

[19] 中华人民共和国国家知识产权局

[51] Int. Cl.

G06F 9/46 (2006.01)

G06F 9/48 (2006.01)



[12] 发明专利申请公开说明书

[21] 申请号 200510004029.8

[43] 公开日 2006年7月12日

[11] 公开号 CN 1801096A

[22] 申请日 2005.1.5

[21] 申请号 200510004029.8

[71] 申请人 国际商业机器公司

地址 美国纽约

[72] 发明人 过晓冰 王哲鹏 顾光导 苏 辉

[74] 专利代理机构 中国国际贸易促进委员会专利商
标事务所
代理人 李 玲

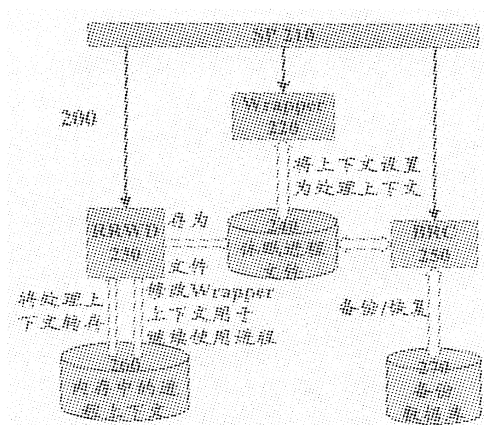
权利要求书 5 页 说明书 14 页 附图 10 页

[54] 发明名称

在计算机系统中用于对进程进行休眠的方法和系统

[57] 摘要

本发明涉及一种在计算机系统中用于单个进程休眠/唤醒的方法和系统。进程休眠包括：隐藏所述进程的所有窗口；挂起所述进程的所有线程；记录所述进程的上下文并将所记录的所述进程上下文作为一个休眠进程文件进行转存；以及终止所述进程以对所述进程休眠。唤醒休眠进程包括：构造一个与所述休眠进程的上下文结构相似的包装器；恢复所述休眠进程的上下文；唤醒所述休眠进程的所有线程；以及显示所述休眠进程的所有窗口以继续使用所述进程。根据本发明，降低了打开/关闭应用程序上的操作，优化了系统资源。并且还可用于运行时间应用程序上下文备份和恢复，自动保存当前工作环境，以及当计算机发生故障时恢复整个环境。



1、一种在计算机系统中用于对进程进行休眠的方法，包括以下步骤：

隐藏所述进程的所有窗口；

挂起所述进程的所有线程；

记录所述进程的上下文并将所记录的所述进程上下文作为一个休眠进程文件进行转存；以及

终止所述进程以对所述进程休眠。

2、根据权利要求1的用于对进程进行休眠的方法，进一步包括，将所记录的所述进程上下文转存到硬盘的步骤，其中包括以下子步骤：

将内存中记录的所述进程的用户空间中的所有内容转存到计算机的硬盘中；

将内存中记录的所述进程所占用的所有资源转存到硬盘上；

将内存中记录的所述进程的运行时间上下文转存到硬盘中。

3、根据权利要求2的用于对进程进行休眠的方法，其中，所述将内存中记录的所述进程的用户空间中的所有内容转存到计算机的硬盘中的子步骤包括：

将关于所述进程的堆栈、代码和数据，动态链接库 DLL/Lib、以及内核接口的内容转存到计算机的硬盘中。

4、根据权利要求2的用于对进程进行休眠的方法，其中，所述将内存中记录的所述进程所占用的所有资源转存到硬盘上的子步骤包括：

将所述进程的基本信息以及关于所述进程的资源描述转存到硬盘上，其中，所述进程的基本信息包括所述进程的名称、父进程、地址空间入口，以及所述进程的资源描述包括资源索引以及所述进程所占用的资源的内容。

5、根据权利要求2的用于对进程进行休眠的方法，其中，所述将内存中记录的所述进程的运行时间上下文转存到硬盘中的子步骤包

括:

将所述进程中的每个线程的线程上下文转存到硬盘中,其中所述线程上下文包括:寄存器值、定时器设置,优先级等。

6、根据权利要求1的用于对进程进行休眠的方法,进一步包括:以一定的时间间隔,将所述转存到硬盘中的所述进程的休眠进程文件自动备份到一个数据库中的步骤。

7、根据权利要求1的用于对进程进行休眠的方法,进一步包括:为用户提供已运行进程的列表,用于由用户选择待休眠的进程。

8、一种在计算机系统中用于对休眠进程进行唤醒的方法,构造一个与所述休眠进程的上下文结构相似的包装器(Wrapper),通过执行以下步骤来唤醒休眠进程,所述步骤包括:

