(19) Europäisches Patentamt
European Patent Office
Office européen des brevets

(11) EP 2 761 462 B1

(12) **EUROPEAN PATENT SPECIFICATION**

(45) Date of publication and mention of the grant of the patent:
03.01.2018 Bulletin 2018/01

(21) Application number: **12834947.9**

(22) Date of filing: **07.09.2012**

(51) Int Cl.:
G06F 11/34 *(2006.01)* ***G06F 11/30*** *(2006.01)*
G06F 9/445 *(2018.01)*

(86) International application number:
PCT/CN2012/081150

(87) International publication number:
WO 2013/044725 (04.04.2013 Gazette 2013/14)

(54) **METHOD AND DEVICE FOR OBTAINING USING-FREQUENCY OF APPLICATION PROGRAM**

VERFAHREN UND VORRICHTUNG ZUR ERMITTLUNG DER VERWENDUNGSHÄUFIGKEIT EINES ANWENDUNGSPROGRAMMS

PROCÉDÉ ET DISPOSITIF D'OBTENTION DE FRÉQUENCE D'UTILISATION D'UN PROGRAMME D'APPLICATION

(84) Designated Contracting States:
AL AT BE BG CH CY CZ DE DK EE ES FI FR GB GR HR HU IE IS IT LI LT LU LV MC MK MT NL NO PL PT RO RS SE SI SK SM TR

(30) Priority: **27.09.2011 CN 201110296513**

(43) Date of publication of application:
06.08.2014 Bulletin 2014/32

(73) Proprietor: **Tencent Technology (Shenzhen) Company Limited**
Shenzhen, Guangdong 518057 (CN)

(72) Inventors:
- **WANG, Baojian**
  Shenzhen
  Guangdong 518057 (CN)
- **HU, Zhongxing**
  Shenzhen
  Guangdong 518057 (CN)
- **ZHANG, Yinghao**
  Shenzhen
  Guangdong 518057 (CN)

(74) Representative: **Reichert & Lindner Partnerschaft Patentanwälte**
Bismarckplatz 8
93047 Regensburg (DE)

(56) References cited:
CN-A- 1 852 538 **CN-A- 101 563 672**
US-A1- 2002 026 631 **US-A1- 2002 122 076**
US-A1- 2010 057 905 **US-B1- 6 324 546**
US-B1- 7 925 635

Printed by Jouve, 75001 PARIS (FR)

Description

Technical Field

[0001] The present invention relates to the computer technology, and more particularly to a method and a device for obtaining a using-frequency of an application program.

Background of the Invention

[0002] With the development of the computer technology, users may install various application programs in operation systems of computers, for satisfying various using needs of the users. For the various application programs, it is important to obtaining using-frequencies of the various application programs, such as to optimize the operation systems of the computers according to the using-frequencies of the various application programs.

[0003] Currently, install/uninstall list the operation systems of the Windows comes with, can judge the using-frequencies of the application programs, and the judgment thereof is performed according to easy algorithms. Thus the result thereof is not accurate, and it is difficult to obtain the real using-frequencies of the application programs, thus it cannot further provide a system optimization scheme for the users.

[0004] US 2010/057905 A1, US 6,324,546 B1 and US 7,925,635 B1 disclose methods and devices for obtaining a using-frequency of a specified application program. US 2010/057905 A1 discloses the steps of obtaining an installing route of the specified application program according to a system-registering information list; recording a last start-up time and the installing route of the specified application program; determining the using-frequency of the specified application program.

SUMMARY OF THE INVENTION

[0005] The present invention provides a method according to appended independent claim 1 and a device according to appended independent claim 4 for obtaining a using frequency of an application program, which can accurately count the using-frequency of the application program installed in the Operation System of the computer, to really report the status of the user using the application program, for provide a system optimization scheme for the user. The method for obtaining a using-frequency of a specified application program, includes:

obtaining an installing route of the specified application program according to a system-registering information list;
calling an application programming interface (API), to monitor operation of the specified application program;
recording a last start-up time and the installing route of the specified application program;
obtaining the last start-up time of the application program based on the installing route as an index;
comparing the last start-up time of the application program and a current time, to determine the using-frequency of the application program.

[0006] Correspondingly, the device for obtaining a using-frequency of a specified application program, includes:

a route-obtaining module, configured for obtaining an installing route of the specified application program according to a system-registering information list;

a program-monitoring module, configured for calling an application programming interface (API) to monitor operation of the specified application program;

an information-recording module, configured for recording a last start-up time and the installing route of the specified application program;

a time-obtaining module, configured for obtaining the last start-up time of the specified application program from the information-recording module, based on the installing route as an index;

a frequency-calculating module, configured for comparing the last start-up time and a current time, to determine the using-frequency of the application program.

[0007] The present invention can intercept the process start functions of the Operation System, to easily and efficiently obtain the using-frequency of the specified application program installed in the Operation System of the computer. Therefore, the present invention may easily manage the application program installed in the computer, and provide the

base for optimizing the Operation System of the computer.

Brief Description of the Drawing

[0008] To explain the technical solution provided by the embodiments of the present invention more clearly, the drawings used by the descriptions of the embodiments will be introduced briefly as follows. Obviously, the drawings described below are a plurality of embodiments of the present invention; those skilled in the art may further obtain other drawings according to these drawings on the premise of having no creative works.

Figure 1 is a flow chart of a method for obtaining a using-frequency of a specified application program in accordance with a first exemplary embodiment of the present invention;

Figure 2 is a schematic view for obtaining an installing route of the specified application program in the method for obtaining the using-frequency of the specified application program in accordance with the first exemplary embodiment of the present invention;

Figure 3 is another schematic view for obtaining an installing route of the specified application program in the method for obtaining the using-frequency of the specified application program in accordance with the first exemplary embodiment of the present invention;

Figure 4 is other schematic view for obtaining an installing route of the specified application program in the method for obtaining the using-frequency of the specified application program in accordance with the first exemplary embodiment of the present invention;

Figure 5 is a flow chart of the method for obtaining the using-frequency of the specified application program in accordance with a second exemplary embodiment of the present invention;

Figure 6 is a flow chart of the method for obtaining the using-frequency of the specified application program in accordance with a third exemplary embodiment of the present invention;

Figure 7 is a schematic view of a device for obtaining a using-frequency of a specified application program in accordance with a first exemplary embodiment of the present invention;

Figure 8 is a schematic view of the device for obtaining the using-frequency of the specified application program in accordance with a second exemplary embodiment of the present invention;

Figure 9 is a schematic view for the device for obtaining the using-frequency of the specified application program in accordance with a third exemplary embodiment of the present invention;

Figure 10 is a schematic view for the device for obtaining the using-frequency of the specified application program in accordance with a fourth exemplary embodiment of the present invention.

Embodiments of the Invention

[0009] The present invention will now be described more specifically with reference to the following embodiments. It is to be noted that the following descriptions of preferred embodiments of this invention are presented herein for purpose of illustration and description only. It is not intended to be exhaustive or to be limited to the precise form disclosed.

[0010] The present invention provides a technology for obtaining a using-frequency of a specified application program, which can accurately count the using-frequency of the specified application program installed in the Operation System of the computer, for really reporting the status of the user using the specified application program, to provide a system optimization scheme for the user.

[0011] To clearly describe the exemplary embodiments of the present invention, the following firstly introduce technical terms involved in the exemplary embodiments of the present invention.

[0012] API (Application Programming Interface): are some pre-defined functions, for providing a capability of accessing a group of routines for application programs and developers based on certain software or hardware, without accessing source codes or understanding particulars of inner operating mechanism.

[0013] Process: is the basis of the operation system; is an application program which is processing; is a program instance which is operating in the computer; is able to be distributed to the processor (the CPU) and is an entity processed

by the processor; is an action unit, which is performing to display in a single sequence, and described by a current status and a group of related system resources.

[0014] Hook: is a system mechanism provided by the Windows for replacing the interrupt under the DOS. After performing the hook for a specific system event, if a hooked event occurs, a program configured for hooking the event is informed by the system, thus the application program can respond to the event in the first period of time.

[0015] Application Program: is also called as "Application Software", and is software developed for a certain specific purpose. The application program may be a specific program, such as a photo browser. Alternatively, the application program also may be a set of a group of programs which are closely associated in function and cooperate with each other, such as the Office software. Alternatively, the application program also may be a large software system consisted of numerous independent programs, such as a database management system.

[0016] FIG. 1 is a flow chart of a method for obtaining a using-frequency of a specified application program in accordance with a first exemplary embodiment of the present invention.

[0017] The method for obtaining the using-frequency of the specified application program provided in the exemplary embodiment of the present invention, comprises:

Step 100: obtaining an installing route of the specified application program, according to a system-registering information list;

Step 101: calling an API, to monitor the operation of the specified application program;

Step 102: recording a last start-up time and the installing route of the specified application program;

Step 103: obtaining the last start-up time of the specified application program and a current time, based on the installing route as an index;

Step 104: comparing the last start-up time and the current time, to determine the using-frequency of the specified application program.

