



(11) **EP 1 331 556 B9**

(12) **CORRECTED EUROPEAN PATENT SPECIFICATION**

(15) Correction information:

Corrected version no 1 (W1 B1)

Corrections, see

Drawings

Drawing(s) replaced or added

(51) Int Cl.:

G06F 9/44 (2006.01)

(48) Corrigendum issued on:

28.10.2009 Bulletin 2009/44

(45) Date of publication and mention
of the grant of the patent:

12.08.2009 Bulletin 2009/33

(21) Application number: **01202210.9**

(22) Date of filing: **08.06.2001**

(54) **Server-based computer environment**

Serverbasierte Rechnerumgebung

Environnement d'ordinateur basé sur un serveur

(84) Designated Contracting States:

**AT BE CH CY DE DK ES FI FR GB GR IE IT LI LU
MC NL PT SE TR**

• **Jansen, Peter Gerardus**
5627 JX Eindhoven (NL)

(43) Date of publication of application:

30.07.2003 Bulletin 2003/31

(74) Representative: **de Vries, Johannes Hendrik**

Fokke

De Vries & Metman

Overschiestraat 180

1062 XK Amsterdam (NL)

(73) Proprietor: **Real Enterprise Solutions
Development B.V.**

5215 MV 's-Hertogenbosch (NL)

(56) References cited:

WO-A-97/28623 US-A- 5 548 779

US-A- 5 793 368 US-A- 5 909 545

(72) Inventors:

• **Janssen, Bob**

4926 RH Lage Zwaluwe (NL)

Note: Within nine months of the publication of the mention of the grant of the European patent in the European Patent Bulletin, any person may give notice to the European Patent Office of opposition to that patent, in accordance with the Implementing Regulations. Notice of opposition shall not be deemed to have been filed until the opposition fee has been paid. (Art. 99(1) European Patent Convention).

EP 1 331 556 B9

Description

[0001] The invention relates to a server-based computing system, comprising at least one server and at least one client computer, connected to the server through a network, wherein the server comprises means for providing the client computer with a user interface, wherein the client computer comprises an input device for providing input to an application and a display device for presenting output from an application through the user interface, wherein the server comprises means for running the application, wherein the client computer comprises means for locally running at least one further application.

[0002] The invention further relates to a method for providing a client computer with a user interface for controlling at least one application that can be run locally on the client computer, which client computer is connected to a server through a network and comprises a display device, an input device and means for running the application.

[0003] Known systems of the above-mentioned kind are capable of providing a user interface to the client computer, enabling the client computer to control an application running on the server. Such a set-up is useful in networks containing several client computers. Applications are all installed on the server and can thus be centrally managed, updated, etc. Because the client computers only process the input from the user and present output from the application to the user, they can be very simple. They can, for example, be equipped with small hard disks or none at all and little processing power. Consequently, such client computers are known as thin clients.

[0004] However, it is not always wise or possible to rely only on server-based computing. It is sometimes desirable to run applications locally on the client. In existing systems, such applications have to be initialised separately. The user shuts down the user interface provided by the server and separately starts up the second, locally running, application.

[0005] In the known system and known method of the kind mentioned above, the client computer generates a user interface for controlling the locally running application. This gives scope for confusion, since a user has to clearly distinguish between user interfaces, remember separate functions, get used to different visual aids, etc. Without careful management and updating of all the client computers, a situation can easily arise wherein each client computer attached to the network has its own user interface.

[0006] WO91/02305 discloses a system containing workstations and a main frame. The workstations can be run in an application selection mode simultaneously displaying representations of both local and remote applications from which either remote or local applications may be started using a common selection procedure. When a remote application is selected a surrogate application is run on the workstation which establishes com-

munication with the remote processor.

[0007] It is an object of the present invention to provide a system and method of the kind mentioned above wherein the user interface is easy to use and manage.

[0008] Accordingly the system according to the invention is characterised in that the system comprises means for controlling the locally run applications through the user interface provided by the server.

[0009] The method according to the invention is defined in claim 11.

[0010] Thus, a system manager can centrally manage the user interface. Updates, for example, need only be installed once. It is not necessary to use cumbersome methods for remotely installing updates on each client computer or install updates from a CD on each client computer. Users can work on a different client computer, without having to adjust the settings according to their preferences or wait until their settings have been downloaded onto the client computer.

[0011] In a preferred embodiment, the system according to the invention comprises means for controlling an application running on the server and further applications, running locally, through the user interface.

[0012] Thus, it is possible to integrate the local and central computing environments, so that a user will not notice any difference between locally and centrally running applications. In the system according to the invention, it is not necessary to switch between separate user interfaces, nor is it necessary to overlay one user interface over another with consequent deterioration of the quality of display and control.

[0013] In another preferred embodiment, the user interface comprises means for initiating a locally run application. This feature further contributes to the single integrated computer environment. The user interface allows initialisation of every application in the same manner, irrespective of whether it is run locally on the client computer or centrally on the server.

[0014] The invention will now be explained in further detail with reference to the enclosed drawings, of which

[0015] Fig. 1 shows an embodiment of the system according to the invention.

[0016] Fig. 2A-D show the composition of the display of the user interface in different situations during use of an embodiment of the system according to the invention.

[0017] Fig. 3 shows schematically some steps in an embodiment of the method according to the invention, which describe in more detail the process by which a locally run application is launched.

[0018] Fig. 1 shows schematically an example of a system according to the invention. The system comprises a server 1, attached to a network 2. The server 1 comprises one or more processors, RAM, and storage means, e.g. an array of hard disks. The processing capacity and memory are sufficient for running several demanding applications concurrently on the server 1. The invention is not limited to server-based computing systems comprising only one server. There can be several central servers,

collectively known as a server farm. In this case, processing and storage capacity could be unevenly distributed over the servers, with some servers dedicated to running demanding applications and others to storing or transmitting data.