恢复所述休眠进程的上下文;

唤醒所述休眠进程的所有线程;以及

显示所述休眠进程的所有窗口以继续使用所述进程。

9、根据权利要求8的用于对休眠进程进行唤醒的方法,进一步包括:

根据在进程休眠过程中转存到硬盘中的休眠进程文件,由所述Wrapper通过执行以下子步骤来恢复所述休眠进程的上下文,所述子步骤包括:

恢复休眠进程的基本信息;

恢复休眠进程所占用的系统资源;以及

恢复休眠进程的用户空间。

10、根据权利要求8的用于对休眠进程进行唤醒的方法,进一步包括:

根据在进程休眠过程中转存到硬盘中的休眠进程文件,由所述Wrapper通过执行以下子步骤来唤醒所述休眠进程的所有线程,所述子步骤包括:

从转存到硬盘中的休眠进程文件中读取所述休眠进程的线程数;

根据休眠进程的数目,修改所述Wrapper的线程数目。

11、根据权利要求 8 的用于对休眠进程进行唤醒的方法，还包括：
在计算机系统出现故障或异常关闭计算机系统时，调用存储在一个备份数据库中的所述休眠进程的休眠进程文件的步骤。

12、根据权利要求 8 的用于对休眠进程进行唤醒的方法，进一步包括：

为用户提供已休眠进程的列表，用于由用户选择待唤醒的进程。

13、一种在计算机系统中用于对进程进行休眠的系统，包括：

一个位于内存中的第一存储器，用于保存所述进程在内存中的进程上下文；

一个原始读/写驱动器 (RRWD)，用于与所述第一存储器进行交互，而对待休眠进程执行以下操作：

隐藏所述进程的所有窗口；

通过挂起所述进程的所有线程来挂起所述进程；

从所述第一存储器中读取记录在其中的所述进程的上下文并将所读取的所述进程上下文作为一个休眠进程文件进行转存；以及

终止所述进程以对所述进程休眠；

一个第二存储器，位于所述计算机系统的硬盘中，用于保存由所述 RRWD 从第一存储器中读取的所述进程的上下文内容。

14、根据权利要求 13 的用于对进程进行休眠的系统，其中配置所述 RRWD 以执行以下操作来将所记录的所述进程上下文转存到硬盘上，所述操作包括：

将内存中记录的所述进程的用户空间中的所有内容转存到计算机的硬盘中；

将内存中记录的所述进程所占用的所有资源转存到硬盘上；

将内存中记录的所述进程的运行时间上下文转存到硬盘中。

15、根据权利要求 13 的用于对进程进行休眠的系统，其中配置所述 RRWD 以执行以下操作来把内存中记录的所述进程的用户空间中的所有内容转存到计算机的硬盘中，所述操作包括：

将关于所述进程的堆栈、代码和数据，动态链接库 DLL/Lib、以

及内核接口的内容转存到计算机的硬盘中。

16、根据权利要求 13 的用于对进程进行休眠的系统，其中配置所述 RRWD 以执行以下操作来把内存中所述进程所占用的所有资源转存到硬盘上，所述操作包括：

将所述进程的基本信息以及关于所述进程的资源描述转存到硬盘上，其中，所述进程的基本信息包括所述进程的名称、父进程、内存空间入口地址，以及所述进程的资源描述包括资源索引以及所述进程所占用的资源的内容。

