

(19) World Intellectual Property Organization
International Bureau



(43) International Publication Date
7 December 2000 (07.12.2000)

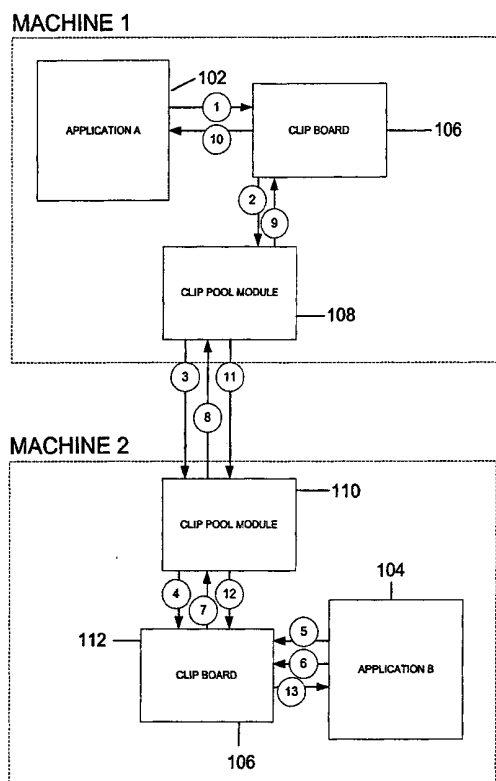
PCT

(10) International Publication Number
WO 00/73902 A1

- (51) International Patent Classification⁷: **G06F 9/46** (74) Agents: **PIPPENGER, Phillip, M.** et al.; Leydig, Voit & Mayer, Ltd., Suite 4900, Two Prudential Plaza, 180 North Stetson, Chicago, IL 60601-6780 (US).
- (21) International Application Number: PCT/US00/13663
- (22) International Filing Date: 18 May 2000 (18.05.2000) (81) Designated States (*national*): AE, AG, AL, AM, AT, AU, AZ, BA, BB, BG, BR, BY, CA, CH, CN, CR, CU, CZ, DE, DK, DM, DZ, 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, MX, MZ, NO, NZ, PL, PT, RO, RU, SD, SE, SG, SI, SK, SL, TJ, TM, TR, TT, TZ, UA, UG, UZ, VN, YU, ZA, ZW.
- (25) Filing Language: English
- (26) Publication Language: English
- (30) Priority Data:
09/324,432 1 June 1999 (01.06.1999) US
- (71) Applicant: **MICROSOFT CORPORATION** [US/US];
One Microsoft Way, Redmond, WA 98052 (US).
- (72) Inventor: **EILEBRECHT, Eric, L.**; 1310 Minor Avenue,
Apartment 209, Seattle, WA 98101 (US).
- (84) Designated States (*regional*): ARIPO patent (GH, GM, KE, LS, MW, MZ, SD, SL, SZ, TZ, UG, ZW), Eurasian patent (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), European patent (AT, BE, CH, CY, DE, DK, ES, FI, FR, GB, GR, IE, IT, LU, MC, NL, PT, SE), OAPI patent (BF, BJ, CF, CG, CI, CM, GA, GN, GW, ML, MR, NE, SN, TD, TG).

[Continued on next page]

(54) Title: SINGLE LOGICAL CLIPBOARD FOR MULTIPLE COMPUTERS



(57) Abstract: A single logical clipboard for use in a network environment utilizes a group of intercommunicative clip pool modules, each hosted on a network computer, to notify all computers which are members of a clip pool of changes in the contents of the local clipboard of any other member computer, and to transfer the contents of a member computer's local clipboard to the local clipboard of another member computer.

WO 00/73902 A1

**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.

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.

SINGLE LOGICAL CLIPBOARD FOR MULTIPLE COMPUTERS

TECHNICAL FIELD

This invention relates generally to clipboards for sharing information between
5 computer applications, and more particularly relates to a single logical clipboard which
may be shared transparently by separate applications running on multiple network
computers.

BACKGROUND OF THE INVENTION

10 It is often necessary for a computer user to transfer information from one
application to another. For example, an application may calculate output information
which is to be cited in a memorandum being prepared using a separate word processing
application. Typically, where the applications of interest reside on the same machine, the
user may utilize a clipboard or similar utility to hold information being cut or copied from
15 the first application until the information is inserted or pasted into the second application.
The user interaction in such instances consists simply of a cut/copy operation in one
application and a paste operation in the other.

This automatic sharing of clipboard information is not traditionally available
where the applications sharing the information reside on separate machines. Instead, in
20 such circumstances, it is necessary for the user to engage in more extensive intervention
to accomplish the transfer of information. For example, a user may physically transfer a
computer-readable disc containing the information from one machine to another.
Alternatively, the user may manually save the information to a file accessible by both
computers via a network.

25 Another existing mechanism for accomplishing such a transfer is to allow one
machine to retrieve the clipboard contents of the other machine using a utility such as
"CLIPBOOK VIEWER," which accompanies the "MICROSOFT" brand "WINDOWS"
operating system. However, this technique requires the user of a first machine to
manually indicate that a clipboard is to be shared, and name the clipboard, after which a
30 user of a second machine must manually request the clipboard by identifying the first
machine and by requesting the clipboard by name.

"MICROSOFT TERMINAL SERVER," to be included in "MICROSOFT WINDOWS 2000" operating system, allows some shared clipboard functionality. However, "MICROSOFT TERMINAL SERVER" is quite limited in that it works only between pairs of machines, and requires that one acts as a "client", while the other acts as a "server." It also supports only nonmutual sharing, from the client to the server.

It is clear that the currently existing mechanisms for transferring data or information between computers do not allow mutual access to a single clipboard by an arbitrary number of machines. Accordingly, a clipboard utility is needed which will allow for the mutual and automatic sharing of information by an arbitrary number of machines running in a networked environment.

SUMMARY OF THE INVENTION

In accordance with this need, the present invention is generally realized in a single logical clipboard which is mutually accessible by all machines that are members of a "clip pool." Each member of the clip pool automatically uses the single logical clipboard as if it were its own clipboard. As an example, consider a computer network comprised of a Computer 1, upon which resides Application A, and a Computer 2, upon which resides Application B, both computers being members of the same clip pool. Using the invention, a user may cut data from within Application A, and subsequently paste the same data within Application B because the single logical clipboard automatically makes available to all clip pool members the data cut by Application A. A clip pool according to the invention may have any number of members.