[0019] The network 2 can be a Local Area Network, a Wide Area Network or the Internet. It can contain wireless connections, dial-up connections or fixed cable connections.

[0020] Fig. 1 shows three client computers 3-5, attached to the network 2. Of course, the invention is not limited to the particular configuration of Fig. 1. Any number of client computers could be attached to the network 2. The client computers 3-5 can be of any kind.

[0021] A first and second client computer 3, 4 respectively, comprise a screen, a keyboard and/or a pointer device, such as a mouse. The server 1 provides these client computers 3,4 with a user interface to applications running on the server 1. Since the first and second client computers 3, 4 are not intended for operation as stand-alone devices, but merely to provide input to and receive output from applications running on the server 1, their processing capacity can be limited. Likewise, they can have limited or no capacity for permanently storing data. The first and second client computers 3,4 are merely provided with such processing capacity and memory as are necessary to process the input and display output, and exchange these with the server 1 over the network 2. For this reason, they are also known as thin clients or network computers.

[0022] The system of Fig. 1 also comprises a third client computer 5, which does comprise means 6 for locally running an application, i.e. the necessary processing power, memory and data storage capacity. When an application is run locally, the processor in the third client computer 5 executes the application code. Because of these extended capabilities, computers such as the third client computer 5 are also known as fat clients.

[0023] There are several reasons as to why it should be desirable to include such fat clients in a server-based computing system. It is possible that an application is not suitable for running centrally. If migration is taking place from a system wherein all applications are run locally, to a central server-based computing environment, some of the old, so-called legacy software might still be needed. Other applications cannot be centrally run in principle, e.g. streaming media applications. Certain applications require specialised graphics capabilities that are best provided locally, or they rely on a locally attached device, for instance a scanner or dongle. In other cases, it might simply be cheaper to have several single-user licenses rather than a multi-user license. It might also be necessary to include in the network 2 workstations with the capability of running an additional or different operating system.

[0024] In addition, it might be necessary or useful to have a fallback mechanism for mission-critical applications that are installed on the server 1. If the server 1

should fail, access to a mission-critical application will be impossible until the problem with the failed server 1 has been solved. If, in such a situation, it is possible to run the application on the third client computer 5, the impact on the user organisation is limited to not being able to use the other, non-mission critical applications installed on the server 1.

[0025] The third client computer 5 shown in Fig. 1 comprises a monitor with a screen 7 for displaying a user interface to an application. It further comprises an input device 8. In the shown example, this is a keyboard, but other input devices can also be connected. The client computer 5 could additionally comprise a mouse, pointer, tablet, web-cam, microphone, etc. The input device 8 forms the means for providing input to an application through a user interface.

[0026] In the system according to the invention, a single user interface is provided to the client computers 3-5. It makes no difference to a user whether he is controlling the running of applications from the first client computer 3, a thin client, or the third client computer 5, a fat client. The only difference is that on the third client computer 5, additional, locally running applications can be initiated and controlled. However, this is done through the one user interface provided by the server 1 for providing input to and receiving output from centrally running applications.

[0027] In the system according to the invention, the server 1 runs an interface management program in order to provide the client computers 3-5 with the user interface. The client computer 5 runs a subscriber program that facilitates the integration of client-based applications in the server-based user interface. The subscriber program running on the client computer 5 co-operates with the interface management program in the exchange between the server 1 and client computer 5 of user input to the interface and output from the interface. Communication between the server 1 and the client computer 5 consists of messages generated by the interface management program and the subscriber program respectively.

[0028] Figs. 2A-D show schematically the composition of the display of the user interface. The figures refer to several situations that occur in embodiments of the system according to the invention.

[0029] Fig. 2A shows a local client screen area 9. This is a screen area generated locally on the client computer 5. Fig. 2A further shows a central application screen area 10. This screen area is generated on the server 1. It comprises a display of output from any applications running centrally on the server 1 and components that are generated by the interface management program regardless of whether any applications are running on the server 1.

[0030] These last-mentioned components comprise a central task bar 11, allowing the user to control the display of output. A button 12 on the task bar 11 forms a means for selecting and changing features of the interface. Further buttons can be present, depending on how many applications are running at any one moment.

[0031] The central application screen area 10 further comprises icons 13-15, which can be selected to launch an application. A first icon 13 launches a server-based application. A second and third icon 14 and 15 respectively, form the means for initiating locally running applications.

[0032] In a preferred embodiment a database in the system comprises information on the available applications that can be run locally on the client computer 5, so that the icons 14, 15 for each of these applications can be generated by the server 1. This database can be stored at one location in the system, for example in the server 1, or parts can be stored locally, for instance in a registry file, and referred to when necessary. Other embodiments are conceivable wherein a number of messages are exchanged between client computer 5 and server 1, for instance when the client computer (5) goes on-line, in order that the latter can establish which icons to generate.

[0033] The local client screen area 9 and the central application screen area 10 are merged into one local client screen 16, which is the screen image shown on the screen 7 of the client computer 5. In the preferred embodiment this merging is taken care of by the subscriber program on the client computer 5, after data relating to the contents of the central application screen area 10 has been transferred to the client computer 5. An alternative embodiment is conceivable, wherein the merging is done on the server 1, after the contents of the local client screen area 9 have been transferred to the server 1. The merged local client screen 16 would then be transferred back from the server 1 to the client computer 5.