17、根据权利要求 13 的用于对进程进行休眠的系统，其中配置所述 RRWD 以执行以下操作来将所述进程的运行时间上下文转存到硬盘中，所述操作包括：

将所述进程中的每个线程的线程上下文转存到硬盘中，其中所述线程上下文包括：寄存器值、定时器设置，优先级等。

18、根据权利要求 13 的用于对进程进行休眠的系统，进一步包括：

一个备份/恢复部件（BRC），用于以一定的时间间隔，自动备份转存到硬盘中的所述进程的休眠进程文件；以及

一个备份数据库，用于保存由所述备份/恢复部件所备份的休眠进程文件。

19、根据权利要求 13 的用于对进程进行休眠的系统，进一步包括：

一个与计算机系统连接的用户接口，用于提供已运行进程列表供用户选择待休眠进程。

20、一种在计算机系统中用于对休眠进程进行唤醒的系统，包括：
一个休眠进程文件存储器，用于保存休眠进程的进程上下文；以及

一个包装器（Wrapper），用于构造与所述休眠进程的上下文结构相似的 Wrapper 上下文，并用于执行以下操作：

恢复所述休眠进程的上下文；

唤醒所述休眠进程的所有线程；以及

显示所述休眠进程的所有窗口以继续使用所述进程。

21、根据权利要求 20 的用于对休眠进程进行唤醒的系统，进一步包括：

根据在休眠进程文件存储器中所保存的休眠进程文件，由所述 Wrapper 通过执行以下操作来恢复所述休眠进程的上下文，所述操作包括：

恢复休眠进程的基本信息；

恢复休眠进程所占用的系统资源；以及

恢复休眠进程的用户空间。

22、根据权利要求 20 的用于对休眠进程进行唤醒的系统，进一步包括：

一个备份数据库，用于保存所述休眠进程的进程上下文的备份文件；

一个备份/恢复部件（BRC），用于在计算机系统出现故障或异常关闭计算机系统时，调用存储在所述备份数据库中的所述休眠进程的备份休眠进程文件。

23、根据权利要求 22 的用于对休眠进程进行唤醒的系统，进一步包括：

一个与计算机系统连接的用户接口，用于提供已休眠进程列表供用户选择待唤醒进程。

在计算机系统中用于对 进程进行休眠的方法和系统

技术领域

本发明一般涉及计算机系统的数据处理技术，特别地，本发明涉及在计算机系统中用于对进程进行休眠的方法和系统，以及对已休眠进程进行唤醒的方法和系统。

背景技术

随着信息技术的快速发展，人们在个人计算机(PC)上运行各种应用程序时，时常会同时打开多个应用程序，显然，这将占用许多系统资源而且会降低系统的性能。即使某些应用程序在系统中处于挂起状态，但是这些应用程序仍旧占用了相关的文件(句柄)资源、硬件/端口/网络/图象资源，这仍会降低计算机的性能。另外，当人们在重新打开应用程序以继续以前的工作时，需要采取一系列的动作，例如在文字编辑时，除了打开编辑软件，还需要选择并打开被编辑的文件，并通过翻页/滚动拖动条定位到上次的编辑对象上，然后才能继续工作，因此需要提供一种可针对单个进程，例如一个已经打开的应用程序，进行休眠的技术方案来解决上述这些问题。

为解决这些问题，现有技术中提供了一些技术解决方案。其中，就本发明的技术人员所了解的，在一种 Palm 操作系统中，使用一种特定的硬件平台（例如，能够被用作内存和硬盘的公共存储器），能将运行的应用程序冻结。但是，在个人计算机（PC）平台，在当前的硬件条件和操作系统下仍旧存在上述问题。

在其他领域曾使用了一些解决方案，其类似于本发明的解决方案。举例来说，以下是一些相关技术和应用程序的示例：

- 1、进程管理应用，诸如 Windows2000/XP 中的 taskmgr.exe。它

能修改进程的工作优先级，但是修改的范围仅限于六个级别：实时（**realtime**）、高（**high**）、高于正常（**above normal**）、正常（**normal**）、低于正常（**below normal**）、以及低（**low**）。这种技术不能完全中止进程、从而回收所有进程占用的资源。

2、一些软件能保存最后工作的场景（**scenario**），例如 **winamp3.exe**。但是，这种技术需要由软件开发者来实施，并且现在大多数的软件都不能支持这个特性。

3、一些集群（**cluster**）/并行系统支持进程移植技术，例如 **MOSIX** 系统。但是这种技术需要完全修改系统内核并且仅能在多计算机系统上实施。

4、备份/恢复应用程序。现在的备份/恢复应用程序只能处理静态的文件镜像。

5、系统休眠技术，这种技术能快照（**snapshot**）整个内存的镜像并将所述镜像保存到硬盘上。但是，系统休眠针对的是整个计算机系统，它不能对计算机系统中运行的单个的进程进行休眠。并且，系统休眠技术还依赖于特定的硬件平台，例如计算机的电源管理必须支持 **ACPI**（高级配置和功率管理接口 **Advanced configuration and Power Interface**）等。