[0018] The exemplary embodiment of the present invention uses the system-registering information list, to accurately obtain the installing route of the specified application program, and monitor start-up situations of the specified application program based on the installing route, such as to judge the using-frequency of the specified application program according to the recorded last start-up time. The method for obtaining the using-frequency of the specified application program provided in the exemplary embodiment of the present invention, is real and effective, impersonal and accurate, such that the user can distinctly know about the using status of the application program installed in the operation system of the computer. Furthermore, the user may further optimize the operation system of the computer according to the obtained using-frequency of the application program.

[0019] The step of obtaining the installing route of the specified application program according to the system-registering information list, may be performed by following modes:

A first mode: obtaining the installing route of the specified application program, according to an entry of a key being InstallLocation in the system-registering information list.

[0020] In detail, a program is called to obtain the installing route of the specified application program from the entry of the key being InstallLocation, which is under SOFTWARE\\Microsoft\\Windows\\Current Version\\Uninstall of the system-registering information list.

[0021] For example, FIG. 2 is a schematic view for showing all entries of the application program of "Autodesk Express Viewer" under SOFTWARE\\Microsoft\\Windows\\Current Version\\Uninstall of the system-registering information list. In FIG. 2, from the entry of the key being "InstallLocation", it can be seen that the installing route is "C:\Program Files\Autodesk\Autodesk Express Viewer".

[0022] A second mode: obtaining the installing route of the specified application program, according to an entry of a key being UnstallString in the system-registering information list.

[0023] In detail, some application programs do not have the entry of the key being "InstallLocation" in the system-registering information list. Therefore, the second mode is performed to call a program to read the entry of the key being UnstallString under SOFTWARE\\Microsoft\\Windows\\Current Version\\Uninstall of the system-registering information list, and then remove exe file name, to obtain the installing route of the application program.

[0024] For example, FIG. 3 is a schematic view for showing all entries of the specified application program of "Adobe Flash Player Activex" under "SOFTWARE\\Microsoft\\Windows\\Current Version\\Uninstall" of the system-registering information list. In FIG. 3, the entry of the key being "InstallLocation" does not existed. Thus, the entry of the key being

"UnstallString" is found, which is presented as "C:\\WINDOWS\system32\Macromed\Flash\ FlashUtil10w_ActiveX.exe-matntain activex". Then it removes the exe file name, to obtain the installing route of "C:\WINDOWS\system32\Macromed\Flash\".

[0025] A third mode: fuzzily matching names of shortcuts of application programs with the name of the specified application program recorded in the system-registering information list, and if successfully matching a name of a shortcut of an application program with the name of the specified application program recorded in the system-registering information list, obtaining a directing route of the shortcut as the installing route of the specified application program.

[0026] In detail, some application programs not only do not have the entry of the key being "InstallLocation", but also do not have the entry of the key being "UnstallString" in the system-registering information list. Therefore, the present invention may further enumerate shortcuts of application programs, which may be in user desktop, start menu, or start shortcut, etc, and then fuzzily matching names of the shortcuts of the application programs with the name of the specified application program in the system-registering information list. If successfully matching, the present invention may regard the directing route of the shortcut as the installing route of the specified application program.

[0027] For example, an operation system of a computer has installed application programs, such as "Tencent TM2009", "CHATM2010", "Windstorm Image Sound", etc, and shortcuts of the application programs are located in the user desktop, the start menu, or the start shortcut, etc., respectively. The exemplary embodiment will enumerate the shortcuts, then matching the names of the shortcuts with the name of the application program in the system-registering information. If a name of a shortcut is "CHATM2010" and it successfully matches with "CHATM trademark-querying software" in the system-registering information list when they fuzzily matches, the directing route of the shortcut is obtained, which is as shown in FIG. 4, and the directing route of the shortcut is "Target (T) C:\Rabbit\chatm\exe\Quety_net.exe". Then the directing route thereof is regarded as the installing route of the application program of "CHATM2010".

[0028] A fourth mode: fuzzily matching names of application programs of each project folder of each disk, with the name of the specified application program recorded in the system-registering information list, and if a name of an application program of a project folder of a disk is successfully matched with the name of the specified application program in the system-registering information list, regarding a route of the application program of the project folder of the disk as the installing route of the specified application program.

[0029] In detail, for some application programs, all of the above three modes cannot obtain the installing route thereof. Thus the fourth mode is used, to enumerate application programs of each folder under the project directory of Program Files of each disk, and match names of the application programs of each fold under the project directory of Program Files of each disk with the name in the system-registering information list. If a name of an application program of a fold under the project directory of Program Files of a disk is successfully matched with the name in the system-registering information list, the route of the application program of the project folder is regarded as the installing route of the specified application program.

[0030] For example, a folder named as "Movie Maker" exists in the folder of Program Files of C disk. If the name of "Movie Maker" is successfully matched with the name of the specified application program in the system-registering information list, the route of the application program of "Movie Maker" is "C:\Program Files\Movie Maker" and is regarded as the installing route of the specified application program.

[0031] For the software located in the directory of system, such as Application Verifier, Flash Palyer, it needs to use the exepath to match, and the method is similar with the above, which is not described herein.

[0032] It should be noted that, the invention obtains the installing route in sequence from the first mode to the fourth mode. The present invention use the above four modes to obtain the installing route of the application program, thus it can furthest accurately obtain the installing route of the specified application program for analyze the using-frequency of the specified application program.

[0033] When or after obtaining the installing route of the specified application program, the exemplary embodiment of the present invention will call an API to monitor the operation of the specified application program.

[0034] Refer to FIG. 5, which is a flow chart of the method for obtaining the using-frequency of the specified application program in accordance with a second exemplary embodiment of the present invention.

[0035] The second exemplary embodiment will describe the flow of calling the API to monitor the operation of the specified application program in detail, which comprises:

Step 200: calling the API, and obtaining command-line arguments of application programs which are running.

[0036] In detail, the monitor of the API Create Process is firstly open. When the Create Process is called, the command-line arguments in the Create Process are obtained.

[0037] Step 201: inputting the command-line arguments of the application programs into an independent processing thread.

[0038] In detail, for not influencing the performance of calling the Create Process, the command-line arguments are inputted into an independent processing thread in the above Step 201.

[0039] Step 202: monitoring the operation of the application programs corresponding to the command-line arguments respectively.

[0040] The second exemplary embodiment provides the method for monitoring the various application programs, and the above monitoring mode can accurately obtain the status of the operation of the various application programs, to obtain the start-up time of the various application programs.

[0041] Refer to FIG. 6, which is a flow chart of the method for obtaining the using-frequency of the specified application program in accordance with a third exemplary embodiment of the present invention.

[0042] The third exemplary embodiment will describe of the process of how to obtaining the last start-up time of the application program and obtaining the using frequency thereof, which is described in following:

Step 300: obtaining the last start-up time of the specified application program from the processing thread and recording it.

[0043] In detail, the present invention monitors the various application programs in the processing thread, thus it can obtain the start-up time of any one specific application program. The exemplary embodiment only needs to record the last start-up time of the application programs. Tools for recording the start-up time thereof may be notepad or SQlite. It should be noted that, the SQlite is a light-duty database, and complies with the ACID-associated database managing system. The SQlite is embedded, and occupies little resource, and can support the main-trend Operation System, such as Windows or Linux or Unix, etc.

[0044] Step 301: performing a standardization process for the installing route.

[0045] In detail, installing routes obtained in the above modes of the present invention may be different in form. Thus the exemplary embodiment needs to perform the standardization process for the installing routes, thus they are quickly looked up and compared. The standardization process for the installing routes, comprises: unifying characters of the installing routes as lowercase characters; unifying names of the installing routes to comprise short file names, long file names, environment variables; standardizing the installing routes according to common-name rules. For example, the installing routes of "%system%" and "C:\windows\system32" should be uniformly standardized as "c:\windows\systems32\".

[0046] Step 302: recording the standardized installing route. In detail, the present invention may use the tools of Notepad or SQlite, to build an application-program information list, for recording last start-up time and installing routes of the various application programs, which are shown in table one:

Application Program	Installing Route	Last Start-Up Time
Autodesk Express Viewer	C:\Program Files\Autodesk\Autodesk Express Viewer	15:00, September 8, 2011
Adobe Flash Player Activex	C:\WINDOWS\system32\Macromed\Flash\ FlashUtil10w_ActiveX.exe-matntain activex	15:00, June 8, 2011
CHATM2010	C:\Rabbit\chatm\exe\Quety_net.exe	18:00, March 20, 2011
Movie Maker	C:\Program Files\Movie Maker	17:30, April26, 2011

[0047] Step 303: obtaining the last start-up time of the specified application program according to the standardized installing routes as index.

[0048] In detail, the present invention uses the standardized installing routes of the above list as the index, to look up the last start-up time of the application program.

[0049] Step 304: comparing the last start-up time and the current time, to determine the using-frequency of the specified application program.

[0050] In detail, when obtaining the last start-up time of the specified application program, the exemplary embodiment compares it with the current time. If the period between the last start-up time and the current time is less than three days, it may determine the application program is often used. If the period between the last start-up time and the current time is less than seven days and more than three days, it may determine the application program is sometimes used. If the period between the last start-up time and the current time is less than thirty days and more than seven days, it may determine the application program is seldom used. If the period between the last start-up time and the current time is

more than thirty days, it may determine the application program is never used.