In order to accomplish the above-described functionality, an application which will be referred to as a "clip pool module" is preferably running on each member machine. Each clip pool module registers itself locally as a "clipboard viewer" application. This ensures that it is informed of any changes to the local clipboard. When informed of such a change, the clip pool module in turn notifies all members of the same pool of the identity of the machine and the various formats available for the data. When an application hosted by a member of the pool initiates an operation which will use the shared clipboard data, the clip pool module running on that machine identifies the last machine to indicate a clipboard update, and requests the clipboard data from the clip pool module running on that machine in the appropriate format. For the computer user, this

entire process is automatic, appearing as only two user steps: a cut (or other data acquisition operation) on one machine, and a paste (or other data deposition operation) on another.

5

BRIEF DESCRIPTION OF THE DRAWINGS

While the appended claims set forth the features of the present invention with particularity, the invention, together with its objects and advantages, may be best understood from the following detailed description taken in conjunction with the accompanying drawings of which:

10 FIG. 1 is a schematic of an exemplary computer system usable in implementing the invention;

 FIG. 2 is a block diagram illustrating the operation of a standard prior art clipboard;

 FIG. 3 is a block diagram illustrating the operation of a single logical clipboard
15 according to an embodiment of the invention;

 FIG. 4 illustrates the preferred structure of a network message;

 FIG. 5 illustrates the preferred data field structure of the CLIPBOARD_CHANGE and CLIPBOARD_REQUEST messages;

 FIG. 6 illustrates the structure of an exemplary network data transmission
20 containing clipboard data;

 FIG. 7 is a block diagram illustrating the operation of a single logical clipboard according to an alternative embodiment of the invention.

DETAILED DESCRIPTION OF THE INVENTION

25 Turning to the drawings, wherein like reference numerals refer to like elements, the invention is illustrated as being implemented in a suitable computing environment. Although not required, the invention will be described in the general context of computer-executable instructions, such as program modules, being executed by a personal computer. Generally, program modules include routines, programs, objects, components,
30 data structures, etc. that perform particular tasks or implement particular abstract data

types. Moreover, those skilled in the art will appreciate that the invention may be practiced with other computer system configurations, including hand-held devices, multi-processor systems, microprocessor based or programmable consumer electronics, network PCs, minicomputers, mainframe computers, and the like. Steps or elements of the invention discussed as occurring or being located upon one computer may also be practiced in distributed computing environments where tasks are performed by remote processing devices that are linked through a communications network. In a distributed computing environment, program modules may be located in both local and remote memory storage devices.

With reference to Fig. 1, part of an exemplary system for implementing the invention includes a general purpose computing device in the form of a conventional personal computer 20, including a processing unit 21, a system memory 22, and a system bus 23 that couples various system components including the system memory to the processing unit 21. The system bus 23 may be any of several types of bus structures including a memory bus or memory controller, a peripheral bus, and a local bus using any of a variety of bus architectures. The system memory includes read only memory (ROM) 24 and random access memory (RAM) 25. A basic input/output system (BIOS) 26, containing the basic routines that help to transfer information between elements within the personal computer 20, such as during start-up, is stored in ROM 24. The personal computer 20 further includes a hard disk drive 27 for reading from and writing to a hard disk, not shown, a magnetic disk drive 28 for reading from or writing to a removable magnetic disk 29, and an optical disk drive 30 for reading from or writing to a removable optical disk 31 such as a CD ROM or other optical media.

The hard disk drive 27, magnetic disk drive 28, and optical disk drive 30 are

connected to the system bus 23 by a hard disk drive interface 32, a magnetic disk drive interface 33, and an optical disk drive interface 34, respectively. The drives and their associated computer-readable media provide nonvolatile storage of computer readable instructions, data structures, program modules and other data for the personal computer

5 20. Although the exemplary environment described herein employs a hard disk, a removable magnetic disk 29, and a removable optical disk 31, it will be appreciated by those skilled in the art that other types of computer readable media which can store data that is accessible by a computer, such as magnetic cassettes, flash memory cards, digital video disks, Bernoulli cartridges, random access memories, read only memories, and the like may also be used in the exemplary operating environment.

A number of program modules may be stored on the hard disk, magnetic disk 29, optical disk 31, ROM 24 or RAM 25, including an operating system 35, one or more applications programs 36, other program modules 37, and program data 38. A user may enter commands and information into the personal computer 20 through input devices 15 such as a keyboard 40 and a pointing device 42. Other input devices (not shown) may include a microphone, joystick, game pad, satellite dish, scanner, or the like. These and other input devices are often connected to the processing unit 21 through a serial port interface 46 that is coupled to the system bus, but may be connected by other interfaces, such as a parallel port, game port or a universal serial bus (USB). A monitor 47 or other 20 type of display device is also connected to the system bus 23 via an interface, such as a video adapter 48. In addition to the monitor, personal computers typically include other peripheral output devices, not shown, such as speakers and printers.

A personal computer 20 used to implement the present invention may operate in a networked environment using logical connections to one or more remote machines, such

as remote computer 49. The remote machine 49 may be another personal computer, a server, a router, a network PC, a peer device or other common network node, and typically includes many or all of the elements described above relative to the personal computer 20, although only a memory storage device 50 has been illustrated in Fig. 1.

5 The logical connections depicted in Fig. 1 include a local area network (LAN) 51 and a wide area network (WAN) 52. Such networking environments are commonplace in offices, enterprise-wide computer networks, intranets and the Internet.

When used in a LAN networking environment, the personal computer 20 is connected to the local network 51 through a network interface or adapter 53. When used
10 in a WAN networking environment, the person computer 20 typically includes a modem 54 or other means for establishing communications over the WAN 52. The modem 54, which may be internal or external, is connected to the system bus 23 via the serial port interface 46. Program modules depicted relative to the personal computer 20, or portions thereof, may be stored in the remote memory storage device.

15 In the description that follows, the invention will be described with reference to acts and symbolic representations of operations that are performed by one or more computers, unless indicated otherwise. As such, it will be understood that such acts and operations, which are at times referred to as being computer-executed, include the manipulation by the processing unit of the computer of electrical signals representing data
20 in a structured form. This manipulation transforms the data or maintains it at locations in the memory system of the computer, which reconfigures or otherwise alters the operation of the computer in a manner well understood by those skilled in the art. The data structures where data is maintained are physical locations of the memory that have particular properties defined by the format of the data. However, while the invention is

being described in the foregoing context, it is not meant to be limiting as those of skill in the art will appreciate that various of the acts and operation described hereinafter may also be implemented in hardware.