由以上对现有技术描述可知，在现有技术中，在提高系统资源利用率、改善计算机性能方面，现有的休眠技术只针对的是整个计算机系统，而没有提供管理单个进程的能力，因此特别对于正在运行的系统而言，并没有为其提供高的运算性能。

发明内容

为解决现有技术中存在的问题，提出了本发明的用于对进程进行休眠的方法和系统。根据本发明，可以简化用户在打开/关闭应用程序时的操作，并且在系统高负荷下提高系统的性能，而且也提供了一种恢复实时工作环境的方法。

根据本发明的一个方面，提供了一种新型的技术方案来休眠和重

新使用（唤醒）进程（运行的应用程序）。

根据本发明的一个方面，提供了一种在计算机系统中用于对进程进行休眠的方法，包括以下步骤：隐藏所述进程的所有窗口；挂起所述进程的所有线程；记录所述进程的上下文并将所记录的所述进程上下文作为一个休眠进程文件进行转存；以及终止所述进程以对所述进程休眠。

根据本发明的另一个方面，提供了一种在计算机系统中用于对休眠进程进行唤醒的方法，构造一个与所述休眠进程的上下文结构相似的包装器（Wrapper），通过执行以下步骤来唤醒休眠进程，所述步骤包括：恢复所述休眠进程的上下文；唤醒所述休眠进程的所有线程；以及显示所述休眠进程的所有窗口以继续使用所述进程。

根据本发明的另一个方面，提供了一种在计算机系统中用于对进程进行休眠的系统，包括：一个位于内存中的第一存储器，用于保存所述进程在内存中的进程上下文；一个原始读/写驱动器（RRWD），用于与所述第一存储器进行交互，而对待休眠进程执行以下操作：隐藏所述进程的所有窗口；通过挂起所述进程的所有线程来挂起所述进程；从所述第一存储器中读取记录在其中的所述进程的上下文并将所读取的所述进程上下文作为一个休眠进程文件进行转存；以及终止所述进程以对所述进程休眠；一个第二存储器，位于所述计算机系统的硬盘中，用于保存由所述 RRWD 从第一存储器中读取的所述进程的上下文内容。

根据本发明的另一个方面，提供了一种在计算机系统中用于对休眠进程进行唤醒的系统，包括：一个休眠进程文件存储器，用于保存休眠进程的进程上下文；以及一个包装器（Wrapper），用于构造与所述休眠进程的上下文结构相似的 Wrapper 上下文，并用于执行以下操作：恢复所述休眠进程的上下文；唤醒所述休眠进程的所有线程；以及显示所述休眠进程的所有窗口以继续使用所述进程。

本发明还提供一种触发进程休眠/唤醒/备份/恢复的用户接口。

根据本发明的在计算机系统中用于对进程进行休眠以及对休眠

进程进行唤醒的方法和系统，降低了打开/关闭应用程序上的操作，并能通过对运行的应用程序执行休眠来释放系统资源从而提供高的性能。并且本发明还可用于运行时间应用程序上下文备份和恢复，本发明可以自动保存当前的工作环境，以及当计算机发生故障时能恢复整个环境。

附图说明

以下结合附图，对本发明的具体实施方式进行详细的说明，由此，本发明的特点、优点、目的和有益效果将会变得更明显，其中：

图1示意性示出了根据现有技术中在进程与操作系统中的内核之间的结构的图；

图2示意性示出了根据本发明的进程休眠系统的结构图；

图3示意性示出了在休眠进程的过程中转存单个进程上下文的过程的示意图；

图4示意性示出了根据本发明的对单个进程执行休眠处理的流程图；

图5示意性示出了根据本发明在休眠进程唤醒过程中，Wrapper恢复休眠进程的基本信息的步骤中，进程1和Wrapper的上下文的示意图；

图6示意性示出了根据本发明在休眠进程唤醒过程中，Wrapper恢复休眠进程所占用的系统资源的步骤中，进程1和Wrapper的上下文的示意图；

图7示意性示出了根据本发明在休眠进程唤醒过程中，Wrapper恢复休眠进程的用户空间内容的步骤中，进程1和Wrapper的上下文的示意图；

图8示意性示出了根据本发明在休眠进程唤醒过程中，Wrapper恢复休眠进程的所有线程上下文的步骤中，进程1和Wrapper的上下文的示意图；

图9示意性地示出了根据本发明的进程休眠过程的流程图；

图 10 示意性地示出了根据本发明的休眠进程唤醒过程的流程图；
图 11 示意性地示出了根据本发明的进程备份过程的流程图；以
及
图 12 示意性地示出了根据本发明的进程恢复过程的流程图。

