**(12) STANDARD PATENT** (11) Application No. **AU 2006280333 B2**
**(19) AUSTRALIAN PATENT OFFICE**

(54) Title
**Archiving data in a virtual application environment**

(51) International Patent Classification(s)
***G06F 9/455*** (2006.01) ***G06F 15/177*** (2006.01)
***G06F 12/00*** (2006.01)

(21) Application No: **2006280333** (22) Date of Filing: **2006.07.19**

(87) WIPO No: **WO07/021435**

(30) Priority Data

| (31) Number | (32) Date | (33) Country |
|---|---|---|
| **11/205,590** | **2005.08.15** | **US** |

(43) Publication Date: **2007.02.22**
(44) Accepted Journal Date: **2011.03.17**

(71) Applicant(s)
**Microsoft Corporation**

(72) Inventor(s)
**Fries, Robert M.**

(74) Agent / Attorney
**Davies Collison Cave, 1 Nicholson Street, Melbourne, VIC, 3000**

(56) Related Art
**US 2003/0037089**
**US 6748591 B1 (LEWALLEN)**
**US 7127526**

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

(19) World Intellectual Property Organization
International Bureau

(43) International Publication Date
22 February 2007 (22.02.2007)

PCT

(10) International Publication Number
WO 2007/021435 A3

(51) International Patent Classification:
*G06F 9/455* (2006.01) *G06F 15/177* (2006.01)
*G06F 12/00* (2006.01)

(21) International Application Number:
PCT/US2006/028111

(22) International Filing Date: 19 July 2006 (19.07.2006)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:
11/205,590 15 August 2005 (15.08.2005) US

(71) Applicant *(for all designated States except US)*: **MICROSOFT CORPORATION** [US/US]; One Microsoft Way, Redmond, Washington 98052-6399 (US).

(72) Inventor: **FRIES, Robert M.**; One Microsoft Way, Redmond, Washington 98052-6399 (US).

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

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

**Declarations under Rule 4.17:**
— *as to applicant's entitlement to apply for and be granted a patent (Rule 4.17(ii))*
— *as to the applicant's entitlement to claim the priority of the earlier application (Rule 4.17(iii))*

**Published:**
— *with international search report*
— *before the expiration of the time limit for amending the claims and to be republished in the event of receipt of amendments*

(88) Date of publication of the international search report:
30 August 2007

*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.*

(54) Title: ARCHIVING DATA IN A VIRTUAL APPLICATION ENVIRONMENT

(57) Abstract: A computer system suitable for archiving data with a corresponding application program in a virtual application environment is presented. The computer system includes a processor, a memory, and a storage. The computer system also includes an operating system, data to be archived, and an application program that correspond to the data, such that the application program can operate on the data. The computer system also includes a virtual application environment generator. In response to an instruction to archive the data, the virtual application environment generator generates a virtual application environment. The virtual application environment comprises the data, the corresponding application program, and the operating system.

WO 2007/021435 A3

# ARCHIVING DATA IN A VIRTUAL APPLICATION ENVIRONMENT

## Field of Invention

The present invention relates to a computer system for archiving user-created data in a virtual application environment; a computer-readable medium bearing computer-executable instructions which, when executed on a computer system; and a method for archiving user-created data in a virtual application environment.

## Background of Invention

Losing data is an unfortunate reality for many computer users. Data can be lost under a variety of conditions and circumstances. Hardware failure, malicious or accidental deletion are common culprits for data loss. To compensate or protect oneself against loss in these circumstances, many computer users wisely create duplicate, backup copies of important data and/or programs.

Unfortunately, there are also many ways in which data may become lost irrespective of whether a valid backup copy of the data has been made and is available. In particular, over time the programs, the operating systems, and/or the hardware become no longer available. When this occurs, a computer user cannot access critical data, even when an otherwise viable backup copy of that data exists.

For example, data may be stored by a proprietary database and then archived. At some time later, when that data is needed, the proprietary database program (or the operating system that supports that database program, or the hardware upon which that operating system runs) is no longer available or in use. Thus, even though the database has been maintained as a valid archive, the contents of the database are inaccessible.

There are, of course, niche markets that specialize in retrieving data created by and/or stored on archaic computer systems or software. Their techniques vary widely. One technique is to maintain old computer hardware, operating systems, and software.

Another, widely divergent technique involves "cracking open" the data stored by proprietary systems, extracting it using sophisticated software techniques and expertise. In all cases, there is a substantial expense for retrieving data that has been archived but whose corresponding programs, operating systems, or hardware is no longer readily available.

It is generally desirable to overcome or ameliorate one or more of the above described difficulties, or to at least provide a useful alternative.