The operation of a traditional clipboard usable by applications residing on a single machine is illustrated in Fig.2. The machine 100 is shown hosting an Application A 102 and an Application B 104. The applications are illustrated as having mutual access to the clipboard 106. Upon execution of a data acquisition operation such as "cut" or "copy" in Application A, Application A notifies the clipboard that data is available to the clipboard, and furnishes to the clipboard a list of the available data formats in step 1. At this point, the Application may or may not supply the actual data in the various formats to the clipboard. In response to a subsequent data deposition operation such as "paste" in Application B 104, Application B queries the clipboard for available formats in step 2, identifies an acceptable format, and requests the data from the clipboard in that format in step 3. If the clipboard has already received the data from Application A, it then sends the data to Application B in the appropriate format. If the clipboard does not have the data, it requests the data from Application A in the appropriate format in step 4, and then forwards the data to Application B in step 5. The user's only interaction throughout this process is to initiate the cut and paste operations, or other data acquisition and deposition operations.

In accordance with the invention, the model of a single logical clipboard is extended to machines connected across a network as depicted in Fig. 3. An application 108, 110 which will be referred to as a "clip pool module" is installed on each member of a group of machines that will share a logical clipboard. Such a group will be referred to

as a "clip pool" of machines. Other machines which are not pool members may host the clip pool module as well, but will not share the single logical clipboard.

When Machine 1 hosting the clip pool module is first powered on after being in a power off state, the user preferably specifies, via a command line option or similar

5 mechanism, the name of the clip pool to which Machine 1 will belong. The clip pool module 108 then sends an identity broadcast network message containing its identity and the name of the clip pool to which it belongs. This is designated as a NEW_PARTICIPANT message. Examples of other messages that may be sent via the network include: OLD_PARTICIPANT (sent in response to a new machine joining the
10 pool); DEAD_PARTICIPANT (indicates that this machine is leaving the pool); CLIPBOARD_CHANGE (indicates that the contents of the machine's clipboard have changed); and CLIPBOARD_REQUEST (requests clipboard data in a particular format).

Machines such as Machine 2 which receive this message and are members of the same clip pool add the identity of the sending machine to a list of pool members, and then
15 respond to the sending machine with similar information regarding their own identity and clip pool membership. It is preferably also possible for the user of a machine to manually specify other members of a pool via a command line option. This may be necessary where, for example, a clip pool member does not receive broadcast messages due to an intervening router or such, or where network congestion hinders receipt of the broadcast
20 message.

The clip pool module 108 preferably registers itself with the local clipboard 106 as an application which will receive update information anytime a change is made to the local clipboard contents. Using the "MICROSOFT" brand "WINDOWS" operating system, this may be accomplished by registering as a "clipboard viewer" via the standard

"WIN32" application programming interface *SetClipboardViewer()*. The clipboard viewer utility is ordinarily utilized to support an application which contemporaneously displays the local clipboard contents. The clip pool module 108, by being registered via this utility, will be notified of any changes in the contents of the local clipboard 106.

5 Similarly, other instances of the clip pool module, such as the instance 110 running on Machine 2, preferably register to receive update information regarding their local clipboard 112.

In this discussion of an exemplary operation using the single logical clipboard of the invention, it will be assumed that information is placed on the clipboard by

10 Application A 102 and retrieved from the clipboard by Application B 104. However, the shared logical clipboard is mutually accessible by all clip pool members in the same way. When a user initiates an operation such as "cut" or "copy" within Application A 102, the corresponding local clipboard 106 is automatically informed by Application A 102 of the availability of the data, as well as the available data formats in step 1. Because the clip
15 pool module 108 is registered to receive updates regarding the local clipboard contents, the clipboard 106 notifies the clip pool module 108 of the available data and formats in step 2.

At this point, the clip pool module 108 informs the clip pool module 110 running on the other clip pool members (Machine 2) that the local clipboard 106 of Machine 1 has
20 been updated, and lists the available formats in step 3. In turn, the clip pool module 110 running on Machine 2 records the identity of Machine 1 as the most recent clip pool member to indicate a clipboard update, and records the available data formats. Clip pool module 110 then notifies the local clipboard 112 of Machine 2 of the availability and format options of the data in step 4.

If a user of Application B 104 on Machine 2 subsequently initiates a data deposition operation such as "paste," which requires clipboard data, Application B queries the local clipboard 112 for available data formats in step 5, selects a format, and requests the data in that format in step 6. In step 7, the local clipboard 112 queries the clip pool module 110 for the data in the appropriate format. In this embodiment, the clip pool module 110 does not have the data, but has recorded Machine 1 as the last machine to indicate a local clipboard update. Therefore, clip pool module 110 contacts clip pool module 108 via the network connection in step 8 and requests the data in the appropriate format. Clip pool module 108 retrieves the local clipboard data by means such as the "WIN32" clipboard application programming interfaces in step 9. This may in turn require the local clipboard 108 to retrieve the data from Application A (step 10) if delayed rendering is being utilized. The data is at this point passed from clip pool module 108 to clip pool module 110 across the network connection (step 11), and from clip pool module 110 to local clipboard 112 (step 12). Subsequently, the data is passed to Application B by the use of the standard WIN32 clipboard application programming interfaces (step 13).

As with the use of a standard clipboard utility on a single machine, the above-described use of a single logical clip pool is automatic from the perspective of the user; once a clip pool has been established, the user can simply cut data from one application on one machine and paste the data into another application on another machine in the same manner as if both applications resided on the same machine. It can be appreciated that the single logical clipboard is also mutually accessible by all members of the clip pool.

Network communications in the above described exemplary implementation are preferably by way of named pipes and mail slots, two network communication devices

available in the "MICROSOFT" brand "WINDOWS" operating system. The named pipes protocol is a streaming protocol; that is, a special file is opened having a name such as `\\.\pipe\[name]`. This file may be written to or read, but the file data is not stored in a computer readable memory. Rather, an application writing to the file is blocked pending
5 a read request by another application. Upon receiving such a request, the file data is transferred from the writing application to the reading application. Access to this file across the network requires that both the machine name and file name be identified. Thus, Application B on Machine B may access such a file on Machine A using the address `\\MachineA\pipe\ClipPool\[name]`. Data sent via this mechanism may be a stream
10 of arbitrary length, and preferably includes header information.