[0034] The subscriber and interface management programs also allow the server 1 to control the display of the user interface on the screen 7 of the third client computer 5. The server 1 controls the display properties of the interface, which comprise, for instance, the desktop size and display.resolution, and adapts them to the screen 7 of the client computer 5. The interface management program generates this information. The subscriber program on the client computer 5 ensures display on the screen 7 according to the specified properties.

[0035] Any changes in either the local client screen area 9 or the central application screen area 10 automatically result in an updated version of the merged local client screen 16. One such change can occur, when an application on the server 1 has been launched by clicking the first icon 13. Fig. 2B shows how this leads to a changed local client screen 16. The application running on the server 1 generates a window 17 in which output from the application is displayed. Of course, within the scope of the invention, some applications can generate more than one window. The creation of the window 17 leads to the creation of a button 18 in the central task bar 11. As described before, the local client screen area 9, which in this case is still empty, and the central application screen area 10 are merged into the merged local client screen 16. A user of the client computer 5 can switch to

the application by clicking the button 18.

[0036] In Fig. 2C the situation is shown, wherein the user has launched an application running locally on the client computer 5, by clicking one of the two icons 14, 15. The exact procedure followed in the system as a result of this launch command is described in further detail below with reference to Fig. 3. The result of the launch of a second application, running locally on the client computer 5, is a window 19 in the local client screen area 9, containing the output of the locally running application. As before, the local client screen area 9 is merged with the central application screen area 10, which in Fig. 2C is unchanged with respect to the Fig. 2B, into the merged local client screen 16. Changes in the contents of the window 19 automatically result in an updated version of the merged local client screen 16.

[0037] As a result of the launch of the locally running second application, there are now two windows 17, 19 in the merged local client screen 16. A user can make one of the windows 17, 19 active, by clicking on that window. As a result of this clicking action, a signal generated by the subscriber program is sent from the client computer 5 to the server 1, to ensure that the user interface can correctly indicate the active window to the user.

[0038] When a new window appears in the local client screen area 9 or in the central application screen area 10, either because an application has just been launched, or because the application has generated a new window in the course of its execution, this is detected. Thus, the new window can be displayed in the merged local client screen 16. A communication to the server 1 further leads to the creation of a button 20 in the central task bar 11 of the merged local client screen 16, in order that a user can switch to a newly initiated locally running application.

[0039] In a similar way, the disappearance of the window 19 associated with the locally running application is communicated to the server 1. The window can disappear when execution of the application has finished, or when there are several windows belonging to one locally running application and the application no longer needs one of them. Using the subscriber program, a signal is sent from the client computer 5 to the server 1 indicating that the window 19 has been closed, so that the user interface can display the correct buttons 12, 18, 20 in the task bar 11, both of applications running on the server 1 and of applications running on the client computer 5.

[0040] In an advantageous embodiment of the invention, a window further comprises a title area, not shown. The title can change in the course of running the application associated with a window. The buttons 12, 18, 20 comprise a caption relating to the title. If the title of the window 17 associated with the locally running application changes, a signal is generated and communicated to the server 1, using the interface management and subscriber programs. The captions in the central task bar 11 are then automatically updated.

[0041] Similarly, each button 12, 18, 20 comprises an icon representing the application associated with it. The

icon can also change in the course of time. Changes lead to an automatic update of the central application screen area 10 and merged local client screen area 16 in a manner similar to the mechanism used when a caption needs to be updated.

[0042] When the user clicks on one of the buttons 12, 18, 20, this input is signalled to the server 1. If the user has clicked on the button 18 associated with the application running on the server 1, the interface management program ensures that the user can control that application through the interface. If the user has clicked on the button 20 associated with the locally running application, a switch is made to that application. In this process, messages are exchanged between the server 1 and the client computer 5, which are handled by the subscriber program and the interface management program.

[0043] The protocol by which signals are exchanged between the client computer 5 and the server 1 further allows for the termination of all locally running applications. The server 1 sends a command to the client computer 5 to terminate the running applications. When all applications have successfully been shut down, this is communicated to the server 1, which can then update the user interface.

[0044] A user can also maximise the window 19 associated with the locally running application, as is shown in Fig. 2D. A maximised window 21 now fills the local client screen area 9. A transferred version 21' of the maximised window 21 in the merged local client screen 16 does not, however obscure the central task bar 11, so that a user can still switch between applications. This follows from the fact that the server comprises means for controlling the display of the local client screen 16 on the screen 7 of the client computer 5.

[0045] Turning now to Fig. 3, a more detailed description of the process, by which the running of an application on the client computer 5 is initiated, will be given. The process is started in a first step 22, by a user clicking on one of the icons 14, 15 for initiating locally run applications. The interface management program processes this input. Once it has been determined that the icon is associated with a local application on the client computer 5, a message is sent in a subsequent step 23 to the client computer 5, to check that the subscriber program is running. In a next step 24, the client computer 5 returns an acknowledgement message, generated using the subscriber program, to the effect that the subscriber program is running.

[0046] The interface management program on the server 1 generates a message, containing a command line, which is sent to the client computer 5 in a subsequent step 25. In a next step 26, the locally running application is launched on the client computer 5 by executing the command line.

[0047] The locally running application generates its window 19, which first appears in the local client screen area 9. As the merged local client screen 16 is automatically updated, the window 19 also appears in the merged

local client screen 16. The creation of the local application window 19 is symbolically depicted in step 27 of Fig. 3.

[0048] In a step 28 subsequent upon the creation of the local window 19, the subscriber program generates a message, signalling the creation of a new application window, which message is passed on to the server 1. This is the cue for a last step 29 in the flow chart of Fig. 3, namely the addition of the button 20 to the central task bar 11 in the merged local client screen 16. The local application is now fully integrated into the user interface and the situation depicted in Fig. 2C has been arrived at.