[0051] It should be noted that, some application programs, such as the software of input method, the plug-in software, etc., which should be specifically processed, have not the exe files. For these application programs, the present invention should call the API Create Process to monitor the using statuses of the dll files, which is same with the above description for monitoring the exe files, and not described in following.

[0052] The method for obtaining the using-frequency of the application program of the present invention, can intercept the process start function of the system, to easily and efficiently obtain the using-frequency of the application program installed in the operation system of the computer. Therefore, the present invention may easily manage the application program installed in the computer, and may provide the base for optimizing the operation system of the computer.

[0053] FIG. 7 is a schematic view of a device for obtaining a using-frequency of a specified application program in accordance with a first exemplary embodiment of the present invention.

[0054] The device for obtaining the using-frequency of the specified application program provided in the exemplary embodiment of the present invention, comprises:

a route-obtaining module 10, configured for obtaining an installing route of the specified application program according to a system-registering information list;

a program-monitoring module 11, configured for calling an API, to monitor the operation of the specified application program;

an information-recording module 12, configured for recording a last start-up time and the installing route of the specified application program;

a time-obtaining module 13, configured for obtaining the last start-up time of the specified application program and the current time from the information-recording module, according to the installing route as an index ;

a frequency-calculating module 14, configured for comparing the last start-up time of the specified application program and the current time, to determine the using-frequency of the application program.

[0055] The device of the exemplary embodiment of the present invention, accurately obtains the installing route of the specified application program according to the system-registering information list, and monitors the start-up situations of the application program based on the installing route, such as to judge the using-frequency of the application program according to the recorded last start-up time. The method for obtaining the using-frequency of the application program provided in the exemplary embodiment of the present invention, is real and effective, impersonal and accurate, such that the user can distinctly know about the using status of the application program installed in the operation system of the computer. Furthermore, the user may further optimize the operation system of the computer according to the obtained using-frequency of the application program.

[0056] FIG. 8 is a schematic view of the device for obtaining the using-frequency of the specified application program in accordance with a second exemplary embodiment of the present invention.

[0057] The exemplary embodiment will describe components of the route-obtaining module 10, which comprises: a directional-obtaining unit 100.

[0058] The directional-obtaining unit 100, is configured for obtaining the installing route of the specified application program, according to an entry of a key being InstallLocation in the system-registering information list, or obtaining the installing route of the specified application program, according to an entry of a key being UnstallString in the system-registering information list.

[0059] In detail, a program is called to obtain the installing route of the application program from the entry of the key being InstallLocation, which is under SOFTWARE\\Microsoft\\Windows\\Current Version\\Uninstall of the system-registering information list.

[0060] For example, as shown in FIG. 2 which is a schematic view for showing all entries of the application program of "Autodesk Express Viewer" under SOFTWARE\\Microsoft\\Windows\\Current Version\\Uninstall of the system-registering information list, from the entries of FIG. 2, the entry of the key being "InstallLocation" is found, to show the installing route is "C:\Program Files\Autodesk\Autodesk Express Viewer".

[0061] In addition, some application programs do not have the entry of the key being "InstallLocation" in the system-registering information list. Therefore, the second mode is performed to call a program to read the entry of the key being UnstallString under SOFTWARE\\Microsoft\\Windows\\Current Version\\Uninstall of the system-registering information list, and then remove exe file name, to obtain the installing route of the application program.

[0062] For example, as shown in FIG. 3 which is a schematic view for showing all entries of the application program of "Adobe Flash Player Activex" under "SOFTWARE\\Microsoft\\Windows\\Current Version\\Uninstall" of the system-

registering information list, the entry of the key being "InstallLocation" does not existed. Thus, the entry of the key being "UnstallString" is found, which is presented as "C:\\WINDOWS\system32\Macromed\Flash\ FlashUtil10w_ActiveX.exe-matntain activex". Then it removes the exe file name, to obtain the installing route of "C:\WINDOWS\system32\Macromed\Flash\".

[0063] The route-obtaining module 10 may further comprise a matching unit 101 and a route-obtaining unit 102.

[0064] The matching unit 101 is configured for fuzzily matching names of shortcuts of application programs with the name of the specified application program recorded in the system-registering information list, or fuzzily matching names of application programs of each project folder of each disk, with the name of the specified application program in the system-registering information list.

[0065] The route-obtaining unit 102 is configured for if successfully matching a name of a shortcut of an application program with the name of the specified application program recorded in the system-registering information list, obtaining a directing route of the shortcut as the installing route of the specified application program; or if successfully matching a name of an application program of a project folder of a disk with the name of the specified application program in the system-registering information list, regarding a route of the application program of the project folder of the disk as the installing route of the specified application program.

[0066] In detail, some application programs not only do not have the entry of the key being "InstallLocation", but also do not have the entry of the key being "UnstallString" in the system-registering information list. Therefore, the present invention may further enumerate shortcuts of application programs, which may be in user desktop, start menu, or start shortcut, etc, and then fuzzily matching names of the shortcuts of the application programs with the name of the specified application program in the system-registering information list by the matching unit 101. If successfully matching, the present invention may regard the directing route of the shortcut as the installing route of the specified application program.

[0067] For example, an operation system of a computer has installed application programs, such as "Tencent TM2009", "CHATM2010", "Windstorm Image Sound", etc, and shortcuts of the application programs are located in the user desktop, the start menu, or the start shortcut, etc., respectively. This step will enumerate the shortcuts, then matching the names of the shortcuts with the names of the application programs in the system-registering information. If a name of a shortcut is "CHATM2010" and it successfully matches with "CHATM trademark-querying software" in the system-registering information list when they fuzzily matches, the directing route of the shortcut is obtained, which is as shown in FIG. 4, and the directing route of the shortcut is "Target (T) C:\Rabbit\chatm\exe\Quety_net.exe". Then the directing route thereof is regarded as the installing route of the application program of "CHATM2010".

[0068] In addition, the matching unit 101 further fuzzily matches names of application programs of each project folder of each disk, with the name of the application program in the system-registering information list, and if a name of an application program of a project folder of a disk is successfully matched with the name of the application program in the system-registering information list, the route-obtaining unit 102 regards a route of the application program of the project folder of the disk as the installing route of the application program.

[0069] In detail, for some application programs, they should enumerate application programs of each folder under the project directory of Project Files of each disk. The matching unit 101 should match names of the application programs with the name of the application program of the system-registering information list. If successfully matching, the route-obtaining unit 102 regards a route of an application program of a project folder which matching with the name of the application program of the system-registering information list, as the installing route of the application program.

[0070] For example, a folder named as "Movie Maker" exists in the folder of Program Files of C disk. If the matching unit 101 successfully matches the name of "Movie Maker" with the name of the application program in the system-registering information list, the route-obtaining unit 102 regards the route of "C:\Program Files\Movie Maker" of the application program of "Movie Maker" as the installing route of the application program.

[0071] For the software located in the directory of system, such as Application Verifier, Flash Palyer, it needs to use the exepath to match, and the method is similar with the above, which is not described herein.

[0072] The route-obtaining module of the exemplary embodiment of the present invention may use the above four modes to obtain the installing route of the application program, thus it can furthest accurately obtain the installing route of the application program for analyze the using-frequency of the application program.

[0073] FIG. 9 is a schematic view of the device for obtaining the using-frequency of the specified application program in accordance with a third exemplary embodiment of the present invention.

[0074] The exemplary embodiment will describe the program-monitoring module 11 of the present invention in detail, which comprises an interface-calling unit 110, a command-line processing unit 111 and a program-monitoring unit 112.

[0075] The interface-calling unit 110 is configured for calling the API to obtain command-line arguments of application programs which are running.

[0076] In detail, the monitor of the API Create Process of the interface-calling unit 110 is open. When the Create Process is called, the command-line arguments in the Create Process are obtained.

[0077] The command-line processing unit 111 is configured for inputting the command-line arguments of the application programs into an independent processing thread.

[0078] In detail, for not influencing the performance of calling the Create Process, the command-line processing unit 111 inputs the command-line arguments into an independent processing thread.

[0079] The program-monitoring unit 112 is configured for monitoring the operation of the application programs corresponding to the command-line arguments respectively.

[0080] The device of the exemplary embodiment may pre-monitor the running statuses of the various application programs, and the above monitoring mode can accurately obtain the status of the various application programs, to obtain the start-up time of the various application programs.

[0081] FIG. 10 is a schematic view of the device for obtaining the using-frequency of the specified application program in accordance with a fourth exemplary embodiment of the present invention.

[0082] The exemplary embodiment will describe the components of the information recording module 12 and the functions thereof, which are described in following.

[0083] The information-recording module 12 comprises a time-recording unit 120. The time-recording unit 120 is configured for obtaining the last start-up time of the application program from the processing thread and recording it.

[0084] In detail, the program-monitoring unit 112 monitors the various application programs in the processing thread, and the time-recording unit 120 may obtain the start-up time of any one specific application program. In the exemplary embodiment, the time-recording unit 120 only needs to record the last start-up time of the application programs. In detail, the time-recording unit 120 may use notepad or SQlite to record the start-up time of the various application programs. It should be noted that, the SQlite is a light-duty database, and complies with the ACID-associated database managing system. The SQlite is embedded, and occupies little resource, and can support the main-trend Operation System, such as Windows or Linux or Unix, etc.