The mail slots device is a network communications device which is packet oriented rather than stream oriented. A mail slot is created in much the same way as a named pipe. It similarly does not correspond to a stored file, but rather corresponds to a thread running on the receiving machine which waits for the data before processing the
15 data and subsequently looping back to the pending state.

According to the naming convention for mail slots and named pipes, a mail slot may be created and identified as `\\.\mailslot\ClipPool\[name]`. The name field in either protocol is preferably populated with the clip pool name selected by the user to identify the particular clip pool. Thus, a named pipe for a clip pool which the user has called
20 "mymachines" may be named `\\.\pipe\ClipPool\mymachines`. In the context of the "MICROSOFT" brand "WINDOWS" operating system, mail slots and named pipes may be manipulated using the following application programming interfaces: `CreateFile`; `WriteFile`; `ReadFile`; `CloseHandle`; `CreateNamedPipe`; `ConnectNamedPipe`; `DisconnectNamedPipe`; and `CreateMailslot`. For further detailed reading on the use of

mail slots and named pipes, the reader is referred to *Network Programming in Windows NT*, by Alok K. Sinha (Addison-Wesley, 1996).

In accordance with the invention, network messages such as the following may be exchanged across the network between clip pool nodes: NEW_PARTICIPANT (indicates that this machine is joining the pool); OLD_PARTICIPANT (sent in response to a new machine joining the pool); DEAD_PARTICIPANT (indicates that this machine is leaving the pool); CLIPBOARD_CHANGE (indicates that the contents of the machine's clipboard have changed); and CLIPBOARD_REQUEST (requests clipboard data in a particular format). Additionally, the actual clipboard data is transmitted over the network at the appropriate time as described above. Mail slots are preferably used for all network communications except the CLIPBOARD_CHANGE and CLIPBOARD_REQUEST messages, and the clipboard data transmission itself. These transmissions may be of arbitrary length and thus preferably use the named pipes protocol which does not limit the length of the data transmission. The preferred structure of network messages is exemplified in Fig. 4.

For NEW_PARTICIPANT, OLD_PARTICIPANT, and DEAD_PARTICIPANT messages, the data field is preferably empty. For CLIPBOARD_CHANGE and CLIPBOARD_REQUEST messages, the data field structure is preferably as shown in Fig.5.

When a CLIPBOARD_REQUEST message is sent over the network, specifying a data format and requesting data, the message is preferably sent using the named pipe mechanism described above. When the clipboard contents of the machine receiving the request are subsequently transmitted, the transmission is made over the same named pipe used for the data request. The structure of the data transmission is illustrated in Fig. 6.

Regarding the format of the clipboard data, a format designation which uniquely specifies a certain format to one network machine may describe a different or non-existent format to a another network machine. For example, using the "MICROSOFT" brand "WINDOWS" operating system, there exist certain standard formats which are defined by a number which is the same for all machines using this operating system. However, it is also possible for an application to register a custom format with the operating system on start up, after which a number is automatically assigned to designate the format. This number is only unique and meaningful within that machine. Thus, a subsequent network message from that machine identifying available data formats preferably will preferably not use that number to describe the custom format. Rather, a naming convention should be used which allows the receiving machine to identify the particular custom format regardless of the number assigned to the format by the sending machine.

As described above, the data field of a CLIPBOARD_CHANGE notification preferably lists available data formats. Preferably, the formats are separated in such a way tat the receiving clip pool module can identify individual formats. For example, formats may be separated an appropriate section of a transmission by a null character or other distinct separator. Preferably, standard system defined formats are preceded by a character such as a number sign or other character, while custom formats are preceded by a different character such as the dollar sign as illustrated. Preferably, the string identifier for custom formats contains sufficient information for the receiving clip pool module to identify a particular custom format referenced. In this manner, a receiving clip pool module may process both standard and custom formats.

It has been found empirically that certain standard formats defined in the "MICROSOFT" brand "WINDOWS" operating system do not work properly with the

above described method. Accordingly, the following standard formats are preferably not processed by the clip pool module: CF_GDIOBJFIRST through CF_GDIOBJLAST; CF_BITMAP; CF_ENHMETAFILE; CF_HDROP; CF_LOCALE; CF_METAFILEPICT; CF_OWNERDISPLAY; and CF_PALETTE.

5 In an alternative embodiment, clipboard data is transferred as soon as it becomes available. Thus in the above example, if this alternative embodiment were to be employed, all messages which were described as conveying the availability and formats of the clipboard data would now also include the data itself in the enumerated formats.

 In an alternative embodiment, it is also possible to utilize the single logical
10 clipboard of the invention to share data automatically and mutually between applications in different sessions on a single multi-user machine. Generally, a machine may allow two or more users to simultaneously log on by dividing and partitioning the memory and resources of the machine among the sessions such that each user has the illusion of being the sole user of the machine. Each session traditionally may have an associated clipboard.
15 According to this embodiment, a single logical clipboard is created such that data placed on the clipboard by an application in one session pursuant to a data acquisition operation may be automatically retrieved by an application in another session running on the same machine pursuant to a data deposition operation.

 Thus the single logical clipboard in this embodiment is actually physically
20 centrally located rather than physically distributed, and the named pipes and mail slots mechanisms are preferably not used. Rather, a single common clipboard is created on the machine such that partitioning does not apply to the clipboard, with respect to members of a clip pool. Thus, member sessions actually share one clipboard, logically and physically.

Figure 7 schematically illustrates this architecture. Terminals A 200, B 202, and C 218

are illustrated as the interfaces for sessions A, B, and C running simultaneously on the shared Machine 208. These sessions are implemented in the shared Machine 208 by specifying which memory locations may be used by which session. These memory locations, 204, 206, and 214 respectively, need not be contiguous or sequential. As well,
5 other machine resources are also divided among sessions. For example, processor time may be temporally divided among sessions. The common clipboard 210 represents memory locations in the machine 208 which are accessible by member sessions.

In a similar manner as the previous embodiment, it is preferred that no session is able to unilaterally authorize the sharing of another session's clipboard data. Thus,
10 preferably each session may add itself to a list 212 of pool members, and may only access clipboard data of the other members of the pool. The list 212 of pool members may be for example stored in a memory location on the Machine 208 which is accessible to every session running on that machine. Thus, in Fig. 7, Sessions A 204 and B 206 are pool members with access to the common clipboard 210, while Session C 214 is not a pool
15 member, and has access only to its own clipboard 216. Furthermore, sessions A and B preferably cannot access the clipboard 216 of Session C 214. The connections depicted in Fig. 7 between sessions, clipboards, and the list are not intended to necessarily illustrate external physical connections, but rather denote simply access to memory locations.

