the matched client computers in a group list;

the management server querying the second database to determine availability of and authority for said desired program to said group list;

the management server assigning a time frame, and, within that time frame, sending the desired program from the second database to the client computers whose unique identifiers are in the group list;

the management server recording instances of programs being sent to client computers, wherein said instance record includes the name of the program, the program version, the client computer, and the date and time of the sending; and

tracking recorded instances of programs sent to client computers so as to verify that sending such programs is in compliance with license obligations.

2. The method of claim 1, further comprising updating the second database to indicate the client computer that has received a program.

3. A computer apparatus comprising: a memory, and a processor coupled to the memory, wherein the processor is configured to:

access a first database containing for each of a plurality of client computers a unique identifier and at least one characteristic;

access a second database containing a plurality of programs;

receive a program request, said program request including a desired value for a specified characteristic and a desired program, wherein said specified characteristic is stored in the first database and said desired program is stored in the second database;

query the first database to identify one or more of said plurality of client computers having a stored characteristic value that matches said desired characteristic value, and store the unique identifiers of the matched client computers in a group list;

assign a time frame, and, within that time frame send the desired program from the second database to the client computers in the group whose unique identifiers are in the group list;

record instances of programs being sent to client computers, wherein said instance record comprises the name of the program, the program version, the client computer, and the date and time of the sending; and

track recorded instances of programs sent to client computers so as to verify that sending such programs is in compliance with license obligations.

4. The apparatus of claim 3, wherein the processor is further configured to send a notification to the client computers whose unique identifiers are in the group list, where said notification includes the assigned time frame.

5. The apparatus of claim 3, wherein the second database is located in an enterprise different from the enterprise associated with the client computers.

6. A computer article comprising a computer-readable medium that stores executable instructions for causing a computer system to:

access a first database containing for each of a plurality of client computers a unique identifier and at least one characteristic;

access a second database containing a plurality of programs;

receive a program request, said program request including a desired value for a specified characteristic and a desired program, wherein said specified

characteristic is stored in the first database and said desired program is stored in the second database;

query said first database to identify one or more of said plurality of client computers having a stored characteristic value that matches said desired characteristic value, and store the unique identifiers of the matched client computers in a group list;

assign a time frame, and, within that time frame send the desired programs from the second database to the client computers whose unique identifiers are in the group list;

record instances of programs being sent to client computers, wherein said instance record comprises the name of the program, the program version, the client computer, and the date and time of the sending; and

track recorded instances of programs sent to client computers so as to verify that sending such programs is in compliance with license obligations.

7. The article of claim 6, comprising instructions for further causing the computer to update the second database to indicate the client computer that has received a program.

8. A method comprising:

for each of a plurality of client computers, storing a unique identifier, client computer hardware configuration data, and at least one characteristic in a database;

receiving a computing platform request, said computing platform request including a desired value for a specified characteristic and a desired computing platform, wherein said specified characteristic is stored in the database;

the management server querying the database to identify one or more of said plurality of client computers having a stored characteristic that matches said desired characteristic value, and storing the unique identifiers of said matched client computers in a group list;

recording instances of computing platforms being sent to client computers, wherein said instance record comprises the name of the computing platform, the computing platform version, the client computer, and the date and time of the sending;

tracking recorded instances of computing platforms being sent to client computers so as to verify that sending such computing platforms is in compliance with license obligations; and

sending to the client computers whose unique identifiers are in the group list the desired computing platform by:

determining a computing platform image based on said client computer hardware configuration data, wherein said platform image comprises a hardware independent portion and a hardware dependent portion;

sending the hardware independent portion; and

sending the hardware dependent portion.

9. The method of claim 8, further comprising modifying the hardware independent portion based on client computer requirements.