[0085] The information-recording module 12 may further comprise a route-standardizing unit 121 configured for performing a standardization process for the installing route.

[0086] In detail, installing routes obtained in the above modes of the present invention may be different in form. In the exemplary embodiment, the route-standardizing unit 121 needs to perform the standardization process for the installing routes, thus they are quickly looked up and compared. The standardization process for the installing routes, comprises: unifying characters of the installing routes as lowercase characters; unifying names of the installing routes to comprise short file names, long file names, environment variables; standardizing the installing routes according to common-name rules.

[0087] The information-recording module 12 may further comprise a route-recording unit 122 configured for recording the standardized installing route. In detail, the route-recording unit 122 may use the tools of Notepad or SQlite, to build an application-program information list, for recording the last start-up time and the installing routes of the various application programs, which are shown in table one.

[0088] The time-obtaining module 13 looks up the last start-up time of the application program under the installing route according to the installing route as the index, in the table one recorded by the route-recording unit 122.

[0089] The frequency-calculating module 14 is configured for comparing the last start-up time and the current time, to determine the using-frequency of the application program.

[0090] In detail, when the time-obtaining module 13 obtains the last start-up time of the application program, the frequency-calculating module 14 compares it with the current time. If the period between the last start-up time and the current time is less than three days, it may determine the application program is often used. If the period between the last start-up time and the current time is less than seven days and more than three days, it may determine the application program is sometimes used. If the period between the last start-up time and the current time is less than thirty days and more than seven days, it may determine the application program is seldom used. If the period between the last start-up time and the current time is more than thirty days, it may determine the application program is never used.

[0091] It should be noted that, some application programs, such as the software of input method, the plug-in software, etc., which should be specifically processed, have not the exe files. For these application programs, the present invention should call the API Create Process to monitor the using statuses of the dll files, which is same with the above description for monitoring the exe files, and not described in following.

[0092] The device for obtaining the using-frequency of the application program of the present invention, can intercept the process start function of the system, to easily and efficiently obtain the using-frequency of the application program installed in the operation system of the computer. Therefore, the device of the present invention may easily manage the application program installed in the computer, and may provide the base for optimizing the operation system of the computer.

[0093] Those skilled in the art may understand that, all or part of the processes in the methods of above-mentioned embodiments may be completed via related hardware instructed by the computer program. The program may be stored in a computer readable storage medium. When executing the program, the processes of above-mentioned method embodiments may be included. The storage medium may be disk, Compact Disc (CD), Read-Only Memory (ROM) or Random Access Memory (RAM), etc.

Claims

1. A method for obtaining a using-frequency of a specified application program comprising:

   obtaining an installing route of the specified application program according to a system-registering information list (step 100);
   calling an application programming interface (API) to monitor operation of the specified application program (step 101);
   recording a last start-up time and the installing route of the specified application program (step 102);
   obtaining the last start-up time of the specified application program based on the installing route as an index (step 103, step 303); the method being **characterised by**
   comparing the last start-up time and a current time to determine the using-frequency of the specified application program (step 104, step 304), wherein the step of obtaining the installing route of the specified application program according to the system-registering information list (step 100) comprises performing the following steps in sequence:

   step 1, obtaining the installing route of the specified application program, according to an entry of a key being InstallLocation in the system-registering information list;
   step 2, obtaining the installing route of the specified application program, according to an entry of a key being UnstallString in the system-registering information list;
   step 3, fuzzily matching names of shortcuts of application programs with name of the specified application program recorded in the system-registering information list, and if successfully matching a name of a shortcut of an application program with the name of the specified application program recorded in the system-registering information list, obtaining a directing route of the shortcut as the installing route of the specified application program;
   step 4, fuzzily matching names of application programs of each project folder of each disk with the name of the specified application program recorded in the system-registering information list, and if successfully matching a name of an application program of a project folder of a disk with the name of the specified application program recorded in the system-registering information list, regarding a route of the application program of the project folder of the disk as the installing route of the specified application program.

2. The method according to claim 1, wherein the step of calling the API to monitor the operation of the specified application program (step 101)comprises:

   calling the API to obtain command-line arguments of the currently-running specified application programs (step 200);
   inputting the command-line arguments of the specified application programs into an independent processing thread (step 201);
   monitoring operation of the specified application programs corresponding to the various command-line arguments respectively (step 202).

3. The method according to claim 2, wherein the step of recording the last start-up time and the installing route of the specified application program (step 102) comprises:

   obtaining the last start-up time of the specified application program from the processing thread and recording it (step 300);
   performing a standardization process for the installing route of the specified application program (step 301);

   recording the standardized installing route (step 302), wherein the step of performing the standardization process for the installing route (step 301) comprises:

   unifying characters of the installing route as lowercase characters;
   unifying names of the installing route to comprise short file names, long file names, and environment variables;
   standardizing the installing route according to common-namerules.

4. A device for obtaining a using-frequency of a specified application program comprising:

   a route-obtaining module (10), configured for obtaining an installing route of the specified application program

according to a system-registering information list;
a program-monitoring module (11), configured for calling an application programming interface (API) to monitor operation of the specified application program;
an information-recording module (12), configured for recording a last start-up time and the installing route of the specified application program;
a time-obtaining module (13), configured for obtaining the last start-up time of the specified application program from the information-recording module (12), based on the installing route as an index; the device being **characterised by**
a frequency-calculating module (14), configured for comparing the last start-up time and a current time, to determine the using-frequency of the specified application program, wherein
the route-obtaining module (10), further configured for obtaining an installing route of the specified application program by performing the following steps in sequence:

step 1, obtaining the installing route of the specified application program, according to an entry of a key being InstallLocation in the system-registering information list;
step 2, obtaining the installing route of the specified application program, according to an entry of a key being UnstallString in the system-registering information list;
step 3, fuzzily matching names of shortcuts of application programs with name of the specified application program recorded in the system-registering information list, and if successfully matching a name of a shortcut of an application program with the name of the specified application program recorded in the system-registering information list, obtaining a directing route of the shortcut as the installing route of the specified application program;
step 4, fuzzily matching names of application programs of each project folder of each disk with the name of the specified application program recorded in the system-registering information list, and if successfully matching a name of an application program of a project folder of a disk with the name of the specified application program recorded in the system-registering information list, regarding a route of the application program of the project folder of the disk as the installing route of the specified application program.

5. The device according to claim 4, wherein the program-monitoring module (11) comprises:

an interface-calling unit (110), configured for calling the API to obtain command-line arguments of specified application programs which are running;
a command-line processing unit (111), configured for inputting the command-line arguments of the specified application programs into an independent processing thread;
a program-monitoring unit (112), configured for monitoring operation of the specified application programs corresponding to the command-line arguments respectively.

6. The device according to claim 5, wherein the information-recording module (12) comprises:

a time-recording unit (120), configured for obtaining the last start-up time of the specified application program from the processing thread and recording it;
a route-standardizing unit (121), configured for performing a standardization process for the installing route;

a route-recording unit (122), configured for recording the standardized installing route, wherein the standardization process performed by the route-standardizing unit (121) comprises:

unifying characters of the installing route as lowercase characters;
unifying names of the installing route to comprise short file names, long file names and environment variables; and standardizing the installing route according to common-name rules.

Patentansprüche

1. Verfahren zum Ermitteln einer Verwendungshäufigkeit eines angegebenen Anwendungsprogramms, umfassend:

Ermitteln eines Installationspfads des angegebenen Anwendungsprogramms gemäß einer Systemregistrierungsinformationsliste (Schritt 100);
Aufrufen einer Anwendungsprogrammierschnittstelle (API) zum Überwachen des Betriebs des angegebenen

Anwendungsprogramms (Schritt 101);
Aufzeichnen eines letzten Startzeitpunkts und des Installationspfads des angegebenen Anwendungsprogramms (Schritt 102);
Ermitteln des letzten Startzeitpunkts des angegebenen Anwendungsprogramms auf Grundlage des Installationspfads als Index (Schritt 103, Schritt 303);
wobei das Verfahren **gekennzeichnet ist durch**
Vergleichen des letzten Startzeitpunkts und eines aktuellen Zeitpunkts, um die Verwendungshäufigkeit des angegebenen Anwendungsprogramms zu bestimmen (Schritt 104, Schritt 304), wobei der Schritt des Ermittelns des Installationspfads des angegebenen Anwendungsprogramms gemäß der Systemregistrierungsinformationsliste (Schritt 100) ein Durchführen der folgenden Schritte nacheinander umfasst:

Schritt 1, Ermitteln des Installationspfads des angegebenen Anwendungsprogramms gemäß einem Eintrag eines Schlüssels, bei dem es sich um InstallLocation handelt, in der Systemregistrierungsinformationsliste;
Schritt 2, Ermitteln des Installationspfads des angegebenen Anwendungsprogramms gemäß einem Eintrag eines Schlüssels, bei dem es sich um UninstallString handelt, in der Systemregistrierungsinformationsliste;
Schritt 3, unscharfes Abgleichen von Namen von Verknüpfungen von Anwendungsprogrammen mit dem Namen des angegebenen Anwendungsprogramms, der in der Systemregistrierungsinformationsliste aufgezeichnet ist, und falls ein Name einer Verknüpfung eines Anwendungsprogramms erfolgreich dem Namen des angegebenen Anwendungsprogramms, der in der Systemregistrierungsinformationsliste aufgezeichnet ist, zugeordnet wird, Ermitteln eines Zielpfads der Verknüpfung als den Installationspfad des angegebenen Anwendungsprogramms;
Schritt 4, unscharfes Abgleichen von Namen von Anwendungsprogrammen jedes Projektordners jedes Datenträgers mit dem Namen des angegebenen Anwendungsprogramms, der in der Systemregistrierungsinformationsliste aufgezeichnet ist, und falls ein Name eines Anwendungsprogramms eines Projektorderns eines Datenträgers erfolgreich dem Namen des angegebenen Anwendungsprogramms, der in der Systemregistrierungsinformationsliste aufgezeichnet ist, zugeordnet wird, Betrachten eines Pfads des Anwendungsprogramms des Projektordners des Datenträgers als den Installationspfad des angegebenen Anwendungsprogramms.