Although an embodiment of the invention has been described with reference to
20 two machines, both of which are members of a clip pool, the same method applies to more than two machines connected over a network, wherein at least two machines are members of a clip pool. In such an environment, the NEW_PARTICIPANT message preferably is still broadcast over the whole network as described above, the CLIPBOARD_CHANGE message is sent to all pool members, and all

OLD_PARTICIPANT messages and data requests and transmissions are between two computers as described above.

All of the references cited herein are incorporated in their entireties by reference.

In view of the many possible embodiments to which the principles of this

5 invention may be applied, it should be recognized that the embodiments described herein with respect to the drawing figures are meant to be illustrative only and should not be taken as limiting the scope of invention. For example, those of skill in the art will recognize that the elements of the illustrated embodiments shown in software may be implemented in hardware and vice versa or that the illustrated embodiments can be
10 modified in arrangement and detail without departing from the spirit of the invention. Those of skill in the art will recognize that the terms "computer" and "machine" are not intended to limit the invention to the embodiments described, but may include other configurations, including but not limited to hand-held devices, multi-processor systems, microprocessor based or programmable consumer electronics, network PCs,
15 minicomputers, mainframe computers, and the like. Those of skill in the art will also recognize that network connections may be entirely physical connections such as wires, or may include non-physical connections such as radio or optical transmissions. Although the single logical clipboard of the invention is described as being shared by different applications, all or some of these applications may be instances of the same application.
20 For example, multiple instances of "MICRSOFT WORD" brand word processing software may reside on multiple machines which share a single logical clipboard across a network in keeping with the invention. Furthermore, although many network communications are described as using named pipes or mail slots, one skilled in the art will realize that there exist many other methods to accomplish the same communications.

For example, the communications described as utilizing the mail slots technique may alternatively take place by way of named pipes or other mechanism. Therefore, the invention as described herein contemplates all such embodiments as may come within the scope of the following claims and equivalents thereof.

CLAIMS**We claim:**

1. A method of mutually and automatically sharing clipboard data between at least a
5 first and second computer connected to a network, the first computer having a first local
clipboard and the second computer having a second local clipboard, the method
comprising the steps of:
 automatically sending a notification from a selected one of the first and second
computers to another of the first and second computers that clipboard data is available to
10 be shared; and
 automatically transferring the clipboard data from the selected computer to the
other computer, whereby the clipboard data is automatically shared between the first local
clipboard and the second local clipboard regardless of which of the first and second
computers is selected.
15
2. The method according to claim 1, wherein each computer hosts a clip pool
module, each of which monitors the local clipboard of the host computer, wherein the step
of automatically sending a notification from a selected computer to another computer that
clipboard data is available to be shared comprises the step of:
20 the clip pool module of the selected computer sending a notification to the
clip pool module of the other computer that clipboard data is available to be shared;
 and the step of automatically transferring the clipboard data from the selected
computer to the other computer, whereby the clipboard data is automatically shared
between the first local clipboard and the second local clipboard, comprises the steps of:
25 automatically transferring the clipboard data from the first local clipboard
to the clip pool module of the selected computer;
 transferring the clipboard data from the clip pool module of the selected
computer to the clip pool module of the other computer; and
 transferring the clipboard data from the clip pool module of the other
30 computer to the second local clipboard.

3. The method according to claim 2, wherein the selected and other computer are members of a clip pool, a third computer is connected to the network, the third computer hosting a third clip pool module and a third local clipboard, the method further including the steps of:

5 determining whether the third computer is a member of the clip pool; and
if the third computer is a member of the clip pool, the third computer automatically receiving the notification that data is available to be shared.

4. The method according to claim 3, wherein the step of the third computer
10 automatically receiving the notification that data is available to be shared comprises the steps of:

the clip pool module of the selected computer receiving notification from the first local clipboard that data is available to be shared;

15 the clip pool module of the third computer receiving notification from the clip pool module of the selected computer that data is available to be shared; and

the third local clipboard receiving notification from the clip pool module of the third computer that data is available to be shared.

5. A method of mutually and automatically sharing clipboard data between a first and
20 second computer connected to a network, wherein the first and second computers host first and second local clipboards respectively, and first and second clip pool modules respectively, the method comprising the steps of:

automatically receiving at the first clip pool module the clipboard data from the first local clipboard;

25 automatically receiving at the second clip pool module the clipboard data from the first clip pool module; and

automatically receiving at the second local clipboard the clipboard data from the second clip pool module.

30 6. For use in a computer network, an apparatus for transferring clipboard data between selected network computers, the apparatus comprising:

network connections communicatively coupling each of the selected network

computers with all other selected network computers; and

a single logical clipboard operable to monitor the local clipboard contents of each selected computer and to automatically transfer the contents of a local clipboard associated with any of the selected network computers to a local clipboard associated with
5 any other of the selected network computers, using the network connections.

7. The apparatus of claim 6, wherein the single logical clipboard comprises:

one clip pool module residing on each of at least two network computers, whereby each clip pool module is operable to monitor the local clipboard of the computer hosting
10 the module and to notify another clip pool module residing on another network computer of a change to the contents of the local clipboard, and to send the contents of the local clipboard to another clip pool module, and to receive a notification from another clip pool module of a change in the contents of the local clipboard monitored by that other clip pool module, and to receive the contents of the local clipboard monitored by that other clip
15 pool module.

8. The apparatus of claim 6, wherein the single logical clipboard comprises:

one clip pool module residing on each of at least two network computers, whereby each clip pool module is operable to monitor the local clipboard of the computer hosting
20 the module and to send the contents of the local clipboard to another clip pool module, and to receive the contents of the local clipboard monitored by another clip pool module.

9. For use by each computer in a computer network comprised of multiple computers, of which selected computers are associated with a group of computers,
25 wherein each group member hosts a local clipboard, a computer-readable medium containing computer-readable instructions for performing the steps of:

monitoring the local clipboard of the host computer to determine when the local clipboard contents change;

if the contents of the local clipboard of the host computer change and if the host
30 computer is a member of the group, sending a notification to all other members of the group indicating that the contents of the local clipboard of the host computer have changed;

receiving a request from another group member for the contents of the local clipboard of the host computer; and

in response to the receipt of said request, transferring the contents of the local clipboard of the host computer to the requesting group member.