具体实施方式

以下结合附图对本发明的优选实施方式进行描述，应该理解，这里描述的实施例仅是例示的作用，本领域技术人员在本发明的教导下，还可以对本发明进行修改、变换，而这些修改或变换均不会脱离本发明的原理。

图 1 示意性示出了根据现有技术中在进程与操作系统中的内核之间的结构的图。现有计算机系统中，通常会有多个应用程序被同时运行，运行的应用程序(进程)与计算机系统的内核进行交互，从而完成数据的传输、处理等交互操作。

在附图 1 中，示意性示出了一个单个进程，例如进程 1。该进程上下文 (Process Context) 被分成两个部分，一个是在用户模式中的用户空间，另一个是在内核模式中的进程记录。

在图 1 中所示出的系统中，在用户模式中，例如进程 1 的用户空间中，包括：进程 1 的堆栈、代码以及数据等信息；一个或多个动态链接库/函数库 (DLL/Lib) 110 (Dynamic-Link Library)；一个与内核相连接的内核接口 120，通过该内核接口 120，每个进程都可以与操作系统中的进程记录部分 130 进行交互。在附图 1 中所示出的内核模式中，包括：一个进程记录部分 130，其中记录了多个进程，例如进程 1，进程 2.....进程 n 中的每一个的进程记录。在图 1 中所示出的进程 1 的进程记录中，包括进程 1 的资源索引 131、进程 1 的基本信息 132，以及进程 1 的运行时间上下文 133。具体来讲，进程 1 的基本信息 132，例如包括进程 1 的名称、父进程、内存数据库地址等；进程 1 的资源索引 131，例如包括由进程所占用的所有资源，其通过与系统资源部分 135 进行交互来获得所述有关资源占用的资源索引信息；运行时间

上下文 133 中记录了进程 1 中的每个线程 TC 1, TC 2,, TC m 的线程上下文, 例如寄存器值、定时器设置, 优先级等等。此外, 在内核模式中, 还包括一个内核部分 140 和一些驱动程序 150, 其中在内核部分 140 中例如进一步可以包括文件系统管理 141, I/O 管理 142, 内存/虚拟内存管理 143...进程/线程管理 144 等功能部件。以上对涉及本发明的现有技术的相关部分进行了描述, 应该知道, 为了说明的目的, 附图 1 中示出的系统做了一些简化。

根据本发明, 提出了一种可对单个进程进行休眠处理的方法和相应的系统。参考附图 2, 其中示意性示出了根据本发明的进程休眠/唤醒系统的结构图。

在根据本发明的进程休眠/唤醒系统 200 中, 主要包括: 原始读/写驱动器 (RRWD) 230; 包装器 (Wrapper) 240; 系统界面补丁部件 (Shell patch SP) 210; 优选地, 还可以包括备份/恢复部件 (BRC) 250。此外在进程休眠系统中, 还包括一个第一外部存储器 240 (例如, 支持普通文件系统的硬盘, 或硬盘上的存储器), 用于保存休眠进程的文件; 一个内部存储器 260 (内存), 其中保存了供中央处理器(CPU)调用的进程上下文数据; 以及一个外部备份存储器 270 作为备份数据库(例如, 支持普通文件系统的硬盘), 用于保存备份的休眠进程的文件。

以下具体描述根据本发明的进程休眠/唤醒系统 200 中各个部件的功能及所实现的相应的方法步骤。具体地, 以下以进程 1 为例来描述根据本发明的对单个进程的休眠处理方法和相应的系统以及对休眠的进程唤醒以继续使用该方法的方法和相应的系统。类似地, 根据本发明的方法和系统可以实现对多个进程同时、或顺序实现进程休眠处理以及相应的进程唤醒处理。

有三类数据需要进行处理。RRWD 将进程上下文从内存转存到硬盘(存为文件), 并且, 修改进程内容以继续使用休眠的进程。BRC 将使用备份数据库来备份/恢复休眠的进程。

I、原始读/写驱动器 (Raw Read/Write Driver RRWD)