**Summary of the Invention**

In accordance with one aspect of the present invention, there is provided a computer system for archiving user-created data in a virtual application environment, the computer system comprising a processor, a memory, a storage, an operating system, user-created data to be archived, and a plurality of application programs including at least one application program suitable for operating on the user-created data, and wherein the computer system further comprises:

a virtual application environment generator in response to an instruction to archive the user-created data, the virtual application environment generator is configured to:

identify which of the plurality of application programs is suitable for operating on the user-created data according to an application installation manifest corresponding to the identified program;

generate a virtual application environment comprising the user-created data, the identified application program, and the operating system;

verify whether the virtual application environment is suitable for execution on a virtual machine corresponding to a computer system upon which the operating system and the identified application program can function; and

store the virtual application environment in the storage if the virtual application environment is suitable for execution on the virtual machine.

In accordance with another aspect of the present invention, there is provided a computer-readable medium bearing computer-executable instructions which, when executed on a computer system, carry out a method for archiving user-created data in a virtual

application environment, the method comprising:

receiving a request to archive user-created data;

determining which of a plurality of application programs is suitable for operating on the user-created data according to an application installation manifest corresponding to the identified application program;

generating a virtual application environment comprising user-created data, the determined application program suitable for operating on the user-created data, and an operating system suitable for supporting the execution of the determined application program;

verifying whether the virtual application environment is suitable for execution on a virtual machine corresponding to a computer system upon which the operating system and the determined application program can function; and

storing the virtual application environment in a storage medium.

In accordance with another aspect of the present invention, there is provided a method for archiving user-created data in a virtual application environment, the method comprising:

receiving a request to archive user-created data;

identifying an application program of a plurality of application programs suitable for operating on the user-created data according to the application installation manifest corresponding to the plurality of identified application programs;

generating a virtual application environment comprising the user-created data, the identified application program suitable for operating in the user-created data, and an operating system suitable for supporting the execution of the identified application program;

verifying whether the virtual application environment is suitable for execution on a virtual machine corresponding to a computer system upon which the operating system and the identified application program can function; and

storing the virtual application environment in the storage if the virtual application environment is suitable for execution on the virtual machine.

According to preferred embodiments of the present invention, a computer system suitable for archiving data with a corresponding application program in a virtual application environment is presented. The computer system includes a processor, a memory, and a storage. The computer system also includes an operating system, data to be archived, and an application program that correspond to the data, such that the application program can operate on the data. The computer system also includes a virtual application environment generator. An example of a virtual application environment generator is Microsoft Corporation's Virtual Server Migration Tool. In response to an instruction to archive the data, the virtual application environment generator generates a virtual application environment. The virtual application environment comprises the data, the corresponding application program, and the operating system.

According to additional preferred embodiments of the present invention, a computer-readable medium bearing computer-executable instructions is presented. When executed on a computer system, the computer-executable instructions carry out a method for archiving data in a virtual application environment. The method comprises receiving a request to archive the data. A virtual application environment is then generated. The virtual application environment includes the data, an application program suitable for operating on the data, and an operating system suitable for supporting the execution of the application program. The virtual application environment is the stored in storage.

According to yet further preferred embodiments of the present invention, a method for archiving data in a virtual application environment is presented. The method comprises receiving a request to archive data. A virtual application environment is then generated. The virtual application environment includes the data, an application program suitable for operating on the data, and an operating system suitable for supporting the execution of the application program. The virtual application environment is the verified to determine whether it is suitable for execution on a virtual machine corresponding to a computer system upon which the operating system and the application program can function. If the virtual application environment is suitable for execution on the virtual machine, the virtual application environment is then stored in a storage.

**Brief Description of the Drawings**

Preferred embodiments of the present invention are hereafter described, by way of non-limiting example only, with reference to the accompanying drawings, in which:

FIGURE 1 is a block diagram illustrating an exemplary operating environment suitable for implementing aspects of the present invention;

FIGURE 2 is a block diagram illustrating an exemplary computer system suitable for archiving data into a virtual application environment;

FIGURE 3 is a flow diagram illustrating an exemplary routine for archiving data according to aspects of the present invention; and

FIGURE 4 is a flow diagram illustrating an exemplary routine for retrieving archived data from an virtual application environment.

**Detailed Description of Preferred Embodiments of the Invention**

FIGURE 1 is a block diagram illustrating an exemplary operating environment 100 suitable for implementing aspects of the present invention. The exemplary operating environment 100 includes a computer system 102, shown in FIGURE 1 in logical layers for illustration purposes. In particular, the exemplary computer system 102 includes a hardware layer 104, an operating system layer 106, and an application layer 108.

The hardware layer 104 includes components such as the processor, memory, mass storage (such as hard drives, optical drives, and the like), graphics display subsystems, and the like. The operating system layer 106, as the name suggests, includes the operating system which supports the application layer 108. This layer also typically includes configuration settings, including user preferences and program settings for application programs in the application layer. The application layer 108 includes software programs and/or services such as application programs 110 and 112, as well as the supporting files, user created data (114 and 116), and the like.