5

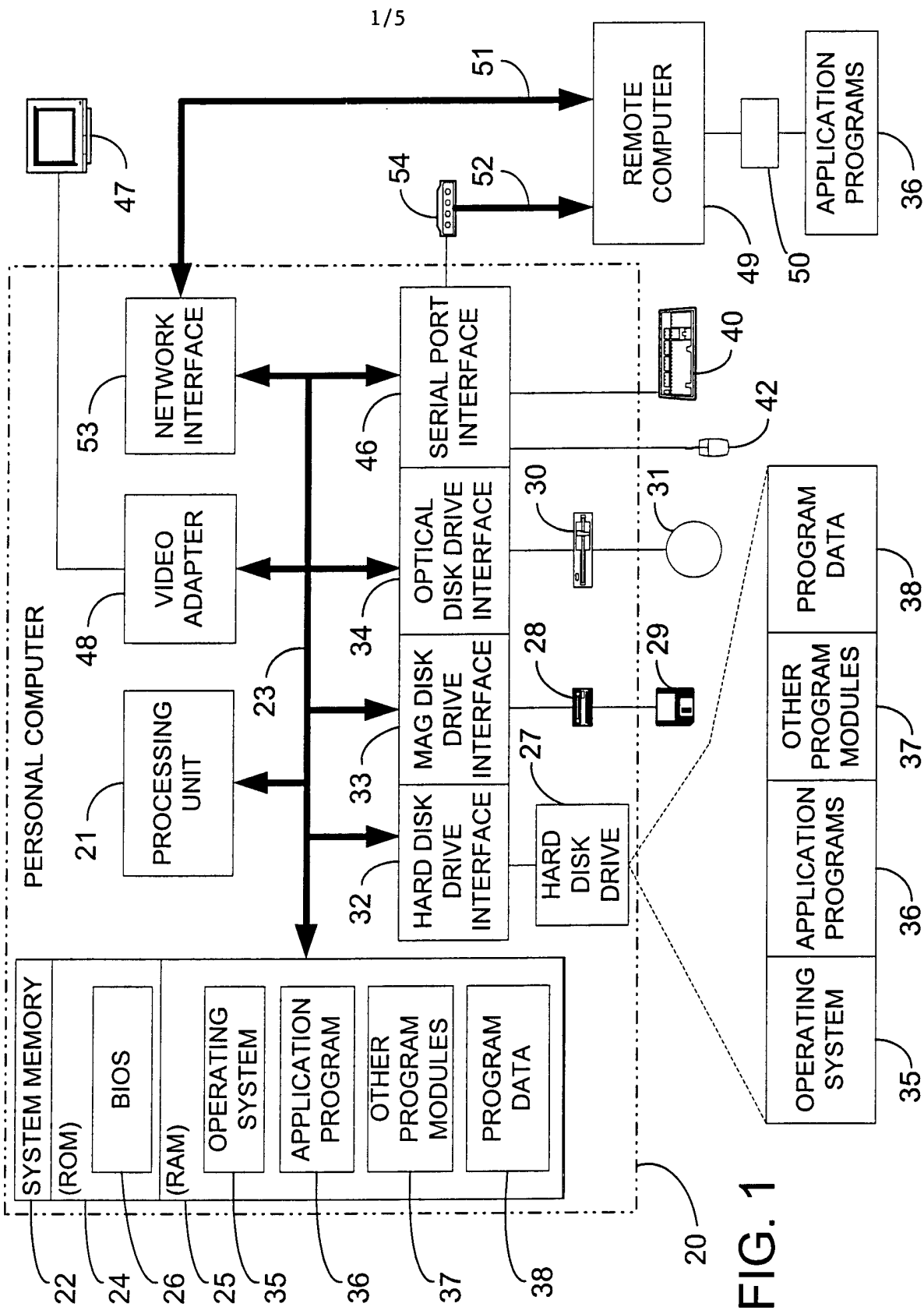
10. A method of mutually and automatically sharing clipboard data between two applications running in different sessions on a computer, comprising the steps of:

establishing a list of clip pool members stored in the memory of the computer, wherein the list is accessible to applications running in all sessions on the computer;

10 establishing a common clipboard which is accessible to any member session, wherein a member session is any session listed in the list of clip pool members;

automatically sending data from a first application running in a first member session to the common clipboard pursuant to a data acquisition operation executed in the first application;

15 automatically sending the data from the common clipboard to a second application running in a second member session pursuant to a data deposition operation executed in the second application, whereby the clipboard data is automatically shared between the first and second applications.