10. A computer apparatus comprising: a memory, and a processor coupled to the memory, wherein the processor is configured to:

access a database containing for each of a plurality of client computers a unique identifier, client computer hardware configuration data, and at least one characteristic;

receive a computing platform request, the request including a desired value for a specified characteristic and a desired computing platform, wherein said specified characteristic is stored in the database;

query the database to identify one or more of said plurality of client computers having a stored characteristic value that matches said desired characteristic value, and store the unique identifiers of the matched client computers in a group list;

record instances of computing platform being sent to client computers, wherein said instance record comprises the name of the computing platform, the computing platform version, the client computer, and the date and time of the sending;

track recorded instances of computing platforms being sent to client computers so as to verify that sending such programs is in compliance with license obligations;

send to the client computers whose unique identifiers are in the group list the desired computing platform, where the processor is further configured to:

determine a computing platform image based on said client computer hardware configuration data, wherein said platform image comprises a hardware independent portion and a hardware dependent portion;

send the hardware independent portion; and

send the hardware dependent portion.

11. The apparatus of claim 10, wherein the computing platform includes a program to determine the hardware configuration of the client computers.

12. The apparatus of claim 10, wherein the hardware independent portion of the computing platform image is created by installing an operating system and application programs onto a test computer.

13. A computer article comprising a computer-readable medium that stores executable instructions for causing a computer system to:

access a database containing for each of a plurality of client computers a unique identifier, client computer hardware configuration data, and at least one characteristic;

receive a computing platform request, the request including a desired value for a specified characteristic and a desired computing platform, wherein said specified characteristic is stored in the database;

record instances of computing platforms being sent to client computers, wherein said instance record comprises the name of the computing platform, the computing platform version, the client computer, and the date and time of the sending;

track recorded instances of computing platforms being sent to client computers so as to verify that sending such computing platforms is in compliance with license obligations; and

send to the client computers whose unique identifiers are in the group list the computing platform, where the article comprises further instructions to:

determine a computing platform image based on said client computer hardware configuration data, wherein said platform image comprises a hardware independent portion and a hardware dependent portion;

send the hardware independent portion; and

send the hardware dependent portion.

14. The article of claim 13, comprising instructions for further causing the computer to modify the hardware independent software image based on client computer requirements.

15. The method of claim 1, wherein said management computer comprises a program management module having a program deployment process and a computing platform deployment process.

16. The method of claim 1, wherein said plurality of client computers are located in a plurality of enterprises.

17. The method of claim 1, wherein said first database is located in an enterprise and said second database is located in another enterprise.