根据本发明设计的所述原始读/写驱动器 230 是一个内核驱动器，用于读取/修改所有用户进程/线程的全部地址空间并挂起/唤醒所有用户进程/线程。所述原始读/写驱动器 230 与其中存储了进程上下文的内部存储器 260 交互作用，一方面，在将单个进程进行休眠操作时，将存储器 260 中所保存的进程上下文数据读出，并作为一个休眠进程的文件转存到硬盘上的存储器 240 中，另一方面，在对休眠进程执行唤醒操作时，对在存储器 260 中的 Wrapper 上下文进行修改以用于休眠进程的唤醒操作，如图 2 所示。

特别地，在执行根据本发明的对单个进程执行休眠处理时，RRWD 230 能将每个进程，例如进程 1 在内存中的所有上下文通过执行以下步骤而作为一个休眠进程 1 的文件转存到计算机系统的硬盘中。参见附图 3，其中示意性示出了在休眠进程的过程中，转存要休眠的进程 1 的上下文的过程的示意图：

(1) RRWD 230 将内存中的进程 1 用户空间中的所有内容分别转存到计算机的硬盘中。在进程 1 的用户空间中的所有内容包括：进程 1 的堆栈、进程 1 的代码、进程 1 的数据以及进程 1 在内存中对应的地址；动态链接库 DLL/Lib、内核接口。

(2) RRWD 记录由进程 1 所占用的所有资源并将其转存到硬盘上。其中，由进程 1 所占用的所有资源包括：进程 1 的基本信息，例如包括进程 1 的名称、父进程等，以及关于进程 1 的资源描述（例如资源索引以及所占用的资源的内容）。

(3) RRWD 将进程 1 的运行时间上下文转存到硬盘。其中，进程 1 的运行时间上下文中包括了进程 1 中的每个线程 TC 1, TC 2, ……，TC m 的线程上下文，例如寄存器值、定时器设置，优先级等等。

应该知道，对以上三个部分的数据的转存不存在时间上的顺序，可以依次执行，也可以同时执行。

如图 3 所示，当对单个进程进行休眠的过程中，将内存中进程 1 的上下文转存到硬盘的步骤以后，会在硬盘中存在一个休眠进程 1 的

文件。该休眠进程 1 的文件中包括：

(1) 关于进程 1 基本信息和资源描述的数据，其中，进程 1 的基本信息中记录了进程 1 的名称、父进程、内存数据库地址等；而进程 1 的资源描述数据包括了资源索引以及所占用的资源的内容；

(2) 关于进程 1 的运行时间上下文数据，其中，进程 1 的运行时间上下文数据包括了进程 1 中的每个线程 TC 1, TC 2, ……., TC m 的线程上下文，例如寄存器值、定时器设置，优先级等等；

(3) 关于进程 1 的用户空间描述的数据，其中，如上所述，记录了进程 1 在其用户空间的所有内容及相关联的地址。

此外，当根据本发明对休眠的进程进行唤醒的过程中，RRWD 能够修改 Wrapper 的上下文。其通过对内存进行直接修改而将 Wrapper 的运行时间上下文恢复为休眠进程。详细的过程描述在参照 Wrapper 部分的描述进行详细说明。

II、Wrapper

Wrapper 是一种用于执行对休眠进程的处理的专用应用程序，其被设计用于部分修改其上下文作为目标进程、其中包括进程名称、用户空间，系统资源、线程数；以及然后利用 RRWD 修改剩余的信息，包括线程上下文、父进程、优先级，以及最后从用户空间去除 Wrapper 代码来实现对休眠进程的唤醒处理。Wrapper 将读取休眠进程的转存文件并在 Wrapper 运行时，将其自身的上下文修改为休眠进程的上下文。

Wrapper 是在用户希望恢复使用休眠进程的时候运行的。如图 2 所示，Wrapper 与在硬盘中的存储器 240 进行交互，通过访问存储器 240 中所保存的休眠进程文件，而将 Wrapper 的上下文设置为休眠进程的上下文。其工作过程通过参见附图 4 进行如下所述的详细描述：

参见附图 4，其中示出了当将 Wrapper 加载到内存时，Wrapper 进程在内存中的上下文镜像。此处，为简明起见，将根据本发明构造的 Wrapper 进程的堆栈、代码和数据在一个单个的方框中示出，并且