[0049] Those skilled in the art will appreciate that the invention is not limited to the above-described embodiment, which can be varied in a number of ways within the scope of the claims. For instance, embodiments of the system can exist that comprise additional means, e.g. function keys, for switching between windows and/or applications.

Claims

1. A server-based computing system comprising at least one server (1) comprising means for running an application and at least one client computer (5) comprising means (6) for locally running at least one further application, the client computer (5) and the server (1) being connected through a network (2), wherein the server (1) comprises:

- means for providing the client computer with a user interface comprising a central screen (10) having means (13,14,15) for initiating the application and the further application;
- means for providing a command to the client computer (5) over the network (2) to launch the further application after initiation of the further application, and

wherein the client computer (5) comprises:

- an input device (8) for providing input to the application and the further application;
- a display device for presenting output from the application and the further application, and
- means for launching the further application in response to receiving the command from the server (1) over the network (2),
- means for generating a local screen (9), said local screen being capable of comprising a window (19) for the further application generated in response to launching the further application;
- means for receiving the user interface comprising the central screen from the server; and
- means for merging the local screen (9) and the central screen (10) to display a merged screen (16) on the display device, such that the further application is fully integrated into the user inter-

face received from the server (1).

2. The Server-based computer system according to claim 1, wherein the server further comprises:

- means for detecting initiation of the further application at the client computer;
- means for determining that the detected initiation relates to the initiation of a locally run further application.

3. The server-based computer system according to claim 1 or 2, wherein the server is further configured to provide a taskbar comprising buttons for the application and the further application in the user interface for the client computer.

4. The server-based computer system according to claim 3, wherein the server is further configured to receive a signalling message from the client computer signalling that the window has been generated and to include the button for the further application in the taskbar in response to receiving the signalling message.

5. The server-based computer system according to claim 3, wherein said system is further configured to expose the taskbar on the display device upon maximizing the window associated with the further application.

6. The server-based computer system according to one or more of the preceding claims, wherein said client computer comprises means for terminating the further application and wherein the server comprises means for receiving a termination signal in response to terminating the application on the client computer and means for sending a termination command to the client computer to terminate the application in response to receiving the termination signal.

7. A method for providing a client computer (5) with a user interface for controlling at least one application that can be run on a server and at least one further application that can be run locally on the client computer (5), which client computer (5) is connected to the server (1) through a network (2) and comprises means (6) for running the further application, an input device (8) for providing input to the application and the further application and a display device (7) for presenting output from the application and the further application, the method comprising the steps of:

- providing the client computer (5) with a user interface comprising a central screen (10) having means for initiating the application and the further application by the server (1);
- providing a command from the server (1) to the

client computer (5) over the network (2) to launch the further application after initiation of the further application, and

- launching the further application at the client computer (5) in response to receiving the command from the server (1) over the network (2),
- generating, at the client computer (5), a local screen (9), said local screen (9) being capable of comprising a window (19) for the further application generated in response to launching the further application;
- receiving the user interface comprising the central screen (10) from the server (1); and
- merging the local screen (9) and the central screen (10) to display a merged screen (16) on the display device, such that the further application is fully integrated into the user interface.

8. The method according to claim 7, further comprising the steps of:

- detecting initiation of the further application at the client computer;
- determining that the detected initiation relates to the initiation of a locally run further application.

9. The method according to claim 7 or 8, further comprising the step of providing a taskbar from the server in the user interface for the client computer, the taskbar comprising buttons for the application and the further application.

10. The method according to claim 9, further comprising the steps of:

- receiving a signalling message at the server from the client computer signalling that the window has been generated;
- including the button for the further application in the taskbar in response to receiving the signalling message.

11. The method according to claim 9 or 10, further comprising the step of exposing the taskbar on the display device upon maximizing the window associated with the further application.

12. The method according to one or more of the claims 7-11, further comprising the steps of:

- receiving, by the server a termination signal in response to terminating the further application on the client computer; - sending a termination command to the client computer to terminate the application in response to receiving the termination signal.

13. Computer program that can be loaded onto a server

connected through a network (2) to a client computer (5), so that the server running the computer program constitutes a server (1) in a system according to any one of claims 1-6.

14. Computer program that can be loaded onto a computer connected through a network (2) to a server, so that the computer running the computer program constitutes a client computer (5) in a system according to any one of claims 1-6.

Patentansprüche

1. Serverbasiertes Rechensystem mit mindestens einem Server

(1) mit Mitteln, eine Anwendung durchführen zu lassen, und mit mindestens einem Client-Rechner (5) mit Mitteln, lokal mindestens eine weitere Anwendung durchführen zu lassen,

wobei der Client-Rechner (5) und der Server (1) durch ein Netzwerk (2) verbunden sind, wobei der Server (1) aufweist:

- Mittel, den Client-Rechner (5) mit einer Benutzeroberfläche auszustatten, welche eine zentrale Bildfläche (10) mit der Möglichkeit (13, 14, 15) des Initiierens der Anwendung und der weiteren Anwendung aufweist;
- Mittel zum Bereitstellen eines Befehls an den Client-Rechner (5) über das Netzwerk (2), um die weitere Anwendung nach Starten der weiteren Anwendung aufzurufen, und

wobei der Client-Rechner (5) aufweist:

- eine Eingabevorrichtung (8) zum Liefern von Eingaben an die Anwendung und die weitere Anwendung,
- eine Anzeigevorrichtung zum Darstellen von Ausgaben der Anwendung und der weiteren Anwendung, und
- Mittel zum Aufrufen der weiteren Anwendung als Reaktion auf das Empfangen des Befehls vom Server (1) über das Netzwerk (2);
- Mittel zum Erzeugen einer lokalen Bildfläche (9), wobei die lokale Bildfläche (9) zum Unterbringen eines als Reaktion auf das Aufrufen der weiteren Anwendung erzeugten Fensters (19) für die weitere Anwendung fähig ist;
- Mittel zum Empfangen der die zentrale Bildfläche enthaltenden Benutzeroberfläche vom Server; und
- Mittel zum Zusammenfügen der lokalen Bildfläche (9) und der zentralen Bildfläche (10), um eine zusammengefügte Bildfläche (16) auf der

Anzeigevorrichtung anzuzeigen, so dass die weitere Anwendung vollständig in die vom Server

- 5 (1) empfangene Benutzeroberfläche integriert ist.

2. Serverbasiertes Rechensystem nach Anspruch 1, wobei der Server weiter aufweist:

- Mittel zum Detektieren des Startens der weiteren Anwendung auf dem Client-Rechner;
- Mittel, um festzustellen, dass der detektierte Start sich auf das Starten einer lokal durchgeführten weiteren Anwendung bezieht.

3. Serverbasiertes Rechensystem nach Anspruch 1 oder 2, wobei der Server weiter so ausgelegt ist, dass er eine Taskleiste mit Schaltflächen für die Anwendung und die weitere Anwendung in der Benutzeroberfläche für den Client-Rechner bereitstellt.

4. Serverbasiertes Rechensystem nach Anspruch 3, wobei der Server weiter so ausgelegt ist, dass er eine Meldung vom Client-Rechner empfängt, welche meldet, dass das Fenster erzeugt worden ist und wobei der Server weiter so ausgelegt ist, dass die Schaltfläche für die weitere Anwendung als Reaktion auf das Empfangen der Meldung in die Taskleiste aufgenommen wird.

5. Serverbasiertes Rechensystem nach Anspruch 3, wobei das System weiter so ausgelegt ist, dass es die Taskleiste beim Maximieren des der weiteren Anwendung zugeordneten Fensters auf der Anzeigevorrichtung freigibt.

6. Serverbasiertes Rechensystem nach einem oder mehreren der vorangegangenen Ansprüche, wobei der Client-Rechner Mittel zum Beenden der weiteren Anwendung aufweist, und bei dem der Server Mittel zum Empfangen eines Beendigungssignals als Reaktion auf das Beenden der Anwendung auf dem Client-Rechner und Mittel zum Senden eines Beendigungsbefehls an den Client-Rechner, um die Anwendung als Reaktion auf das Empfangen des Beendigungssignals zu beenden, enthält.

7. Verfahren zum Bereitstellen eines Client-Rechners (5) mit einer Benutzeroberfläche zum Steuern mindestens einer auf einem Server durchführbaren Anwendung und mindestens einer lokal auf dem Client-Rechner (5) durchführbaren weiteren Anwendung, wobei selbiger Client-Rechner durch ein Netzwerk (2) mit dem Server (1) verbunden ist und Mittel (6) zum Ausführen der weiteren Anwendung, eine Eingabevorrichtung (8) zum Liefern von Eingaben an die Anwendung und die weitere Anwendung, und

eine Anzeigevorrichtung (7) zum Darstellen von Ausgaben der Anwendung und der weiteren Anwendung aufweist, wobei das Verfahren folgende Schritte aufweist:

- Ausstatten des Client-Rechners (5) mit einer Benutzeroberfläche mit einer zentralen Bildfläche (10) mit Mitteln zum Starten der Anwendung und der weiteren Anwendung durch den Server (1);
 - Bereitstellen eines Befehls von dem Server (1) an den Client-Rechner (5) über das Netzwerk (2), um die weitere Anwendung nach Starten der weiteren Anwendung aufzurufen, und
 - Aufrufen der weiteren Anwendung auf dem Client-Rechner (5) als Reaktion auf das Empfangen des Befehls vom Server (1) über das Netzwerk (2),
 - Erzeugen einer lokalen Bildfläche (9) am Client-Rechner (5), wobei die lokale Bildfläche (9) zum Unterbringen eines als Reaktion auf das Aufrufen der weiteren Anwendung erzeugten Fensters (19) für die weitere Anwendung fähig ist,
 - Empfangen der die zentrale Bildfläche (10) beinhaltenden Benutzeroberfläche vom Server (1); und
 - Zusammenfügen der lokalen Bildfläche (9) und der zentralen Bildfläche (10), um eine zusammengefügte Bildfläche (16) auf der Anzeigevorrichtung anzuzeigen, so dass die weitere Anwendung vollständig in die Benutzeroberfläche integriert ist.
8. Verfahren nach Anspruch 7, welches weiter folgende Schritte aufweist:
- Detektieren des Startens der weiteren Anwendung auf dem Client-Rechner;
 - Feststellen, dass der detektierte Start sich auf das Starten einer lokal durchgeführten weiteren Anwendung bezieht.
9. Verfahren nach Anspruch 7 oder 8, welches weiter den Schritt des Bereitstellens einer Taskleiste in der Benutzeroberfläche für den Client-Rechner durch den Server aufweist, wobei die Taskleiste Schaltflächen für die Anwendung und die weitere Anwendung aufweist.
10. Verfahren nach Anspruch 9, welches weiter folgende Schritte aufweist:
- Empfangen einer Meldung vom Client-Rechner am Server, welche meldet, dass das Fenster erzeugt worden ist;
 - Aufnehmen der Schaltfläche für die weitere Anwendung in die Taskleiste als Reaktion auf das

Empfangen der Meldung.