2. Verfahren nach Anspruch 1, wobei der Schritt des Aufrufens der API zum Überwachen des Betriebs des angegebenen Anwendungsprogramms (Schritt 101) umfasst:

Aufrufen der API zum Ermitteln von Befehlszeilenargumenten der derzeit ausgeführten angegebenen Anwendungsprogramme (Schritt 200);
Eingeben der Befehlszeilenargumente der angegebenen Anwendungsprogramme in einen unabhängigen Verarbeitungsstrang (Schritt 201);
Überwachen des Betriebs der angegebenen Anwendungsprogramme entsprechend der jeweiligen verschiedenen Befehlszeilenargumente (Schritt 202).

3. Verfahren nach Anspruch 2, wobei der Schritt des Aufzeichnens des letzten Startzeitpunkts und des Installationspfads des angegebenen Anwendungsprogramms (Schritt 102) umfasst:

Ermitteln des letzten Startzeitpunkts des angegebenen Anwendungsprogramms aus dem Verarbeitungsstrang und Aufzeichnen desselben (Schritt 300);
Durchführen eines Standardisierungsprozesses für den Installationspfad des angegebenen Anwendungsprogramms (Schritt 301);
Aufzeichnen des standardisierten Installationspfads (Schritt 302), wobei der Schritt des Durchführens des Standardisierungsprozesses für den Installationspfad (Schritt 301) umfasst:

Vereinheitlichen von Zeichen des Installationspfads als kleingeschriebene Zeichen;
Vereinheitlichen von Namen des Installationspfads, so dass sie kurze Dateinamen, lange Dateinamen und Umgebungsvariablen umfassen;
Standardisieren des Installationspfads gemäß allgemeinen Namensregeln.

4. Vorrichtung zum Ermitteln einer Verwendungshäufigkeit eines angegebenen Anwendungsprogramms, umfassend:

ein Pfadermittlungsmodul (10), das zum Ermitteln eines Installationspfads des angegebenen Anwendungsprogramms gemäß einer Systemregistrierungsinformationsliste konfiguriert ist;

ein Programmüberwachungsmodul (11), das zum Aufrufen einer Anwendungsprogrammschnittstelle (API) zum Überwachen des Betriebs des angegebenen Anwendungsprogramms konfiguriert ist;
ein Informationsaufzeichnungsmodul (12), das zum Aufzeichnen eines letzten Startzeitpunkts und des Installationspfads des angegebenen Anwendungsprogramms konfiguriert ist;
ein Zeitermittlungsmodul (13), das zum Ermitteln des letzten Startzeitpunkts des angegebenen Anwendungsprogramms vom Informationsaufzeichnungsmodul (12) auf Grundlage des Installationspfads als Index konfiguriert ist;
wobei die Vorrichtung **gekennzeichnet ist durch**
ein Häufigkeitsberechnungsmodul (14), das zum Vergleichen des letzten Startzeitpunkts und eines aktuellen Zeitpunkts konfiguriert ist, um die Verwendungshäufigkeit des angegebenen Anwendungsprogramms zu bestimmen, wobei
das Pfadermittlungsmodul (10) ferner zum Ermitteln eines Installationspfads des angegebenen Anwendungsprogramms durch Durchführen der folgenden Schritte nacheinander konfiguriert ist:

Schritt 1, Ermitteln des Installationspfads des angegebenen Anwendungsprogramms gemäß einem Eintrag eines Schlüssels, bei dem es sich um InstallLocation handelt, in der Systemregistrierungsinformationsliste;
Schritt 2, Ermitteln des Installationspfads des angegebenen Anwendungsprogramms gemäß einem Eintrag eines Schlüssels, bei dem es sich um UninstallString handelt, in der Systemregistrierungsinformationsliste;
Schritt 3, unscharfes Abgleichen von Namen von Verknüpfungen von Anwendungsprogrammen mit dem Namen des angegebenen Anwendungsprogramms, der in der Systemregistrierungsinformationsliste aufgezeichnet ist, und falls ein Name einer Verknüpfung eines Anwendungsprogramms erfolgreich dem Namen des angegebenen Anwendungsprogramms, der in der Systemregistrierungsinformationsliste aufgezeichnet ist, zugeordnet wird, Ermitteln eines Zielpfads der Verknüpfung als den Installationspfad des angegebenen Anwendungsprogramms;
Schritt 4, unscharfes Abgleichen von Namen von Anwendungsprogrammen jedes Projektordners jedes Datenträgers mit dem Namen des angegebenen Anwendungsprogramms, der in der Systemregistrierungsinformationsliste aufgezeichnet ist, und falls ein Name eines Anwendungsprogramms eines Projektorderns eines Datenträgers erfolgreich dem Namen des angegebenen Anwendungsprogramms, der in der Systemregistrierungsinformationsliste aufgezeichnet ist, zugeordnet wird, Betrachten eines Pfads des Anwendungsprogramms des Projektordners des Datenträgers als den Installationspfad des angegebenen Anwendungsprogramms.

5. Vorrichtung nach Anspruch 4, wobei das Programmüberwachungsmodul (11) umfasst:

eine Schnittstellenaufrufeinheit (110), die zum Aufrufen der API zum Ermitteln von Befehlszeilenargumenten von angegebenen Anwendungsprogrammen, die ausgeführt werden, konfiguriert ist;
eine Befehlszeilenverarbeitungseinheit (111), die zum Eingeben der Befehlszeilenargumente der angegebenen Anwendungsprogramme in einen unabhängigen Verarbeitungsstrang konfiguriert ist;
eine Programmüberwachungseinheit (112), die zum Überwachen des Betriebs der angegebenen Anwendungsprogramme entsprechend den jeweiligen Befehlszeilenargumenten konfiguriert ist.

6. Vorrichtung nach Anspruch 5, wobei das Informationsaufzeichnungsmodul (12) umfasst:

eine Zeitaufzeichnungseinheit (120), die zum Ermitteln des letzten Startzeitpunkts des angegebenen Anwendungsprogramms aus dem Verarbeitungsstrang und Aufzeichnen desselben konfiguriert ist;
eine Pfadstandardisierungseinheit (121), die zum Durchführen eines Standardisierungsprozesses für den Installationspfad konfiguriert ist;
eine Pfadaufzeichnungseinheit (122), die zum Aufzeichnen des standardisierten Installationspfads konfiguriert ist, wobei der von der Pfadstandardisierungseinheit (121) durchgeführte Standardisierungsprozess umfasst:

Vereinheitlichen von Zeichen des Installationspfads als kleingeschriebene Zeichen;
Vereinheitlichen von Namen des Installationspfads, so dass sie kurze Dateinamen, lange Dateinamen und Umgebungsvariablen umfassen; und
Standardisieren des Installationspfads gemäß allgemeinen Namensregeln.

Revendications

1. Procédé d'obtention d'une fréquence d'utilisation d'un programme d'application spécifié comprenant :

   l'obtention d'un chemin d'installation du programme d'application spécifié selon une liste d'informations d'inscription de système (étape 100) ;
   l'appel d'une interface de programmation d'applications (API) pour surveiller un fonctionnement du programme d'application spécifié (étape 101) ;
   l'enregistrement d'un dernier temps de démarrage et du chemin d'installation du programme d'application spécifié (étape 102) ;
   l'obtention du dernier temps de démarrage du programme d'application spécifié d'après le chemin d'installation en tant qu'indice (étape 103, étape 303) ;
   le procédé étant **caractérisé par**
   la comparaison du dernier temps de démarrage et d'un temps actuel pour déterminer la fréquence d'utilisation du programme d'application spécifié (étape 104, étape 304), dans lequel l'étape d'obtention du chemin d'installation du programme d'application spécifié selon la liste d'informations d'inscription de système (étape 100) comprend la réalisation des étapes suivantes en séquence :

   étape 1, obtention du chemin d'installation du programme d'application spécifié, selon une entrée d'une clé qui est InstallLocation dans la liste d'informations d'inscription de système ;
   étape 2, obtention du chemin d'installation du programme d'application spécifié, selon une entrée d'une clé qui est UnstallString dans la liste d'informations d'inscription de système ;
   étape 3, concordance floue de noms de raccourcis de programmes d'application avec un nom du programme d'application spécifié enregistré dans la liste d'informations d'inscription de système, et en cas de concordance réussie d'un nom d'un raccourci d'un programme d'application avec le nom du programme d'application spécifié enregistré dans la liste d'informations d'inscription de système, obtention d'un chemin de direction du raccourci en tant que chemin d'installation du programme d'application spécifié ;
   étape 4, concordance floue de noms de programmes d'application de chaque dossier projet de chaque disque avec le nom du programme d'application spécifié enregistré dans la liste d'informations d'inscription de système, et en cas de concordance réussie d'un nom d'un programme d'application d'un dossier projet d'un disque avec le nom du programme d'application spécifié enregistré dans la liste d'informations d'inscription de système, fait de considérer un chemin du programme d'application du dossier projet du disque en tant que chemin d'installation du programme d'application spécifié.