在这里还假设 **Wrapper** 只有一个线程。

类似地，在附图 4 中示出的 **Wrapper** 上下文中，可以看到，在用户模式中，**Wrapper** 的用户空间也包括：**Wrapper** 的堆栈、代码和数据；以及与内核之间的内核接口。而 **Wrapper** 在内核模式中，包括了 **Wrapper** 所占用的资源、所占用资源的索引、基本信息和运行时间上下文。

(1) 恢复休眠进程的基本信息

Wrapper 首先根据在对单个进程休眠的过程中转存到硬盘中的休眠进程的文件来恢复出进程 1 的基本信息。也就是说，根据硬盘中转存的休眠进程文件中的休眠进程 1 的内容，将 **Wrapper** 的基本信息修改为进程 1 的内容，例如进程名称。

参见附图 5，其中示意性示出了根据本发明在休眠进程唤醒过程中，**Wrapper** 恢复休眠进程的基本信息的步骤中，进程 1 和 **Wrapper** 的上下文的示意图。

通过这个步骤，在比较附图 4 与附图 5 之后可以看到：通过首先恢复出进程 1 的基本信息，附图 4 中所示出的初始 **Wrapper** 上下文变成当前附图 5 中所示出的进程 1&**Wrapper** 的上下文。

(2) 恢复休眠进程所占用的系统资源

Wrapper 从在对单个进程休眠的过程中转存到硬盘中的休眠进程的文件中，查询由休眠进程所占用的全部系统资源并且重新应用所述休眠进程的系统资源。

参见附图 6，其中示意性示出了根据本发明在休眠进程唤醒过程中，**Wrapper** 恢复休眠进程所占用的系统资源的步骤中，进程 1 和 **Wrapper** 的上下文的示意图。

通过这个步骤，在比较附图 5 与附图 6 之后可以进一步得到：在附图 6 中所示出的进程 1&**Wrapper** 的上下文中，初始 **Wrapper** 所占用的系统资源变成了休眠进程 1 所占用的系统资源，以及初始 **Wrapper** 中的资源索引也变成了休眠进程 1 的资源索引。

此外，优选地，如果当前系统不能满足全部资源应用程序的需求，

则提示用户故障并退出对休眠进程的唤醒过程。

(3) 恢复休眠进程的用户空间

Wrapper 找出休眠进程的用户空间中的空闲地址空间, 并且将当前的代码移动到空闲地址空间。

除内核接口模块以外, 从在对单个进程休眠的过程中转存到硬盘中的休眠进程的文件中, 读取休眠进程的用户空间中的所有内容并覆盖到 **Wrapper** 用户空间的对应地址上。

参见附图 7, 其中示意性示出了根据本发明在休眠进程唤醒过程中, **Wrapper** 恢复休眠进程的用户空间内容的步骤中, 进程 1 和 **Wrapper** 的上下文的示意图。

通过这个步骤, 在比较附图 6 与附图 7 之后可以进一步得到: 在附图 7 中所示出的进程 1 & **Wrapper** 的上下文中, 休眠进程 1 在用户模式下的大部分内容, 例如进程 1 的堆栈、代码、数据, 以及进程 1 的 DLL/Lib 的内容都得到了恢复。

(4) 恢复休眠进程的线程上下文

在这个步骤中, **Wrapper** 从对单个进程休眠的过程中转存到硬盘中的休眠进程的文件中, 读取休眠进程 1 的线程数目, 并且启动相同的线程数目。

然后, **Wrapper** 向系统申请挂起所有线程, 包括主要的线程。

在这个步骤之后, **Wrapper** 的上下文被部分恢复成休眠进程的上下文。然后, **RRWD** 把每个线程的上下文修改为休眠线程的上下文, 并且恢复一些关键信息, 例如父进程等。

最后的步骤是从用户空间去除 **Wrapper** 的内容。此时, 原来 **Wrapper** 的进程上下文已经完全被进程 1 的上下文所替换, 也即, 休眠进程上下文被完全恢复。如果 **RRWD** 唤醒进程 1 的所有线程, 则休眠进程将得以继续使用。

参见附图 8, 其中示意性示出了根据本发明在休眠进程唤醒过程中, **Wrapper** 恢复休眠进程的所有线程上下文的步骤中, 进程 1 和 **Wrapper** 的上下文的示意图。