2/5

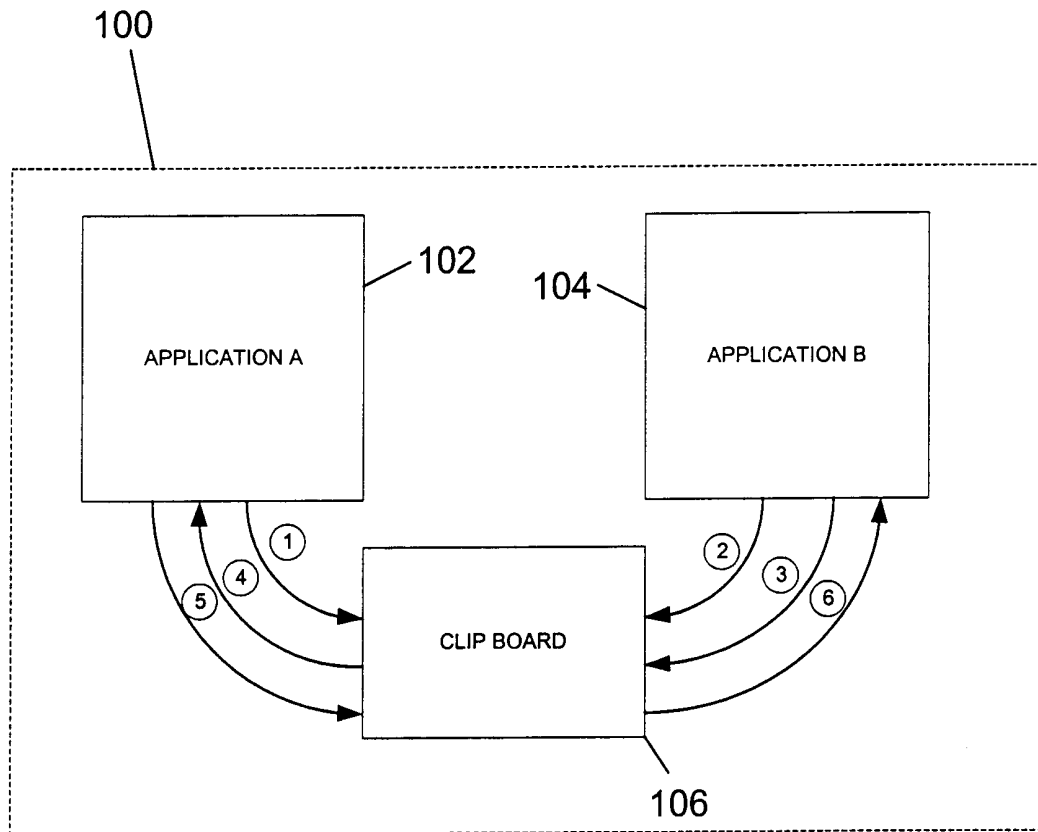
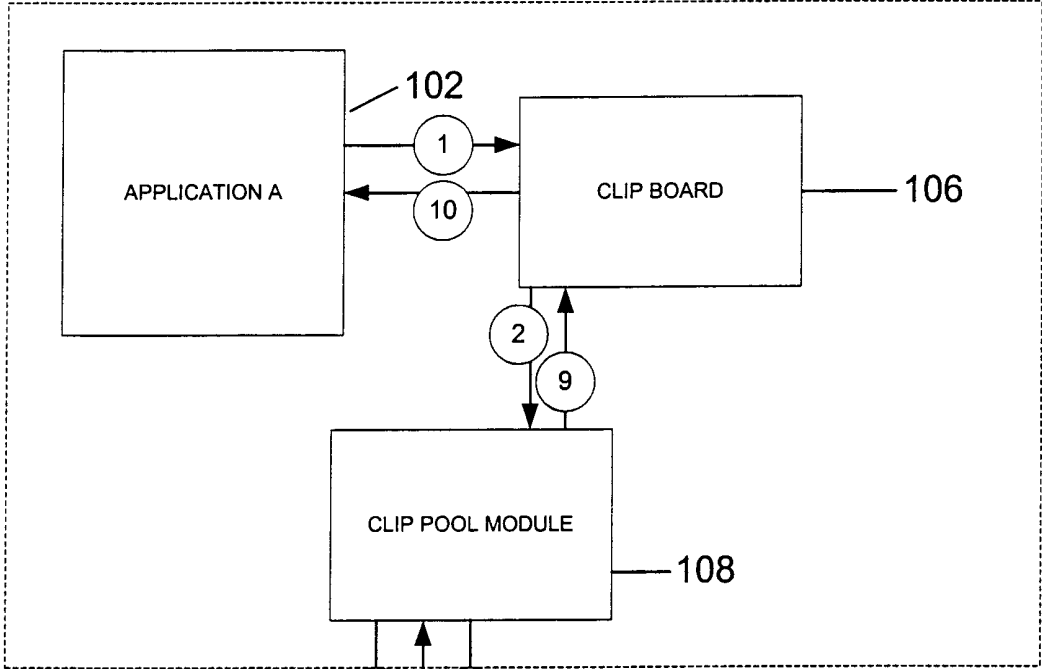


FIG.2 (PRIOR ART)

3/5

MACHINE 1



MACHINE 2

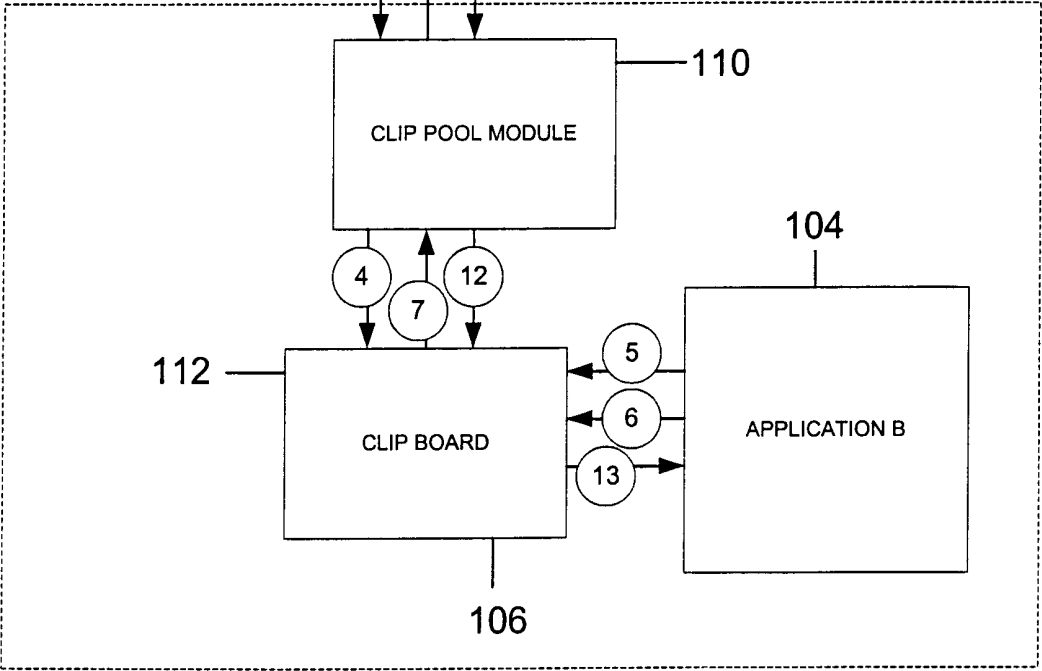


FIG.3

4/5

```

{
    DWORD    length; //the length of the message, in bytes
    DWORD    version; //the version of ClipPool which generated this message
    MessageType type; //the type of the message
    char      source[MAX_COMPUTERNAME_LENGTH+1]; //the name of the
    machine that sent the message

    //
    // Followed by any message-specific information
    //

    BYTE      data[0];
};

```

FIG.4

```

{
    //
    // This is the data buffer for a CLIPBOARD_CHANGE or
    CLIPBOARD_REQUEST message
    //

    DWORD nFormats;
    char data[0]; //the names of each format, separated by null-terminators
};