2. Procédé selon la revendication 1, dans lequel l'étape d'appel de l'API pour surveiller le fonctionnement du programme d'application spécifié (étape 101) comprend :

   l'appel de l'API pour obtenir des arguments de ligne de commande des programmes d'application spécifiés s'exécutant actuellement (étape 200) ;
   l'entrée des arguments de ligne de commande des programmes d'application spécifiés dans un fil de traitement indépendant (étape 201) ;
   la surveillance du fonctionnement des programmes d'application spécifiés correspondant aux divers arguments de ligne de commande, respectivement (étape 202).

3. Procédé selon la revendication 2, dans lequel l'étape d'enregistrement du dernier temps de démarrage et du chemin d'installation du programme d'application spécifié (étape 102) comprend :

   l'obtention du dernier temps de démarrage du programme d'application spécifié à partir du fil de traitement et son enregistrement (étape 300) ;
   la réalisation d'un processus de standardisation pour le chemin d'installation du programme d'application spécifié (étape 301) ;
   l'enregistrement du chemin d'installation standardisé (étape 302), dans lequel l'étape de réalisation du processus de standardisation pour le chemin d'installation (étape 301) comprend :

   l'uniformisation de caractères du chemin d'installation en tant que caractères en minuscule ;
   l'uniformisation de noms du chemin d'installation pour qu'ils comprennent des noms de fichiers courts, des noms de fichiers longs, et des variables d'environnement ;
   la standardisation du chemin d'installation selon des règles de nom courant.

4. Dispositif d'obtention d'une fréquence d'utilisation d'un programme d'application spécifié, comprenant :

un module d'obtention de chemin (10), configuré pour obtenir un chemin d'installation du programme d'application spécifié selon une liste d'informations d'inscription de système ;
un module de surveillance de programme (11), configuré pour appeler une interface de programmation d'applications (API) pour surveiller un fonctionnement du programme d'application spécifié ;
un module d'enregistrement d'informations (12), configuré pour enregistrer un dernier temps de démarrage et le chemin d'installation du programme d'application spécifié ;
un module d'obtention de temps (13), configuré pour obtenir le dernier temps de démarrage du programme d'application spécifié à partir du module d'enregistrement d'informations (12), d'après le chemin d'installation en tant qu'indice ;
le dispositif étant **caractérisé par**
un module de calcul de fréquence (14), configuré pour comparer le dernier temps de démarrage et un temps actuel, afin de déterminer la fréquence d'utilisation du programme d'application spécifié, dans lequel
le module d'obtention de chemin (10), configuré en outre pour obtenir un chemin d'installation du programme d'application spécifié en réalisant les étapes suivantes en séquence :

étape 1, obtention du chemin d'installation du programme d'application spécifié, selon une entrée d'une clé qui est InstallLocation dans la liste d'informations d'inscription de système ;
étape 2, obtention du chemin d'installation du programme d'application spécifié, selon une entrée d'une clé qui est UnstallString dans la liste d'informations d'inscription de système ;
étape 3, concordance floue de noms de raccourcis de programmes d'application avec un nom du programme d'application spécifié enregistré dans la liste d'informations d'inscription de système, et en cas de concordance réussie d'un nom d'un raccourci d'un programme d'application avec le nom du programme d'application spécifié enregistré dans la liste d'informations d'inscription de système, obtention d'un chemin de direction du raccourci en tant que chemin d'installation du programme d'application spécifié ;
étape 4, concordance floue de noms de programmes d'application de chaque dossier projet de chaque disque avec le nom du programme d'application spécifié enregistré dans la liste d'informations d'inscription de système, et en cas de concordance réussie d'un nom d'un programme d'application d'un dossier projet d'un disque avec le nom du programme d'application spécifié enregistré dans la liste d'informations d'inscription de système, fait de considérer un chemin du programme d'application du dossier projet du disque en tant que chemin d'installation du programme d'application spécifié.

5. Dispositif selon la revendication 4, dans lequel le module de surveillance de programme (11) comprend :

une unité d'appel d'interface (110), configurée pour appeler l'API afin d'obtenir des arguments de ligne de commande de programmes d'application spécifiés qui sont exécutés ;
une unité de traitement de ligne de commande (111), configurée pour entrer les arguments de ligne de commande des programmes d'application spécifiés dans un fil de traitement indépendant ;
une unité de surveillance de programme (112), configurée pour surveiller un fonctionnement des programmes d'application spécifiés correspondant aux arguments de ligne de commande respectivement.

6. Dispositif selon la revendication 5, dans lequel le module d'enregistrement d'informations (12) comprend :

une unité d'enregistrement de temps (120), configurée pour obtenir le dernier temps de démarrage du programme d'application spécifié à partir du fil de traitement et pour l'enregistrer ;
une unité de standardisation de chemin (121), configurée pour réaliser un processus de standardisation pour le chemin d'installation ;
une unité d'enregistrement de chemin (122), configurée pour enregistrer le chemin d'installation standardisé, dans lequel le processus de standardisation réalisé par l'unité de standardisation de chemin (121) comprend :

l'uniformisation de caractères du chemin d'installation en tant que caractères en minuscule ;
l'uniformisation de noms du chemin d'installation pour qu'ils comprennent des noms de fichiers courts, des noms de fichiers longs, et des variables d'environnement ; et
la standardisation du chemin d'installation selon des règles de nom courant.

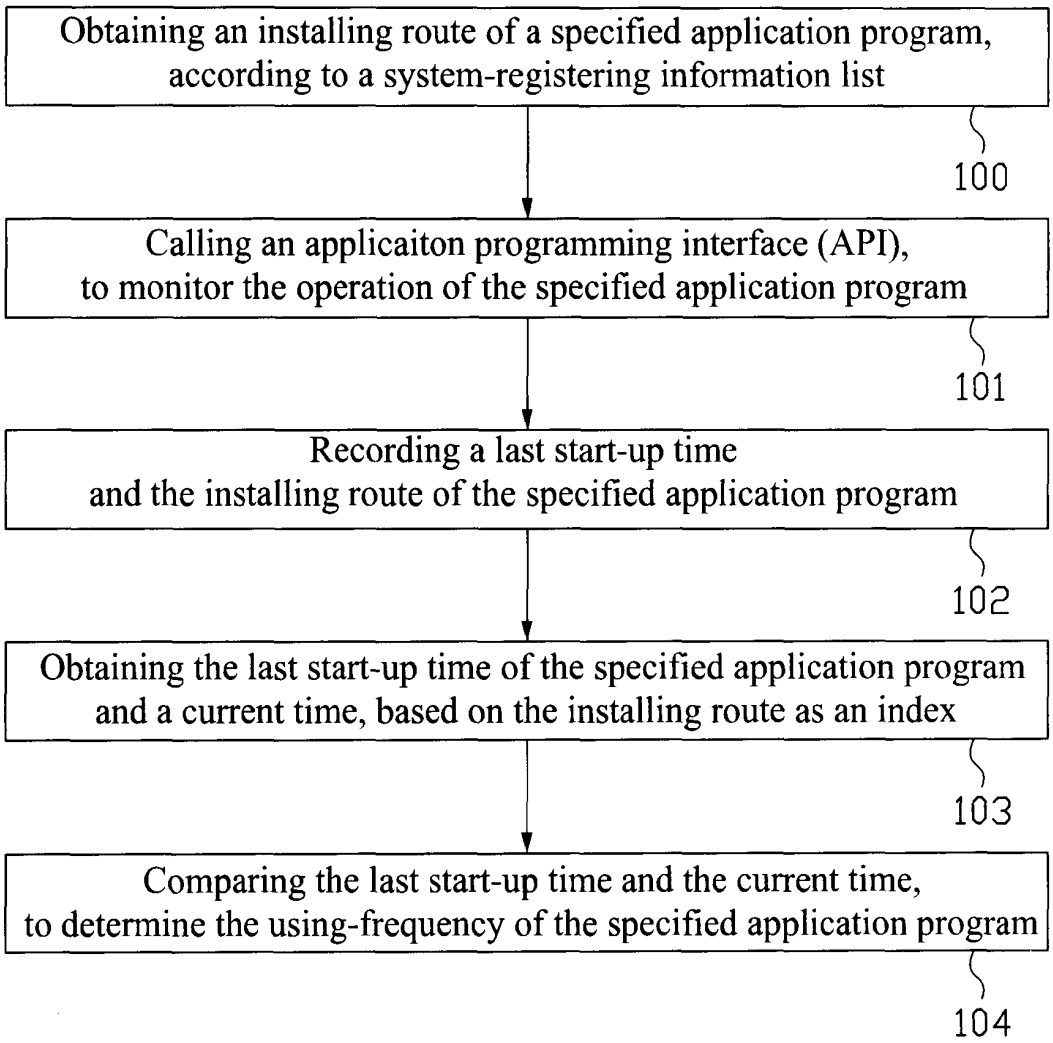
Obtaining an installing route of a specified application program, according to a system-registering information list
100
Calling an applicaiton programming interface (API), to monitor the operation of the specified application program
101
Recording a last start-up time and the installing route of the specified application program
102
Obtaining the last start-up time of the specified application program and a current time, based on the installing route as an index
103
Comparing the last start-up time and the current time, to determine the using-frequency of the specified application program
104