While FIGURE 1 illustrates the applications 110 and 112 as single modules, those skilled in the art will appreciate that almost all application programs are typically not monolithic bodies of code. Instead, application programs are typically comprised of various components, including code modules, text modules, data files, resources, configuration files, and the like. Thus, the illustration of application programs 110 and 112 (and others throughout the following discussion) as single component entities should be viewed as a logical construct for illustration purposes, and should not be viewed as limiting upon the present invention.

In addition to application program components, as a product of using most application programs, users typically create data files 114 and 116. In fact, such data files are typically what is most important to the user. More particularly, the application program, if corrupted or deleted from a computer system, can typically be restored from the media with which the user was provided when the application program was purchased. However, the data files created by the application programs, such as data files 114 and 116, are the product of user interactions and creations, and as such, are unique and should be preserved against accidental, malicious, or inadvertent loss.

According to preferred embodiments of the present invention, in order to preserve the data files such that they can be used at a future time when the application program, operating system, and/or hardware is no longer available or useable, the data files are archived in a virtual application environment 122 which can be stored on media such as optical disks, hard drives, or other backup services such as online storage. The virtual application environment 122 includes the operating system and application programs necessary to operate on the co-archived data files.

According to preferred embodiments of the present invention, the virtual application environment 122 is generated such that it can be executed on a virtual machine. As those skilled in the art will appreciate, a virtual machine is an abstraction of computer hardware, implemented in software, that can then be executed on other computer hardware. Generating and using virtual machine technology is known in the art. For example,

Microsoft Corporation's Virtual Server Migration Tool is one example of a product that generates a virtual machine. As shown in FIGURE 1, the virtual machine 124 is generated by a virtual machine generator 126. While not shown, the virtual machine 124 may be stored on any number of backup or storage targets including, but not limited to, magnetic tape, hard disk storage, optical media, and the like.

By generating the virtual operating environment to run on a virtual machine, such as virtual machine 124, it is no longer necessary to maintain the computer hardware 104 in order to access the data generated by obsolete application programs. Thus, after years have passed and the hardware, the operating system, and the application programs are no longer commercially available, the archived data, such as data file 114 or 116, is still accessible. Using the virtual machine technology, implementing a virtual machine 124 corresponding to the hardware from which the virtual application environment 122 was generated, the data is accessible on virtually any computer hardware running any operating system.

In order to ensure that the virtual application environment 122 can later be executed, in one embodiment, the virtual operating environment includes information identifying the virtual machine platform, i.e., the corresponding computer hardware from which the virtual application environment was generated.

The virtual application environment 122 is created using a virtual operating environment generator 118 that converts the data files, the application program or programs, and the operating system into the virtual operating environment 122. In particular, virtual application environment generator 118 determines which components (especially with regard to a particular application program) need to be included in the virtual application environment 122 such that the application program corresponding to the subject matter data file can operate.

According to one embodiment, the virtual application environment generator 118 includes all application programs and the operating system on the computer system 102 into the

virtual application environment 122, thereby ensuring that all components of the application program corresponding to the subject matter data file are included. Alternatively, the virtual operating environment generator 118 uses an application manifest 120 that identifies the components of an application program that should be included in the virtual application environment 122. Of course, other information may also be used to determine the components of an application program, such as installation manifests, operating system registry information, and the like.

In at least one embodiment of the present invention, the virtual application environment 122 is generated such that it operates as a bootable device. In other words, the virtual application environment 122 can function as the boot device of actual computer hardware, or of a virtual machine.

FIGURE 2 is a block diagram illustrating an exemplary computer system 200 suitable for archiving data into a virtual application environment 122. In particular, the computer system 200 includes computer hardware 104, an operating system 106, one or more application programs 110 and 112, as well as corresponding data files 114 and 116. As shown in FIGURE 2, the computer hardware includes a processor 202, a memory 204, and a storage 206. Of course, as those skilled in the art will appreciate, these hardware components are illustrative of the many components that make up the computer hardware, and should not be viewed as limiting upon the present invention.

The exemplary computer system 200 also includes a virtual application environment generator 118. As indicated above, the virtual application environment generator 118 combines the data file, such as data file 114, with its corresponding application program and the operating system operating on the computer hardware 104, into a virtual application environment 122 (FIGURE 1). However, it should be appreciated that while the virtual application environment generator 118 is illustrated as residing upon the exemplary computer system 200, in an alternative embodiment, the virtual application environment generator is on an external computer system, and configured to operated remotely on the exemplary computer system 202.