18. The method of claim 1, further comprising notifying client computers of said assigned time frame.

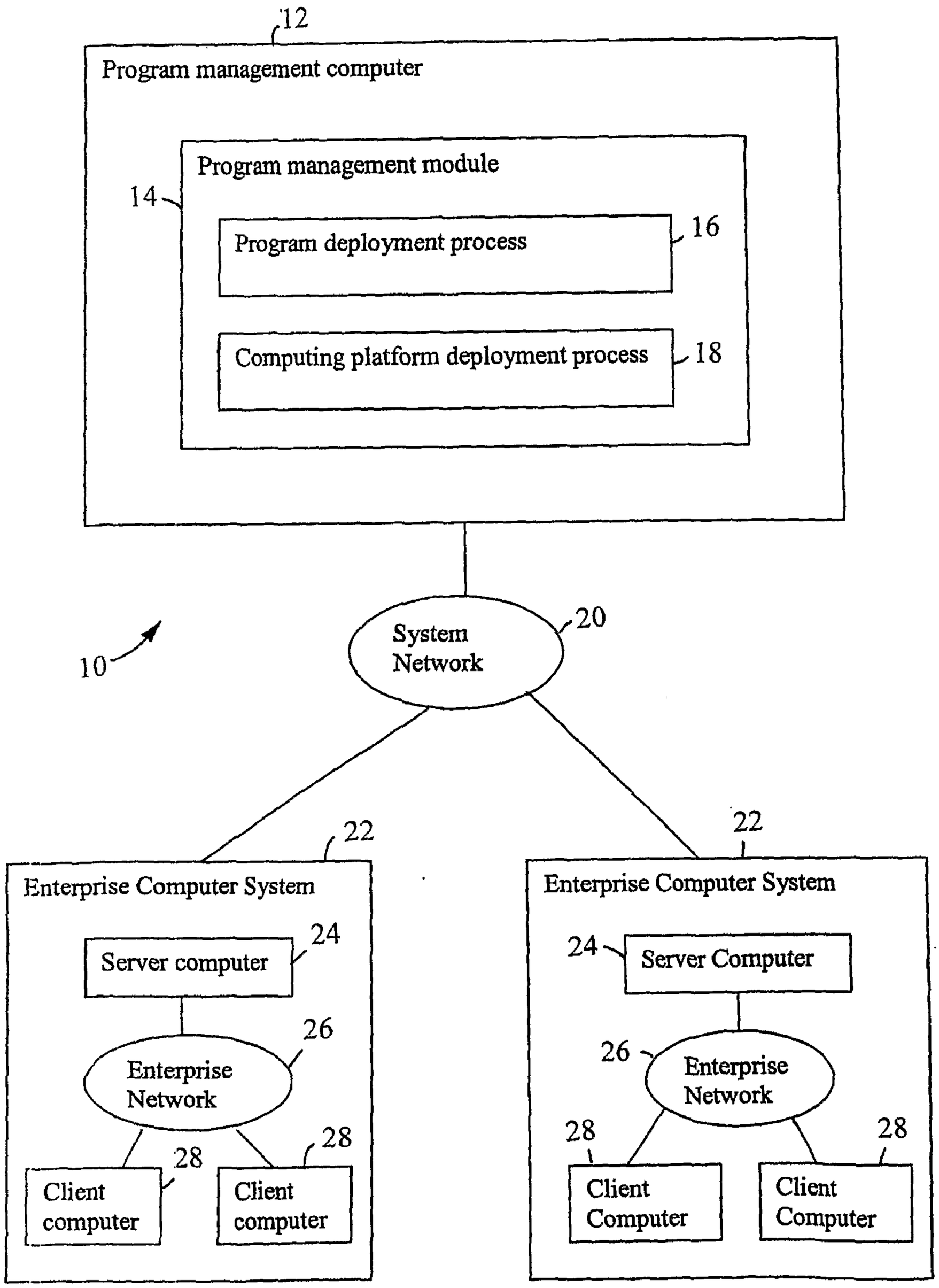

12
Program management computer
Program management module
14
Program deployment process
16
Computing platform deployment process
18
10
System Network
20
22
Enterprise Computer System
Server computer
24
26
Enterprise Network
28
Client computer
Client computer
22
Enterprise Computer System
24
Server Computer
26
Enterprise Network
28
Client Computer
Client Computer