11. Verfahren nach Anspruch 9 oder 10, welches weiter den Schritt des Anzeigens der Taskleiste auf der Anzeigevorrichtung beim Maximieren des der weiteren Anwendung zugeordneten Fensters aufweist.
12. Verfahren nach einem oder mehreren der Ansprüche 7 - 11, welches weiter folgende Schritte aufweist:
- Empfangen eines Beendigungssignals durch den Server als Reaktion auf das Beenden der weiteren Anwendung auf dem Client-Rechner;
 - Senden eines Beendigungsbefehls an den Client-Rechner, um die Anwendung als Reaktion auf das Empfangen des Beendigungssignals zu beenden.
13. Rechenprogramm, welches auf einen durch ein Netzwerk (2) mit einem Client-Rechner (5) verbundenen Server geladen werden kann, so dass der das Rechenprogramm durchführende Server einen Server (1) in einem System gemäß jedem der Ansprüche 1 - 6 darstellt.
14. Rechenprogramm, welches auf einen durch ein Netzwerk (2) mit einem Server verbundenen Rechner geladen werden kann, so dass das Rechenprogramm durchführende Rechner einen Client-Rechner (5) in einem System gemäß jedem der Ansprüche 1 - 6 darstellt.

Revendications

1. Environnement d'ordinateur basé sur un serveur, comprenant au moins un serveur (1) doté de moyens pour exécuter une application et au moins un ordinateur client (5) comprenant des moyens (6) pour exécuter localement au moins une autre application, l'ordinateur client (5) et le serveur (1) étant reliés via un réseau (2), dans lequel le serveur (1) comprend :
- des moyens pour fournir à l'ordinateur client une interface utilisateur comprenant un écran central (16) doté de moyens (13, 14, 15) pour initier l'application et l'autre application ;
 - des moyens pour donner un ordre à l'ordinateur client (5) sur le réseau pour lancer l'autre application après avoir initié l'autre application, et
- dans lequel l'ordinateur client (5) comprend :
- un dispositif d'entrée (8) pour fournir une entrée à l'application et à l'autre application ;
 - un dispositif d'affichage pour présenter une sortie générée par l'application ou l'autre appli-

- cation, et
- des moyens pour lancer l'autre application en réponse à la réception de l'ordre du serveur (1) sur le réseau (2),
 - des moyens pour générer un écran local (9), ledit écran local étant capable de comporter une fenêtre (19) pour l'autre application, générée en réponse au lancement de l'autre application ;
 - des moyens pour recevoir du serveur l'interface utilisateur comprenant l'écran central ; et
 - des moyens pour fusionner l'écran local (9) et l'écran central (10) afin d'afficher un écran fusionné (16) sur le dispositif d'affichage, de telle façon que l'autre application soit totalement intégrée dans l'interface utilisateur reçue du serveur (1).
2. Environnement d'ordinateur basé sur un serveur selon la revendication 1, dans lequel le serveur comprend, en outre :
- des moyens pour détecter le lancement de l'autre application au niveau de l'ordinateur client ;
 - des moyens pour déterminer que le lancement détecté se rapporte au lancement d'une autre application exécutée localement.
3. Environnement d'ordinateur basé sur un serveur selon la revendication 1 ou 2, dans lequel le serveur est configuré, en outre, pour fournir une barre de tâches comprenant des boutons pour l'application et l'autre application dans l'interface utilisateur destinée à l'ordinateur client.
4. Environnement d'ordinateur basé sur un serveur selon la revendication 3, dans lequel le serveur est configuré, en outre, pour recevoir un message de signalisation de l'ordinateur client signalant que la fenêtre a été générée et pour inclure le bouton destiné à l'autre application dans la barre de tâches en réponse à la réception du message de signalisation.
5. Environnement d'ordinateur basé sur un serveur selon la revendication 3, dans lequel ledit environnement est, en outre, configuré pour exposer la barre de tâches sur le dispositif d'affichage lors de la maximisation de la fenêtre associée à l'autre application.
6. Environnement d'ordinateur basé sur un serveur selon l'une ou plusieurs des revendications précédentes, dans lequel ledit ordinateur client comprend des moyens pour arrêter l'autre application et dans lequel le serveur comprend des moyens pour recevoir un signal d'arrêt en réponse à l'arrêt de l'application sur l'ordinateur client et des moyens pour transmettre un ordre d'arrêt à l'ordinateur client pour arrêter l'application en réponse à la réception du signal d'ar-

rêt.

7. Procédé destiné à fournir une interface utilisateur à un ordinateur client (5) pour commander au moins une application qui peut être exécutée sur un serveur et au moins une autre application qui peut être exécutée localement sur l'ordinateur client (5), lequel ordinateur client (5) est relié au serveur (1) via un réseau (2) et comprend des moyens (6) pour exécuter l'autre application, un dispositif d'entrée (8) pour fournir une entrée à l'application et l'autre application et un dispositif d'affichage (7) pour présenter une sortie générée par l'application et l'autre application, le procédé comprenant les étapes suivantes :

- fournir à l'ordinateur client (5) une interface utilisateur comprenant un écran central (16) doté de moyens pour initier l'application et l'autre application par le serveur (1) ;
- donner un ordre du serveur (1) à l'ordinateur client (5) sur le réseau (2) pour lancer l'autre application après avoir initié l'autre application, et
- lancer l'autre application au niveau de l'ordinateur client (5) en réponse à la réception de l'ordre du serveur (1) sur le réseau (2),
- générer, au niveau de l'ordinateur client (5), un écran local (9), ledit écran local (9) étant capable de comporter une fenêtre (19) pour l'autre application, générée en réponse au lancement de l'autre application ;
- recevoir du serveur (1) l'interface utilisateur comprenant l'écran central (10) ; et
- fusionner l'écran local (9) et l'écran central (10) afin d'afficher un écran fusionné (16) sur le dispositif d'affichage, de telle façon que l'autre application soit totalement intégrée dans l'interface utilisateur.