FIGURE 3 is a flow diagram illustrating an exemplary routine 300 for archiving data in a virtual application environment 122 according to aspects of the present invention. Beginning at block 302, a request to archive a data file is received. At block 304, the application program (or programs) associated with the data file, its components and associated files, are identified. As discussed above, the application program's components may be identified using an application manifest 120, using computer system settings such as registry entries, using installation manifests, and the like.

In an alternative embodiment, a request to archive a volume may be received. As those skilled in the art will appreciate, a volume may include any number of data files, some of which created by disparate application programs. Thus, similar to the logic

described above, an exemplary routine would identify, for each data file in the volume to be archived, the corresponding application program (or programs) for inclusion in the virtual application environment 122.

At block 306, the virtual application environment 122, including the identified application program components, the data file, and an operating system capable of operating the application program is generated. The virtual application environment 122 may also include information identifying the virtual machine or actual computer hardware upon which the virtual application environment can execute.

At decision block 308, the virtual application environment 122 is optionally validated to determine whether it can be executed on a virtual machine corresponding to the computer hardware from which the components of the virtual application environment were generated. If the virtual application environment 122 proves invalid, at block 310 an error is reported, and thereafter the routine 300 terminates.

If the virtual application environment 122 is properly validated, at block 312, the virtual application environment is stored. Thereafter, the routine terminates.

It should be appreciated that while the above description of routine 300 treats the data file as a single object, in fact the data to be archived may comprise a plurality of files and components, and the present invention should not be construed as limited to archiving a single data file in the virtual application environment 122.

FIGURE 4 is a flow diagram illustrating an exemplary routine 400 for retrieving archived data from a virtual application environment 122. Beginning at block 402, a request to restore data from a virtual application environment 122 is received. At block 404, the virtual application environment 122 is obtained.

At block 406, information regarding the virtual machine 124 corresponding to and necessary to support the virtual application environment 122 is obtained. At block 408, the virtual machine 124 corresponding to virtual application environment 122 is executed (if it is not already executing). Of course, as discussed earlier, the virtual application environment 122 may be executed on a virtual machine 124, or alternatively, on computer hardware (not shown) corresponding to the original computer hardware on which the virtual application environment 122 was generated.

Executing the virtual application environment 122 may involve booting the virtual application environment upon the virtual machine. In such cased, the virtual application environment 122 should be configured to operate as a bootable device.

Once the virtual application environment 122 is executing, at block 410, the application program corresponding to the data file which is to be retrieved is executed. At block 412, the data is extracted using the corresponding application program, either in an automated manner, or at the direction of a computer user. Thereafter, the routine 400 terminates.

While the preferred embodiment of the invention has been illustrated and described, it will be appreciated that various changes can be made therein without departing from the spirit and scope of the invention.

Throughout this specification and the claims which follow, unless the context requires otherwise, the word "comprise", and variations such as "comprises" and "comprising", will be understood to imply the inclusion of a stated integer or step or group of integers or steps but not the exclusion of any other integer or step or group of integers or steps.

The reference in this specification to any prior publication (or information derived from it), or to any matter which is known, is not, and should not be taken as an acknowledgment or admission or any form of suggestion that that prior publication (or information derived from it) or known matter forms part of the common general knowledge in the field of endeavour to which this specification relates.

THE CLAIMS DEFINING THE INVENTION ARE AS FOLLOWS:

1. A computer system for archiving user-created data in a virtual application environment, the computer system comprising a processor, a memory, a storage, an operating system, user-created data to be archived, and a plurality of application programs including at least one application program suitable for operating on the user-created data, and wherein the computer system further comprises:

a virtual application environment generator in response to an instruction to archive the user-created data, the virtual application environment generator is configured to:

identify which of the plurality of application programs is suitable for operating on the user-created data according to an application installation manifest corresponding to the identified program;

generate a virtual application environment comprising the user-created data, the identified application program, and the operating system;

verify whether the virtual application environment is suitable for execution on a virtual machine corresponding to a computer system upon which the operating system and the identified application program can function; and

store the virtual application environment in the storage if the virtual application environment is suitable for execution on the virtual machine.

2. The computer system of claim 1, wherein the virtual application environment is generated to operate on a virtual machine corresponding to the computer system.

3. The computer system of claim 2, wherein the virtual application environment is further generated to operate as a bootable device on a virtual machine corresponding to the computer system.

4. The computer system of claim 2, wherein the virtual application environment includes in formation that identifies the virtual machine upon which the virtual application environment can operate.

5. The computer system of claim 1, wherein the virtual application environment is generated to include all of the application programs from the computer system, including the identified application program suitable for operating on the user-created data.

6. The computer system of claim 1, wherein the virtual application environment generator generates the virtual application environment such that only the corresponding application program of the plurality of application programs is included with the operating system and user-created data.