Figure 1

Name	Type	Data
(Default)	REG_SZ	(Default)
DisplayName	REG_SZ	Autodesk Express Viewer
DisplayVersion	REG_SZ	3.1
HelpLink	REG_SZ	http://www.autodesk.com/expressviewer-chs
InstallDate	REG_SZ	2011-08-23 00:51:59
InstallLocation	REG_SZ	C:\Program Files\Autodesk\Autodesk Express...
InstallSource	REG_SZ	D:\Autodesk AutoCAD 2004 原厂简体中文版\Bi...
Publisher	REG_SZ	Autodesk , Inc.
UninstallString	REG_SZ	C:\PROGRA~1\Autodesk\AUTODE~1\Setup.ex...
URLInfoAbout	REG_SZ	http://www.autodesk.com/expressviewer-chs
URLUpdateInfo	REG_SZ	http://www.autodesk.com/expressviewer-chs

Figure 2

Name	Type	Data
(Default)	REG_SZ	(Default)
DisplayIcon	REG_SZ	C:\WINDOWS\system32\Macromed\FlashUtil10w_ActiveX.exe
DisplayName	REG_SZ	Adobe Flash player 10 Activex
DisplayVersion	REG_SZ	10.3.183.7
EstimatedSize	REG_DWORD	0x00001800(6144)
HelpLink	REG_SZ	http://www.adobe.com/flashplayer_support/
NoModify	REG_DWORD	0x00000001(1)
NoRepair	REG_DWORD	0x00000001(1)
Publisher	REG_SZ	Adobe Systems Incorporated
RequiresIESysF...	REG_SZ	4.70.0.1155
UninstallString	REG_SZ	C:\WINDOWS\system32\Macromed\Flash\FlashUtil10w_ActiveX.exe-maintain...
URLInfoAbout	REG_SZ	http://www.adobe.com
URLUpdateInfo	REG_SZ	http://www.adobe.com/getflashplayer/
VersionMajor	REG_DWORD	0x0000000a(10)
VersionMinor	REG_DWORD	0x00000003(3)

Figure 3

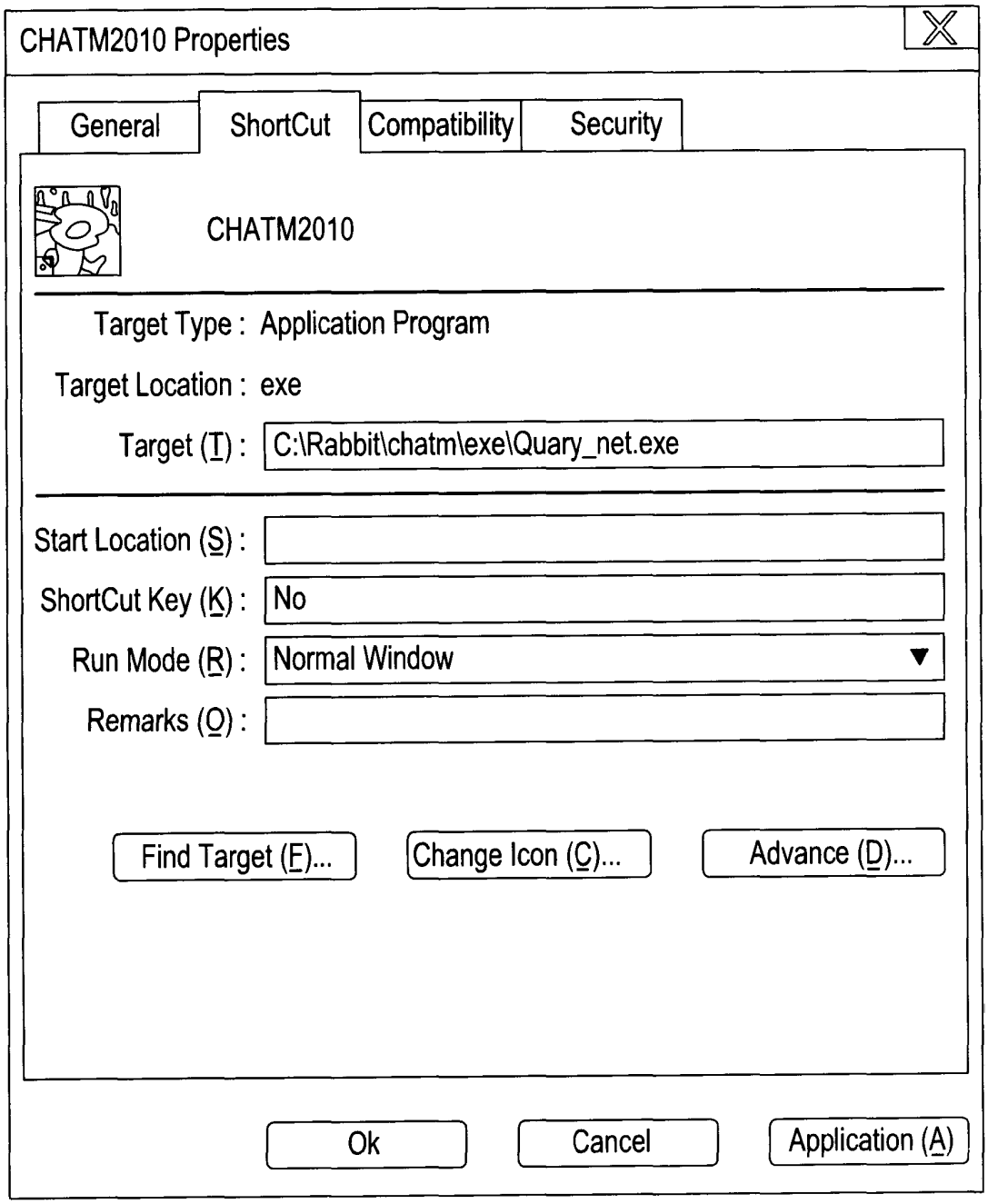
CHATM2010 Properties
General
ShortCut
Compatibility
Security
CHATM2010
Target Type : Application Program
Target Location : exe
Target (T) : C:\Rabbit\chatm\exe\Quary_net.exe
Start Location (S) :
ShortCut Key (K) : No
Run Mode (R) : Normal Window
Remarks (O) :
Find Target (F)...
Change Icon (C)...
Advance (D)...
Ok
Cancel
Application (A)