8. Procédé selon la revendication 7, comprenant, en outre, les étapes suivantes :

- détecter le lancement de l'autre application au niveau de l'ordinateur client ;
- déterminer que le lancement détecté se rapporte au lancement d'une autre application exécutée localement.

9. Procédé selon la revendication 7 ou 8, comprenant, en outre, l'étape consistant à prévoir une barre de tâches fournie par le serveur dans l'interface utilisateur de l'ordinateur client, la barre de tâches comprenant des boutons pour l'application et l'autre application.

10. Procédé selon la revendication 9, comprenant, en outre, les étapes suivantes :

- recevoir, au niveau du serveur, un message de signalisation de l'ordinateur client signalant que la fenêtre a été générée ;
- inclure le bouton destiné à l'autre application dans la barre de tâches en réponse à la réception du message de signalisation. 5

11. Procédé selon la revendication 9 ou 10, comprenant, en outre, l'étape consistant à exposer la barre de tâches sur le dispositif d'affichage lors de la maximisation de la fenêtre associée à l'autre application. 10

12. Procédé selon l'une ou plusieurs des revendications précédentes, comprenant, en outre, les étapes suivantes : 15

- réception par le serveur d'un signal d'arrêt en réponse à l'arrêt de l'application sur l'ordinateur client ;
- transmission d'un ordre d'arrêt à l'ordinateur client pour arrêter l'application en réponse à la réception du signal d'arrêt. 20

13. Programme informatique pouvant être chargé sur un serveur relié via un réseau (2) à un ordinateur client (5) de façon que le serveur exécutant le programme informatique constitue un serveur (1) dans un environnement selon l'une quelconque des revendications 1 à 6. 25

30

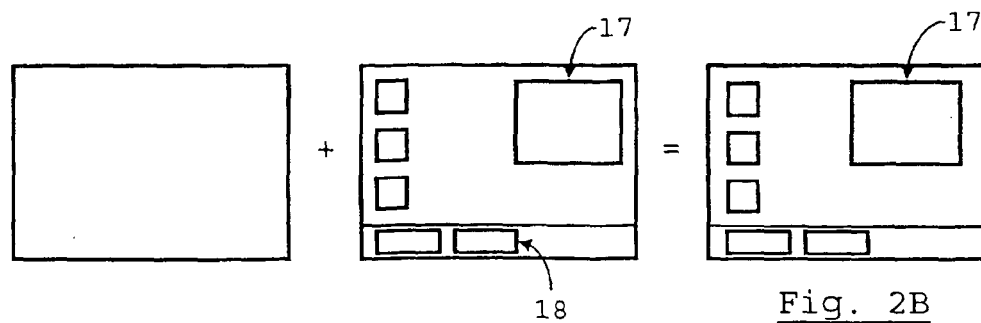
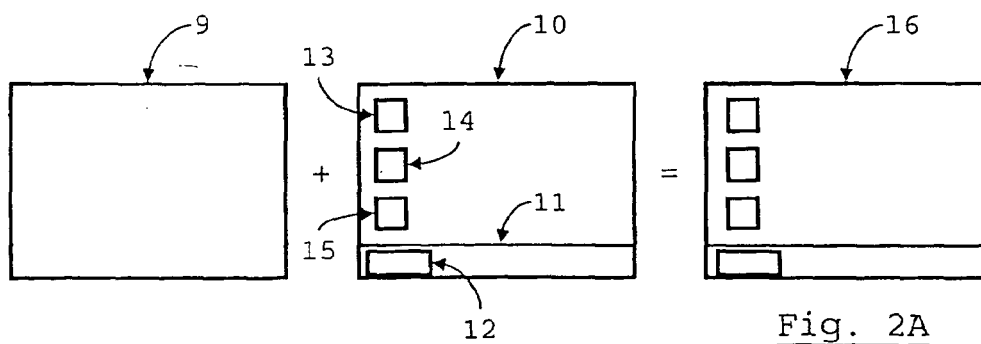
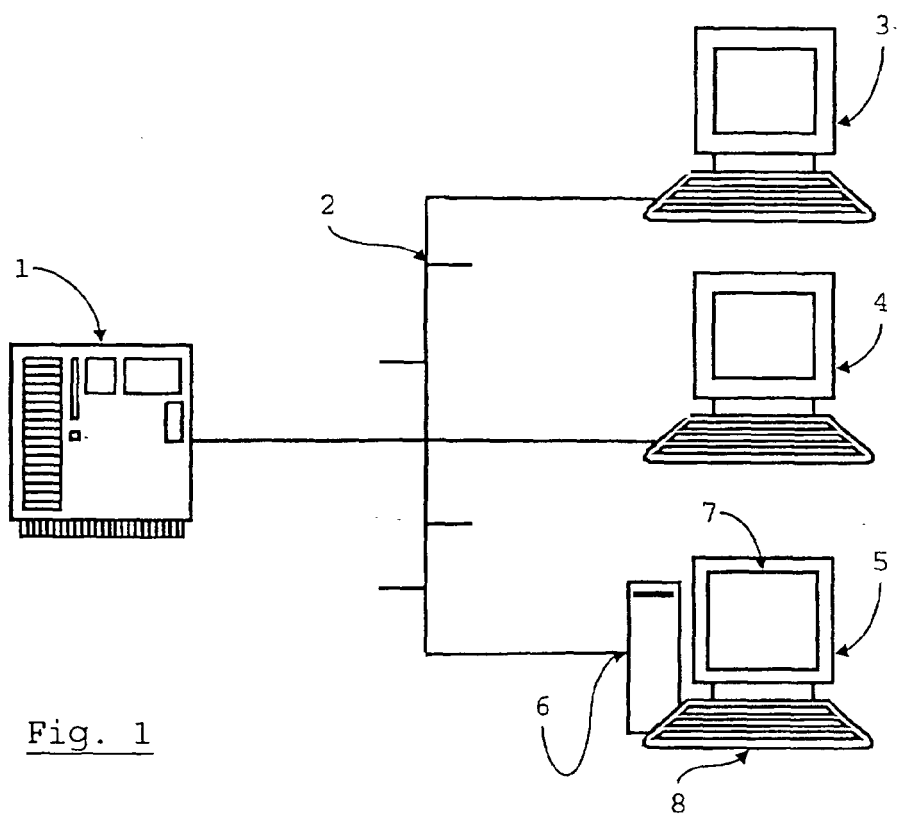
14. Programme informatique pouvant être chargé sur un ordinateur relié via un réseau (2) à un serveur de façon que l'ordinateur exécutant le programme informatique constitue un ordinateur client (5) dans un environnement selon l'une quelconque des revendications 1 à 6. 35