7. The computer system of claim 1, wherein the identified application program suitable for operating on the user-created data comprises a plurality of components distributed on the computer system, and where the virtual application environment generator identifies the components of the identified application program and includes them as the identified application program in the virtual application environment.

8. A computer-readable medium bearing computer-executable instructions which, when executed on a computer system, carry out a method for archiving user-created data in a virtual application environment, the method comprising:

receiving a request to archive user-created data;

determining which of a plurality of application programs is suitable for operating on the user-created data according to an application installation manifest corresponding to the identified application program;

generating a virtual application environment comprising user-created data, the determined application program suitable for operating on the user-created data, and an operating system suitable for supporting the execution of the determined application program;

verifying whether the virtual application environment is suitable for execution on a virtual machine corresponding to a computer system upon which the operating system and the determined application program can function; and

storing the virtual application environment in a storage medium.

9. The computer-readable medium of claim 8, wherein the virtual application environment is generated for execution on a virtual machine corresponding to the computer system.

10. The computer-readable medium of claim 9, wherein the virtual application environment is further generated to operate as a bootable device when executed on a virtual machine corresponding to the computer system.

11. The computer-readable medium of claim 10, wherein the virtual application environment is further generated to include information for identifying the virtual machine upon which the virtual application environment may be executed.

12. The computer-readable medium of claim 8, wherein the virtual application environment is generated to include a plurality of application programs installed on the computer system including the determined application program suitable for operating on the user-created data.

13. A method for archiving user-created data in a virtual application environment, the method comprising:

receiving a request to archive user-created data;

identifying an application program of a plurality of application programs suitable for operating on the user-created data according to the application installation manifest corresponding to the plurality of identified application programs;

generating a virtual application environment comprising the user-created data, the identified application program suitable for operating in the user-created data, and an operating system suitable for supporting the execution of the identified application program;

verifying whether the virtual application environment is suitable for execution on a virtual machine corresponding to a computer system upon which the operating system and the identified application program can function; and

storing the virtual application environment in the storage if the virtual application

environment is suitable for execution on the virtual machine.

14. The method of claim 13, wherein the virtual application environment is further generated to operate as a bootable device when executed on a virtual machine corresponding to the computer system.

15. The method of claim 14, wherein identifying an application program of a plurality of application programs is suitable for operating on the user-created data comprises identifying an application program of a plurality of application programs is suitable for operating on the user-created data according to operating system registry information.

16. The method of claim 13 wherein the user-created data to be archived comprises a plurality of user-created data files corresponding to at least two distinct application programs, and wherein the method further comprises for each user-created data file in the user-created data to be archived, identifying an application program suitable for operating on the user-created data file, and wherein generating the virtual application environment further comprises generating the virtual application environment to include the identified application programs suitable for operating on each user-created data file in the user-created data to be archived.

17. A computer system for archiving user-created data in a virtual application environment substantially as hereinbefore described, with reference to the accompanying drawings.

18. A computer-readable medium bearing computer-executable instructions which, when executed on a computer system substantially as hereinbefore described, with reference to the accompanying drawings.

19. A method for archiving user-created data in a virtual application environment substantially as hereinbefore described, with reference to the accompanying drawings.