FIG. 1

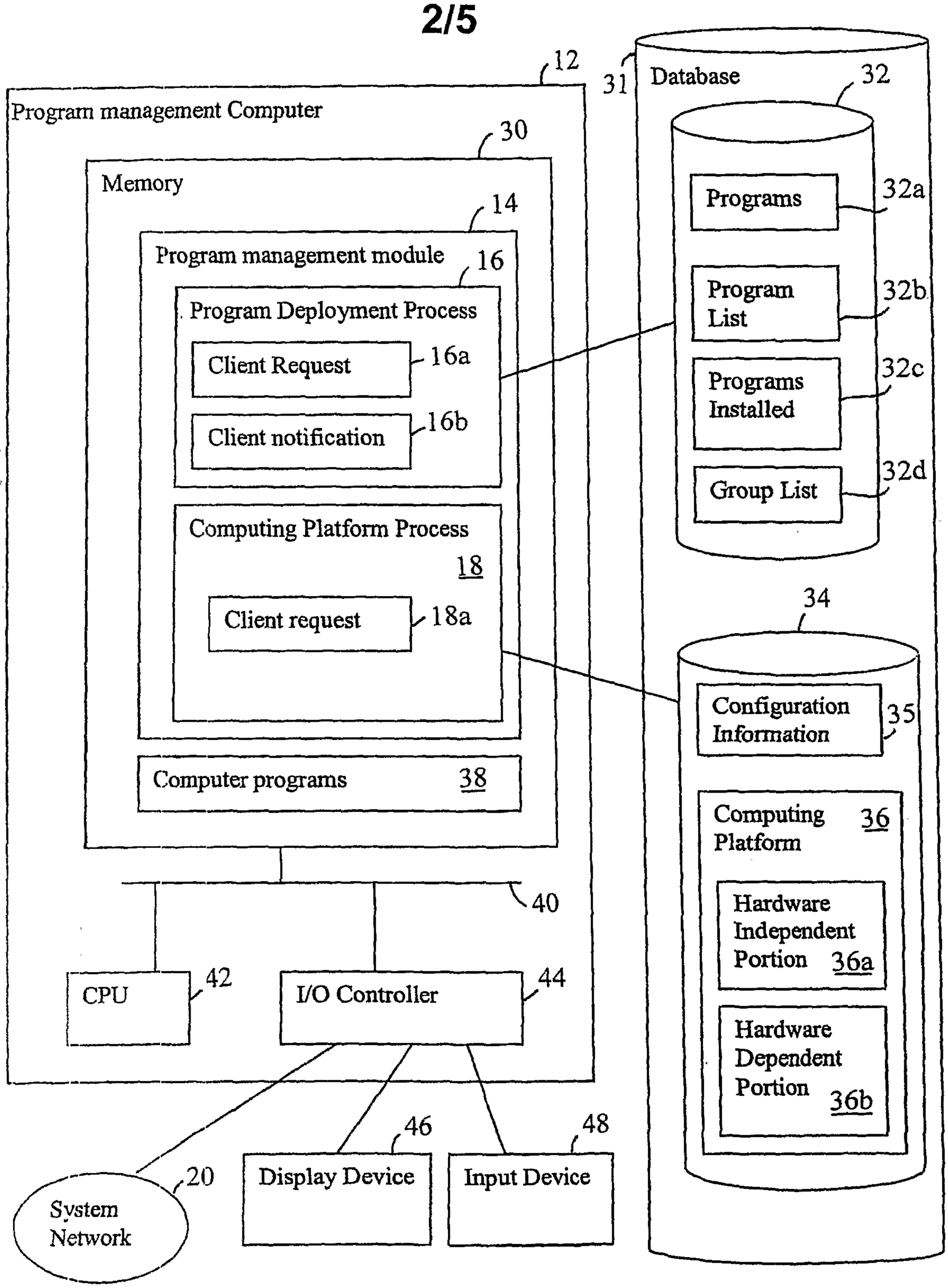

12
Program management Computer
30
Memory
14
Program management module
16
Program Deployment Process
Client Request
16a
Client notification
16b
Computing Platform Process
18
Client request
18a
Computer programs
38
40
CPU
42
I/O Controller
44
20
System Network
46
Display Device
48
Input Device
31
Database
32
Programs
32a
Program List
32b
Programs Installed
32c
Group List
32d
34
Configuration Information
35
Computing Platform
36
Hardware Independent Portion
36a
Hardware Dependent Portion
36b