40

45

50

55



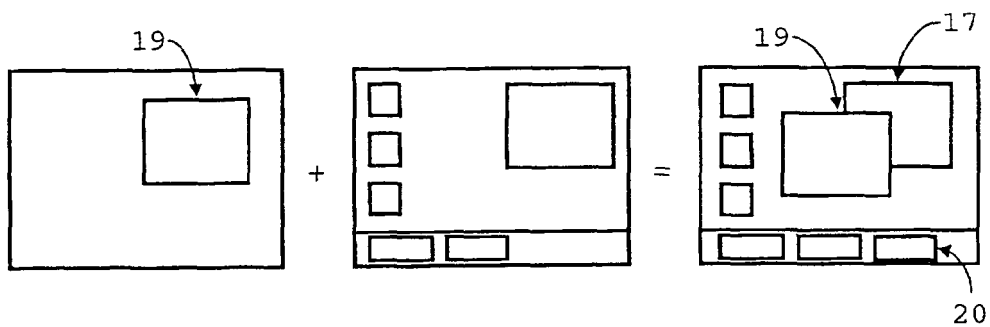


Fig. 2C

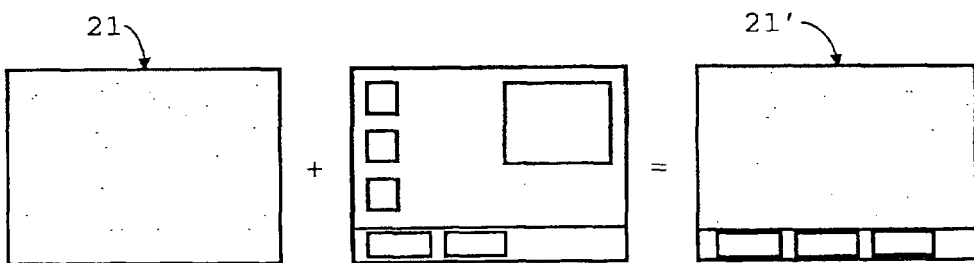


Fig. 2D

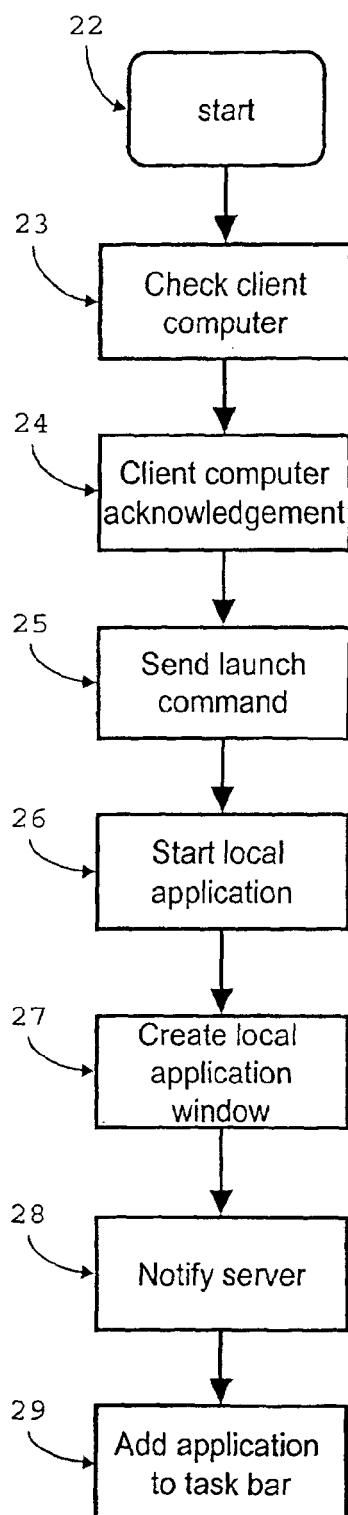


Fig. 3

REFERENCES CITED IN THE DESCRIPTION

This list of references cited by the applicant is for the reader's convenience only. It does not form part of the European patent document. Even though great care has been taken in compiling the references, errors or omissions cannot be excluded and the EPO disclaims all liability in this regard.

Patent documents cited in the description

- WO 9102305 A [0006]