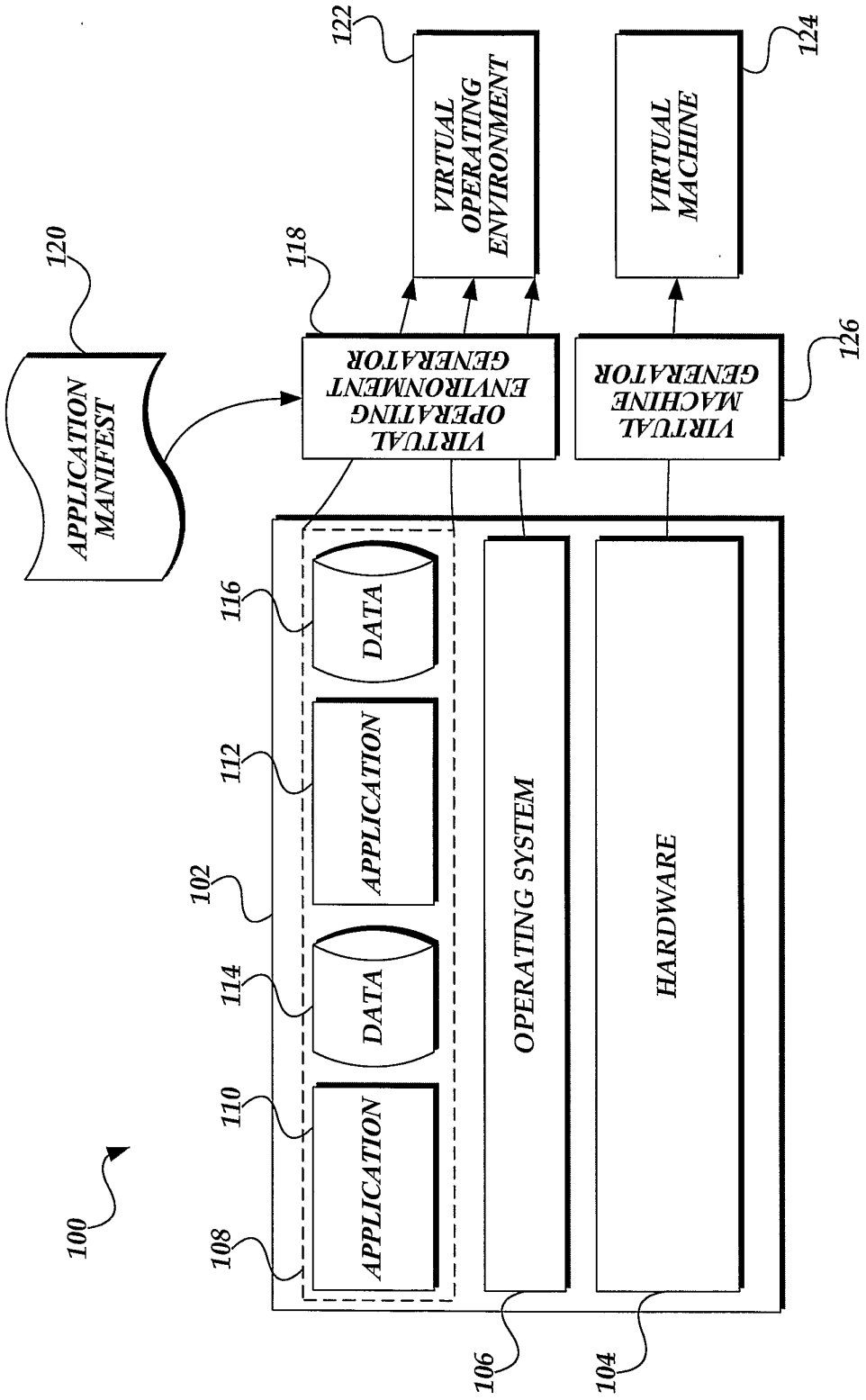
100
APPLICATION
110
DATA
114
APPLICATION
112
DATA
116
102
108
OPERATING SYSTEM
106
HARDWARE
104
APPLICATION MANIFEST
120
VIRTUAL OPERATING ENVIRONMENT GENERATOR
118
VIRTUAL MACHINE GENERATOR
126
VIRTUAL OPERATING ENVIRONMENT
122
VIRTUAL MACHINE
124

Fig.1.

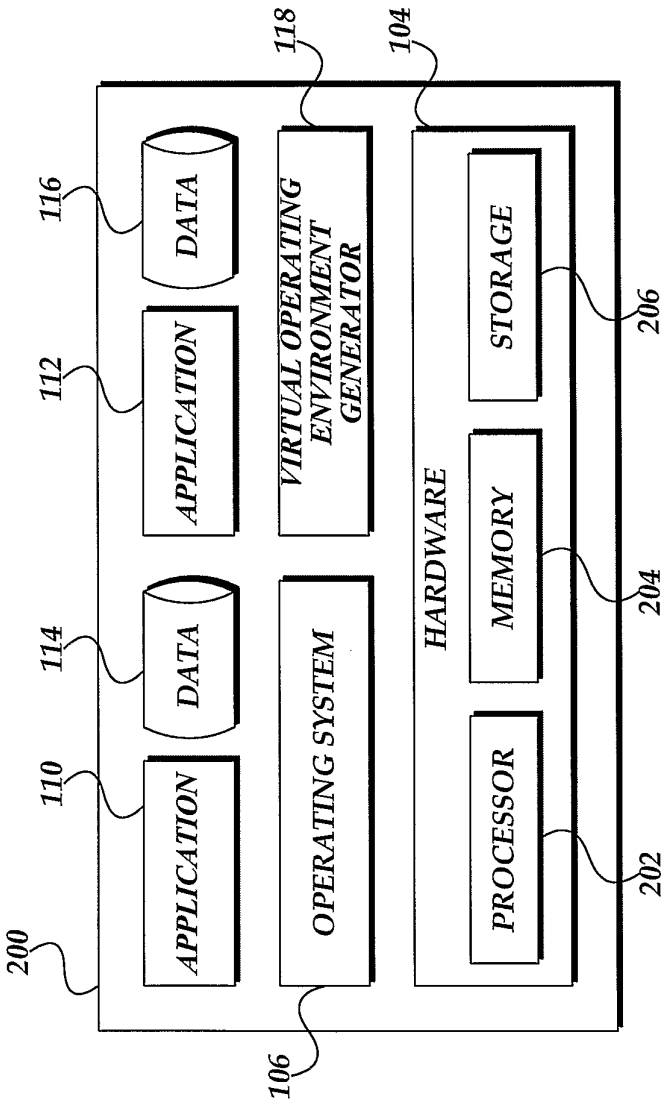
200
APPLICATION
110
DATA
114
APPLICATION
112
DATA
116
OPERATING SYSTEM
106
VIRTUAL OPERATING ENVIRONMENT GENERATOR
118
HARDWARE
104
PROCESSOR
202
MEMORY
204
STORAGE
206