FIG. 2

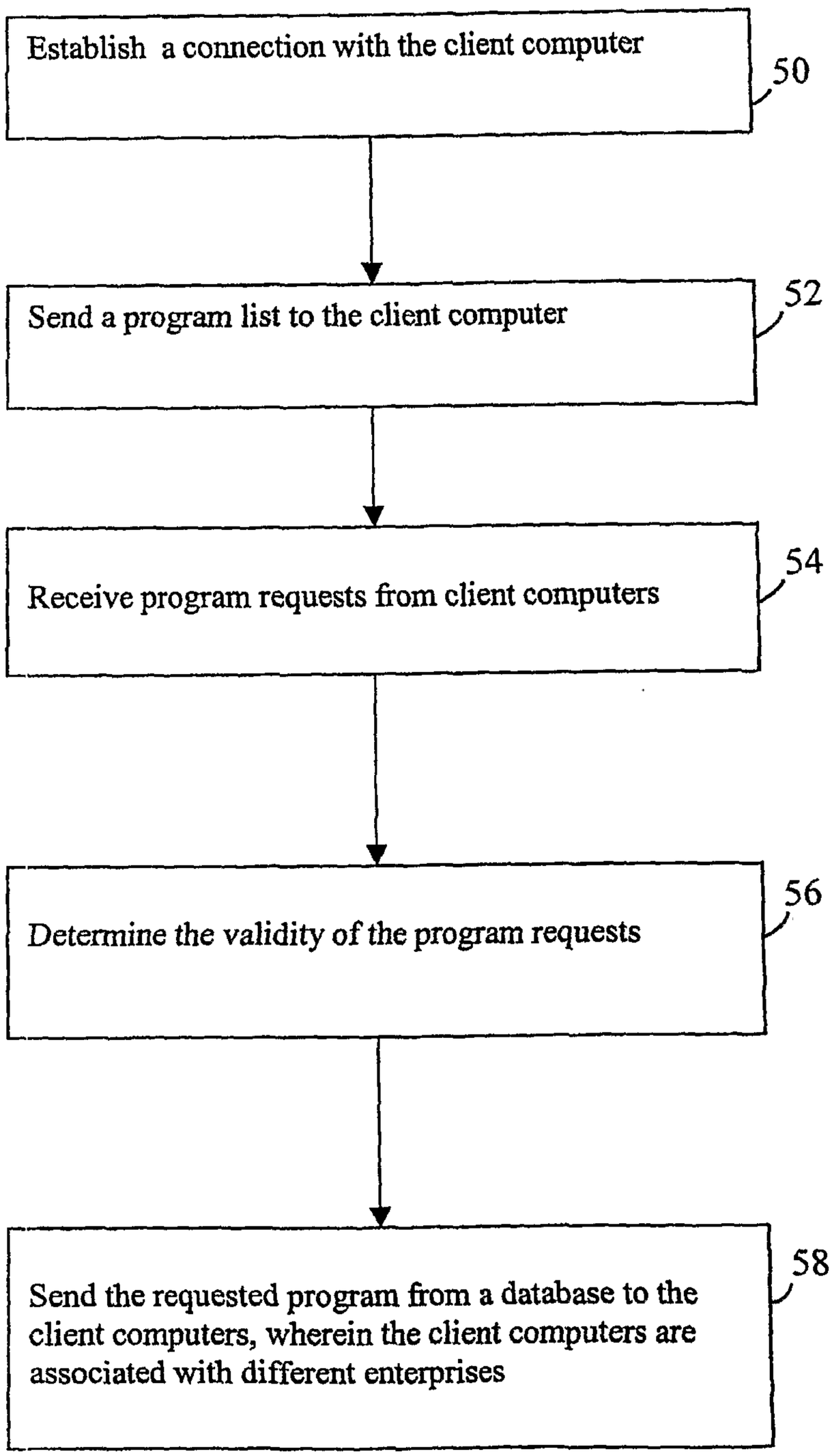

Establish a connection with the client computer
50
Send a program list to the client computer
52
Receive program requests from client computers
54
Determine the validity of the program requests
56
Send the requested program from a database to the client computers, wherein the client computers are associated with different enterprises
58