```

FIG.5

```

{
    //
    // This is the buffer that is returned in response to a CLIPBOARD_REQUEST
    message
    //

    DWORD      dataLength;
    BYTE      data[0];
};

```

FIG.6

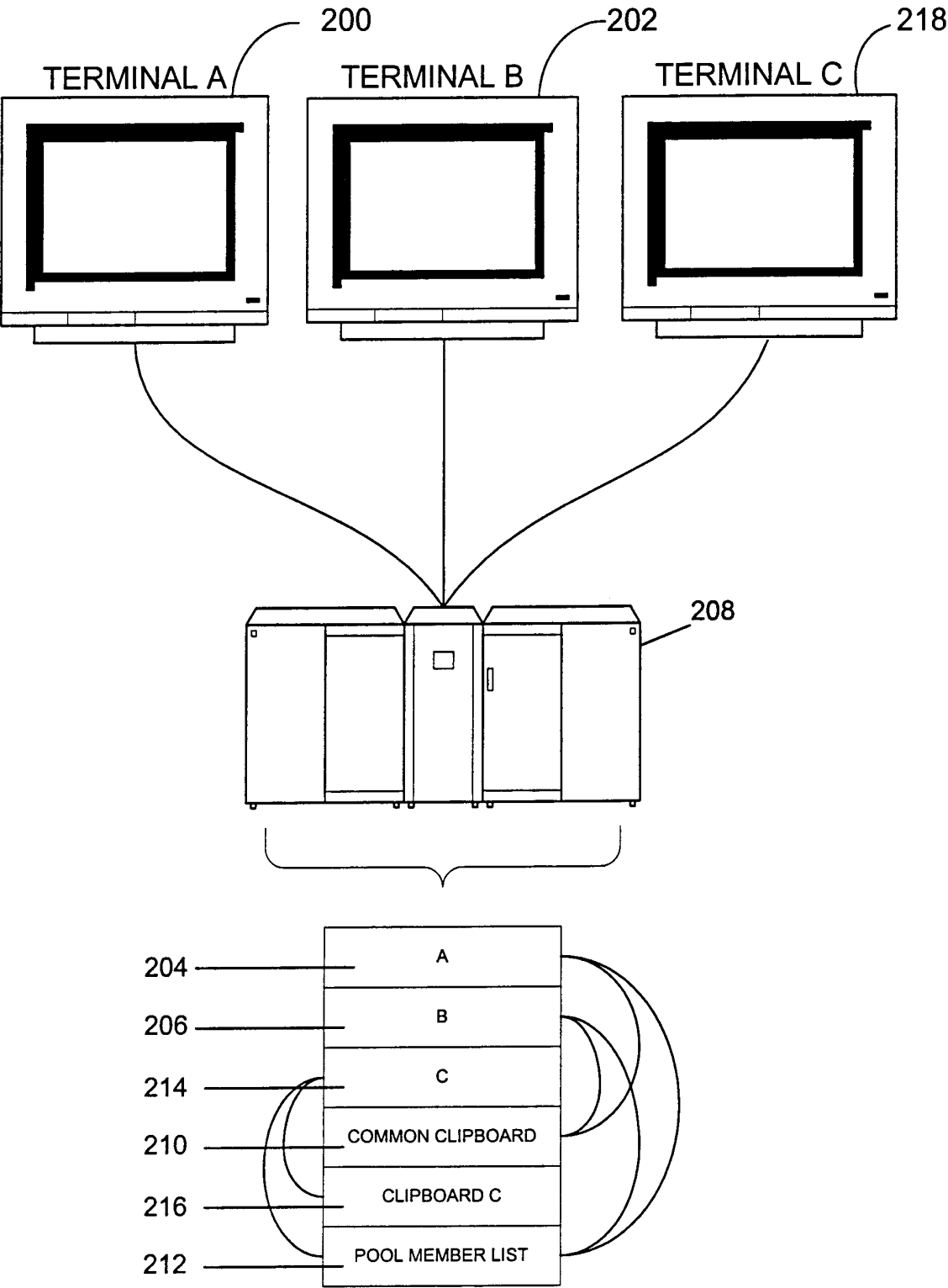


FIG.7

INTERNATIONAL SEARCH REPORT

International Application No
PCT/US 00/13663

A. CLASSIFICATION OF SUBJECT MATTER
IPC 7 G06F9/46

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)
IPC 7 G06F

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

Electronic data base consulted during the international search (name of data base and, where practical, search terms used)

EPO-Internal, INSPEC, IBM-TDB

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category °	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	EP 0 597 575 A (IBM) 18 May 1994 (1994-05-18) the whole document ---	1-10
X	EP 0 744 692 A (IBM) 27 November 1996 (1996-11-27) the whole document ---	1-10
X	EP 0 670 547 A (CANON KK) 6 September 1995 (1995-09-06) the whole document -----	1-10

☐ Further documents are listed in the continuation of box C.

☒ Patent family members are listed in annex.

° Special categories of cited documents :

"A" document defining the general state of the art which is not considered to be of particular relevance
"E" earlier document but published on or after the international filing date
"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)
"O" document referring to an oral disclosure, use, exhibition or other means
"P" document published prior to the international filing date but later than the priority date claimed

"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention
"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone
"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art.
"&" document member of the same patent family

Date of the actual completion of the international search

26 September 2000

Date of mailing of the international search report

05/10/2000

Name and mailing address of the ISA

European Patent Office, P.B. 5818 Patentlaan 2
NL - 2280 HV Rijswijk
Tel. (+31-70) 340-2040, Tx. 31 651 epo nl,
Fax: (+31-70) 340-3016

Authorized officer

Fonderson, A

INTERNATIONAL SEARCH REPORT

Information on patent family members

International Application No

PCT/US 00/13663

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
EP 0597575 A	18-05-1994	GB 2272544 A	18-05-1994
		DE 69327384 D	27-01-2000
		DE 69327384 T	08-06-2000
		JP 2540009 B	02-10-1996
		JP 6202997 A	22-07-1994
EP 0744692 A	27-11-1996	JP 2859559 B	17-02-1999
		JP 8329012 A	13-12-1996
		US 5752254 A	12-05-1998
EP 0670547 A	06-09-1995	JP 7244720 A	19-09-1995
		JP 8044517 A	16-02-1996
		US 5926633 A	20-07-1999