*Fig.2.*

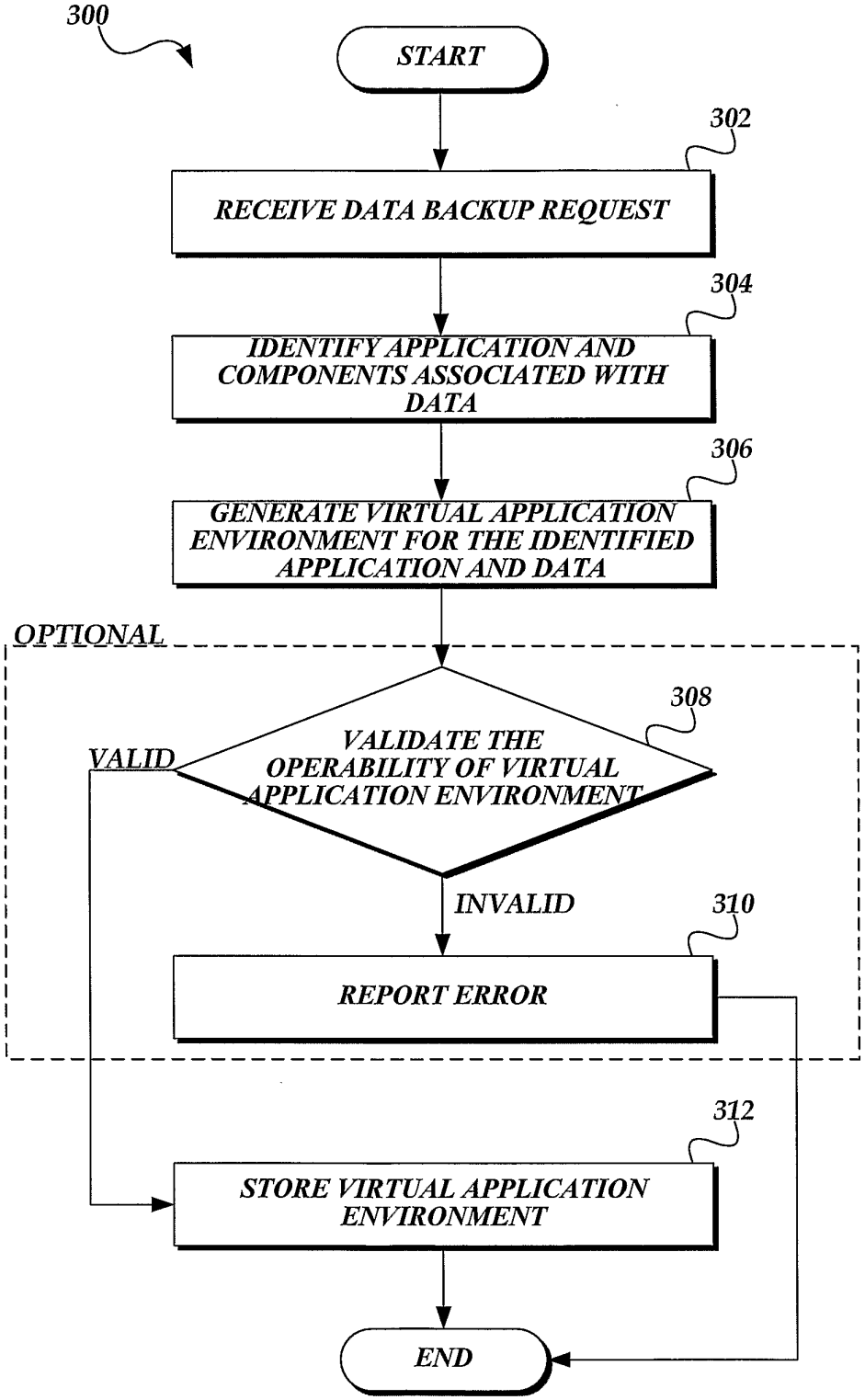
300
START
302
RECEIVE DATA BACKUP REQUEST
304
IDENTIFY APPLICATION AND COMPONENTS ASSOCIATED WITH DATA
306
GENERATE VIRTUAL APPLICATION ENVIRONMENT FOR THE IDENTIFIED APPLICATION AND DATA
OPTIONAL
308
VALIDATE THE OPERABILITY OF VIRTUAL APPLICATION ENVIRONMENT
VALID
INVALID
310
REPORT ERROR
312
STORE VIRTUAL APPLICATION ENVIRONMENT
END