FIG. 3

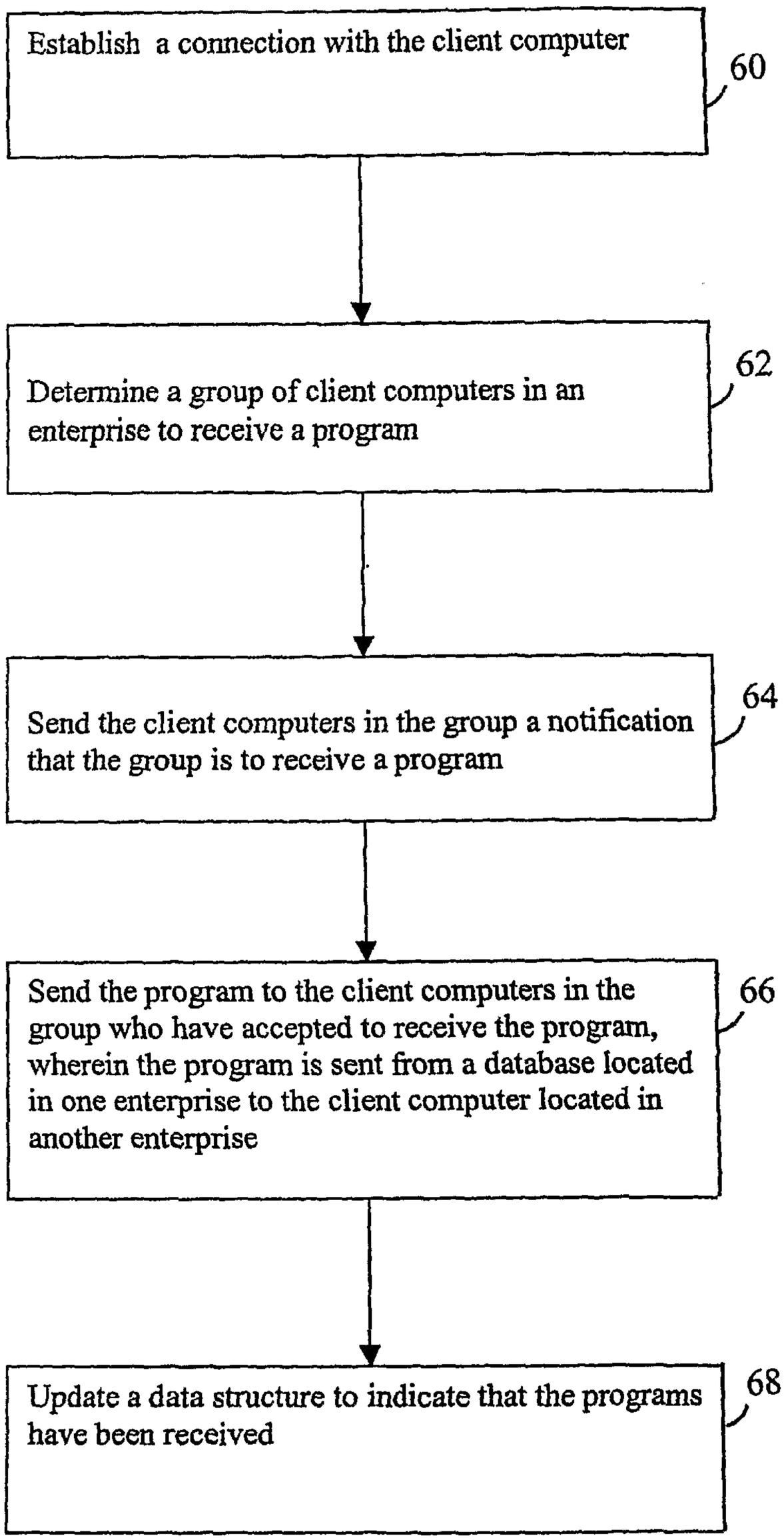

Establish a connection with the client computer
60
Determine a group of client computers in an enterprise to receive a program
62
Send the client computers in the group a notification that the group is to receive a program
64
Send the program to the client computers in the group who have accepted to receive the program, wherein the program is sent from a database located in one enterprise to the client computer located in another enterprise
66
Update a data structure to indicate that the programs have been received
68