Figure 4

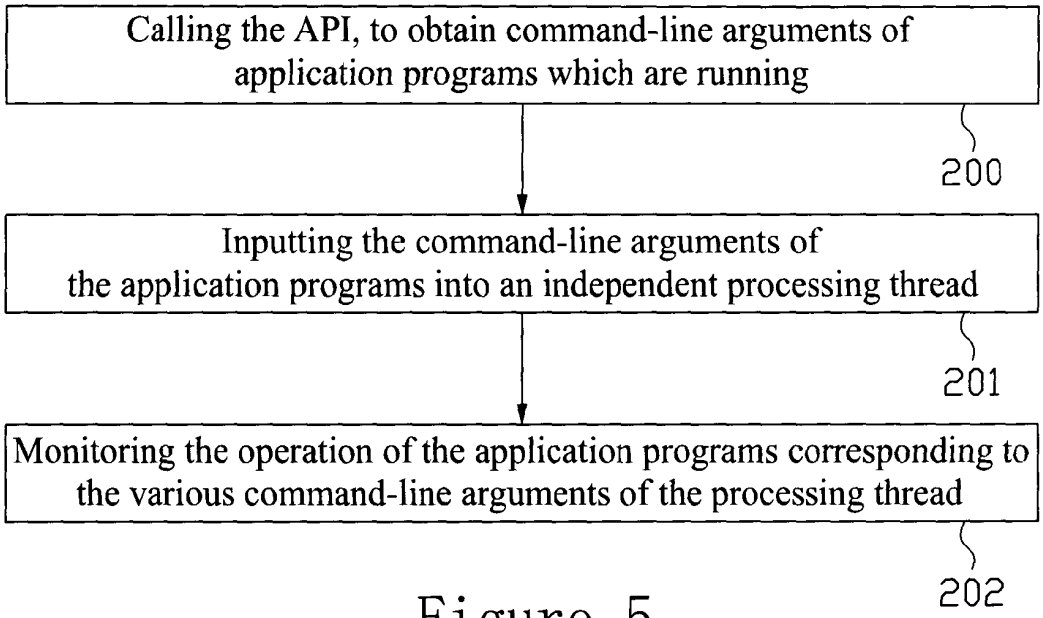

Figure 5

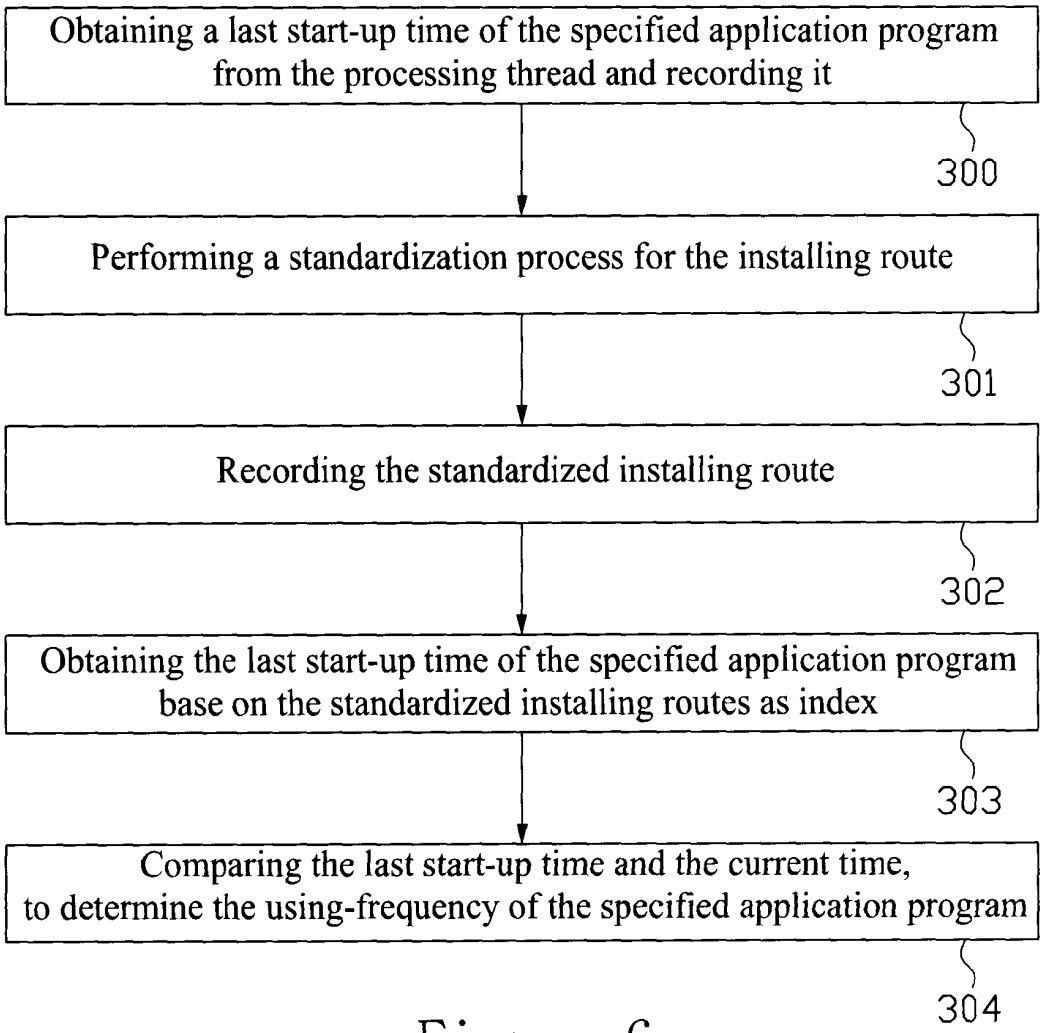

Figure 6

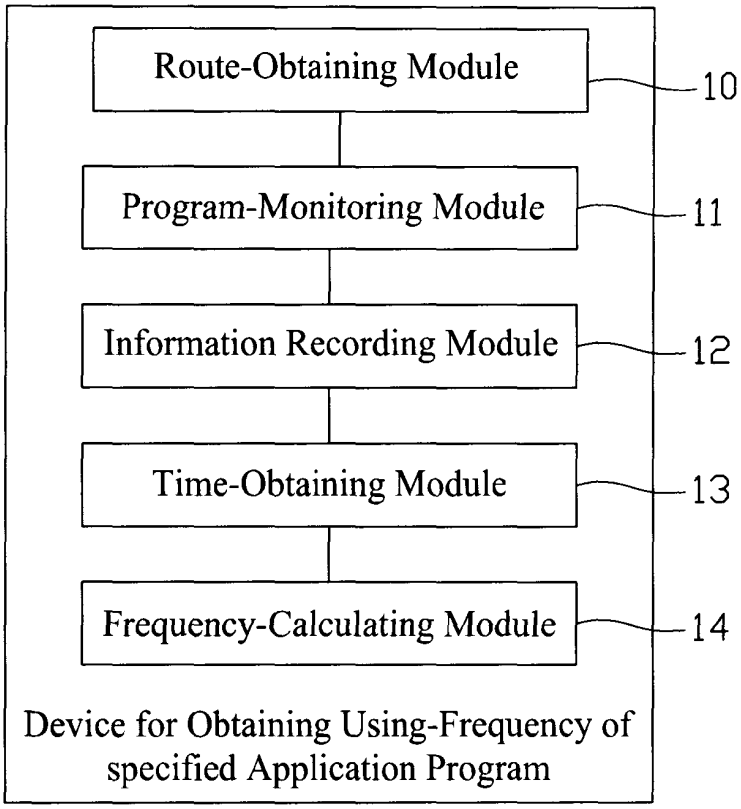

Figure 7

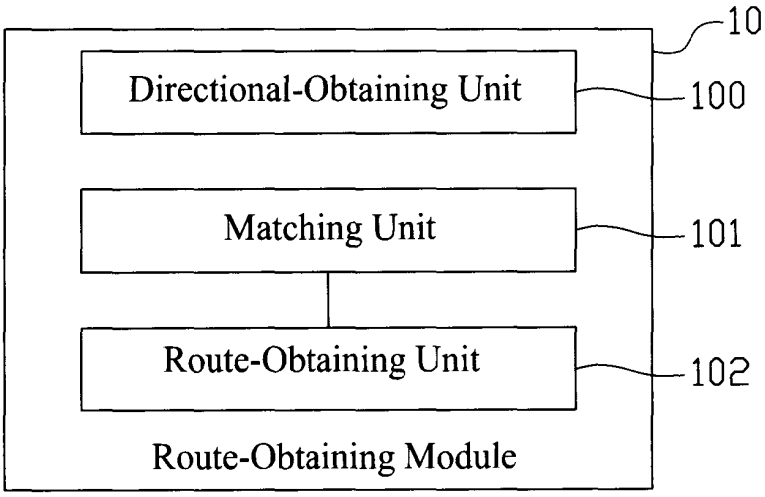

Figure 8

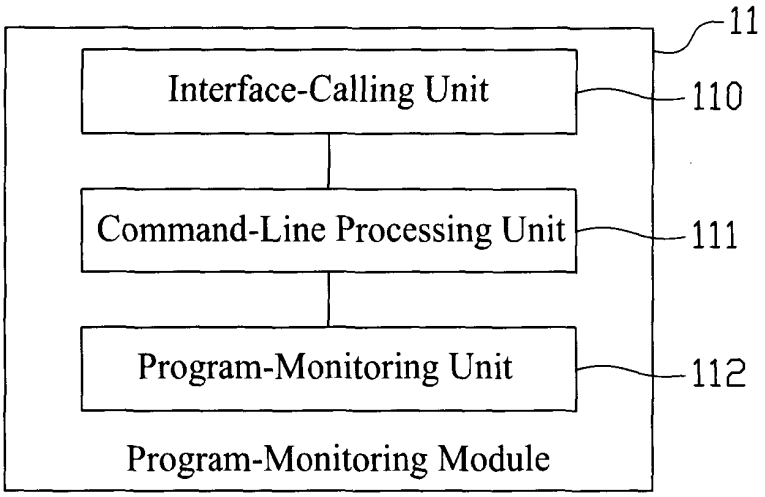

Figure 9

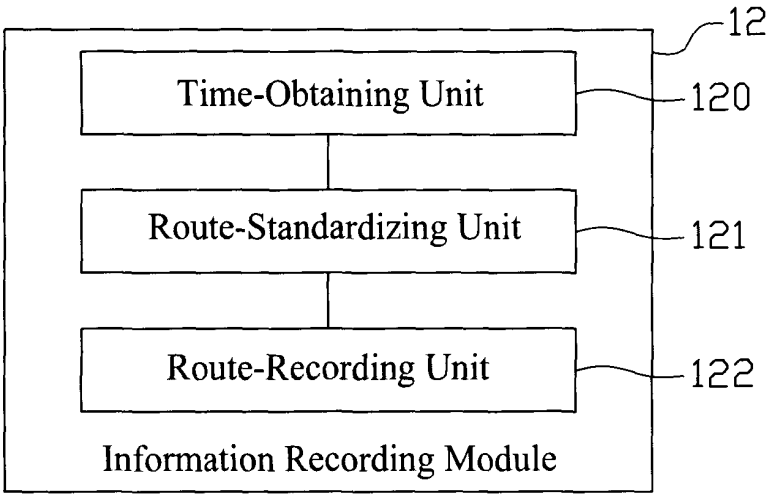

Figure 10

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

- US 2010057905 A1 **[0004]**
- US 6324546 B1 **[0004]**
- US 7925635 B1 **[0004]**