*Fig.3.*

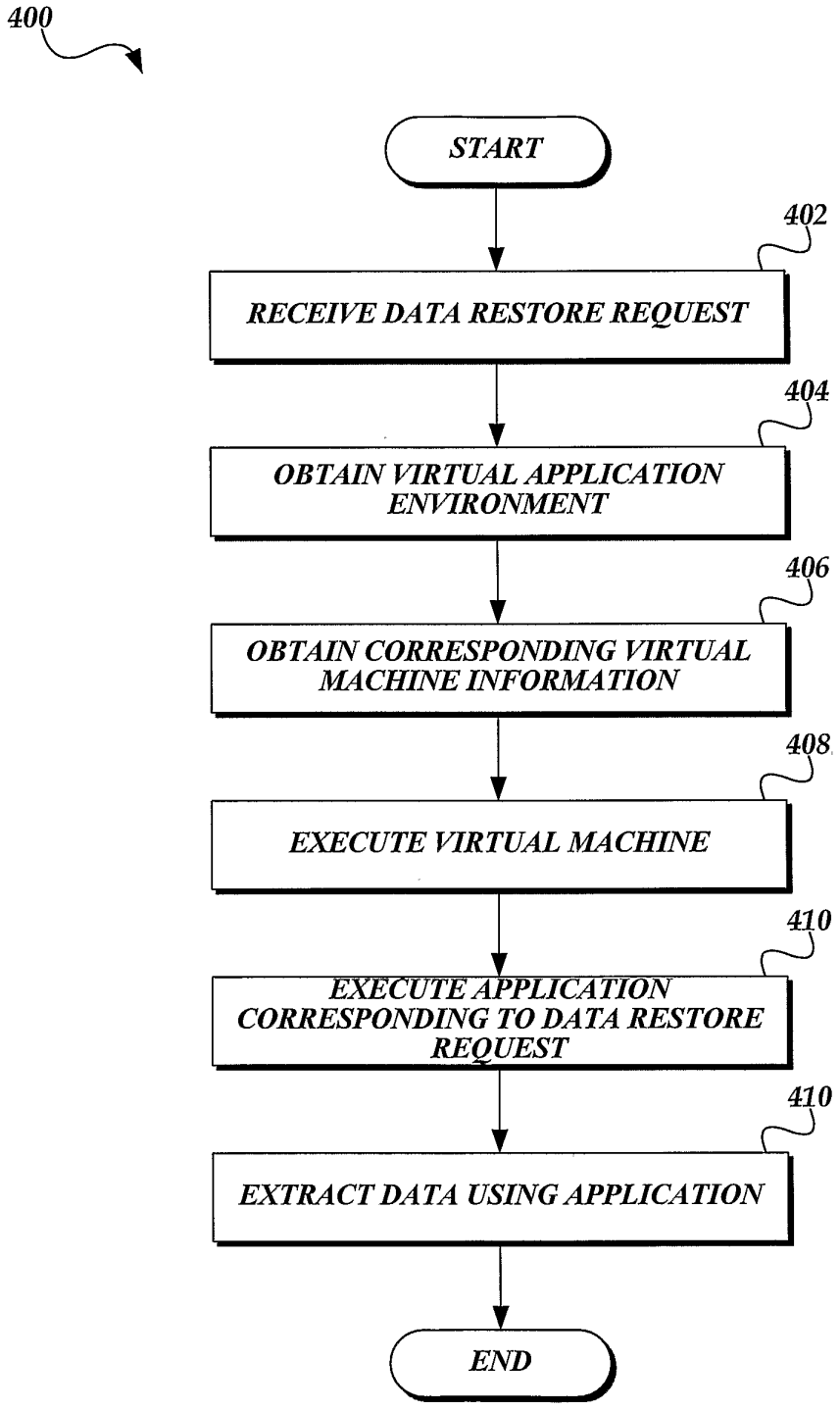
400
START
402
RECEIVE DATA RESTORE REQUEST
404
OBTAIN VIRTUAL APPLICATION ENVIRONMENT
406
OBTAIN CORRESPONDING VIRTUAL MACHINE INFORMATION
408
EXECUTE VIRTUAL MACHINE
410
EXECUTE APPLICATION CORRESPONDING TO DATA RESTORE REQUEST
410
EXTRACT DATA USING APPLICATION
END

*Fig.4.*