FIG. 4

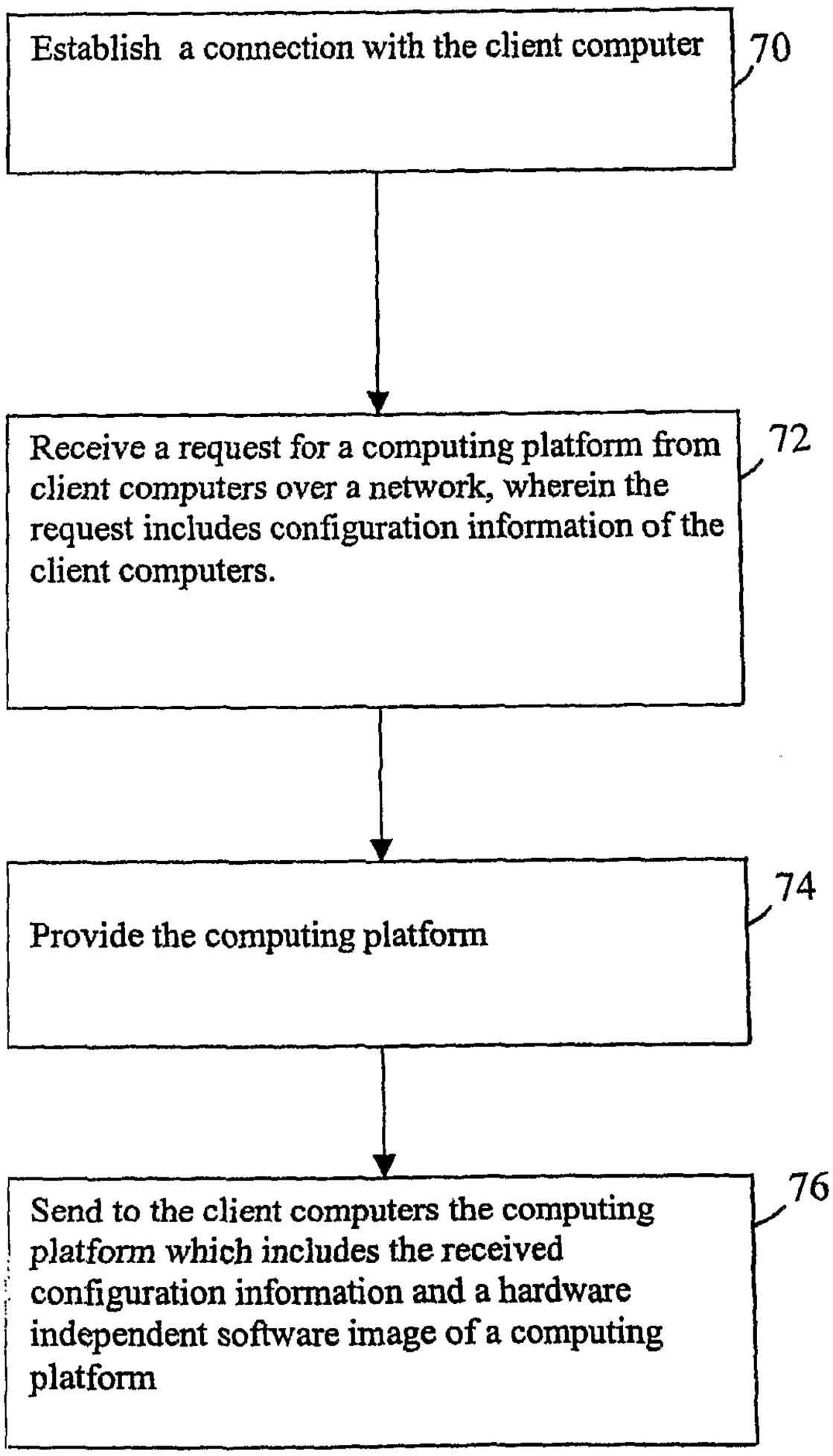
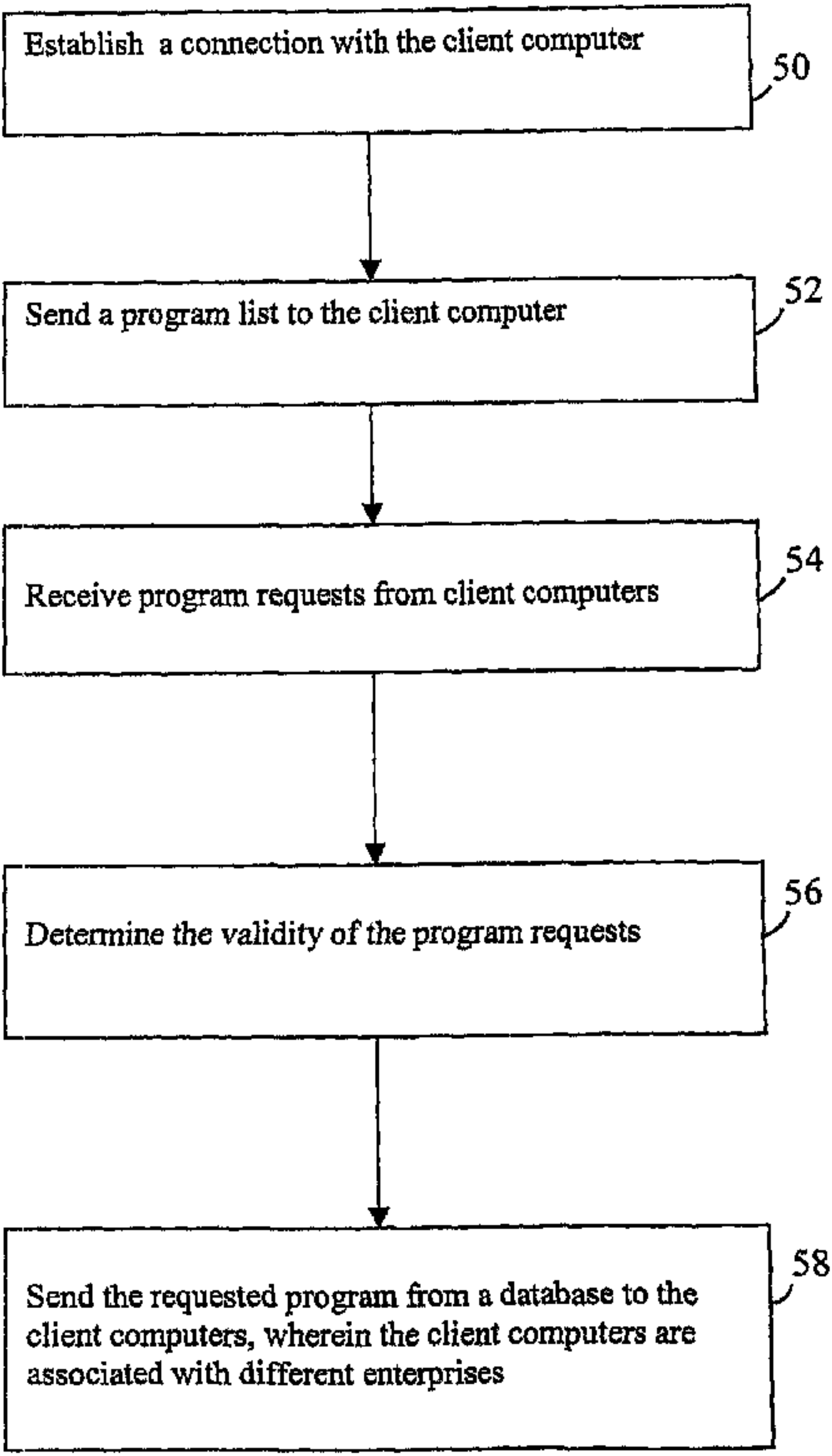

Establish a connection with the client computer
70
Receive a request for a computing platform from client computers over a network, wherein the request includes configuration information of the client computers.
72
Provide the computing platform
74
Send to the client computers the computing platform which includes the received configuration information and a hardware independent software image of a computing platform
76

FIG. 5


Establish a connection with the client computer
50
Send a program list to the client computer
52
Receive program requests from client computers
54
Determine the validity of the program requests
56
Send the requested program from a database to the client computers, wherein the client computers are associated with different enterprises
58