通过这个步骤，在比较附图 7 与附图 8 之后可以进一步得到：在附图 8 中所示出的进程 1&Wrapper 的上下文中，由 Wrapper 初始只具有一个线程而恢复到实现休眠进程 1 所有线程得到完全恢复。也就是说，在完成附图 8 的步骤后，实现了对休眠进程的唤醒以及可以使休眠进程在得到唤醒后得以继续使用。

III 备份/恢复部件 (Backup & Restore Component BRC)

在根据本发明的一种优选实施方式中，还可以包括一个备份/恢复部件 (BRC)，用于以一定的时间周期自动保存所有进程上下文，以便在计算机系统出现故障或异常关闭的情况下，能利用所保存的进程的最近上下文来恢复进程或休眠进程的内容。该部件可以通过定时器或用户首选项 (preference) 来进行触发，例如，它将每隔五分钟自动调用 RRWD 以保存所有进程上下文。当计算机启动时，如果计算机最后的关闭是异常的，例如计算机崩溃并重启时，则它可以根据用户的设定，恢复最后保存的进程所有上下文。

如图 2 所示，BRC 与存储器 240 进行交互，以每隔一定时间间隔将在进程休眠处理过程中转存到硬盘中的休眠进程文件自动保存到一个备份数据库 270 中。

IV. Shell Patch (系统界面补丁) (SP)。

SP 是用于 Linux (例如 Gnome) 或 Windows Explorer (视窗浏览器) 中的回调函数 (call back) 功能的桌面平台的补丁 (patch)。它集中在用户接口 (UI) 设计上：能在用户输入热键 (例如“Alt-F5”) 时，触发用于当前进程的休眠处理，并且如果用户输入另一个热键 (例如“Ctrl-Tab”) 则可以列出硬盘中的所有休眠进程。其在 Shell 中增加一个将触发继续使用指定进程的上下文菜单项。

再次参见附图 2，SP 是提供进程休眠系统与操作系统之间交互的接口。通过所述 SP，用户可以选择待休眠的进程或者待唤醒的进程；当用户选择某待休眠进程时，SP 通过调用 RRWD 对进程进行休眠；当用户选择待唤醒进程时，SP 通过选择 Wrapper 和 RRWD 对休眠进程进行唤醒以继续使用，当需要备份所有进程上下文时，SP 通过选择

BRC 来备份进程等。

以下通过参考附图 2 并结合附图 9-12 的流程图来描述根据本发明的进程休眠/唤醒过程的具体步骤。

附图 9 示意性地示出了根据本发明的进程休眠过程的流程图。参见附图 9, 首先当启动对单个进程的进程休眠过程时, 在步骤 S905 中, **RRWD** 将隐藏 (最小化) 指定要进行休眠的进程的所有窗口。然后, 在步骤 S910 中, **RRWD** 通过挂起该进程的所有线程而将该指定进程挂起。接着在步骤 S915 中, **RRWD** 将该指定休眠的进程的上下文转存到计算机的硬盘中。具体地, 在该步骤中, **RRWD** 将以下指定休眠进程的上下文内容转存到计算机的硬盘中: (1) 将内存中指定进程的用户空间中的所有内容转存到计算机的硬盘中; 例如, 指定进程的用户空间中的所有内容包括: 指定进程的堆栈、代码、数据; 动态链接库 **DLL/Lib**、内核接口。(2) 查询由指定进程所占用的所有资源并将其转存到硬盘上。其中, 由指定进程所占用的所有资源包括: 指定进程的基本信息, 例如包括指定进程的名称、父进程、地址空间入口等, 以及关于指定进程的资源描述, 例如资源索引以及所占用的资源的内容。(3) 将指定进程的进程状态信息, 例如运行时间上下文转存到硬盘。其中, 所述运行时间上下文中包括了进程中的每个线程 **TC 1**, **TC 2**,, **TC m** 的线程上下文, 例如寄存器值、定时器设置, 优先级等等。最后, 在步骤 S920 中, 终止该进程, 即进程休眠成功, 以及释放该指定进程所占用的所有资源。

附图 10 示意性地示出了根据本发明的休眠进程唤醒过程的流程图。参见附图 10, 休眠进程唤醒过程的简单工作过程如下所示。在步骤 S1005 中, **Wrapper** 将恢复休眠进程上下文。在所述恢复休眠进程上下文的步骤中, 进一步包括以下子步骤: (1) 恢复休眠进程的基本信息。也就是说, 根据硬盘中转存的休眠进程文件中的休眠进程的内容, 修改其基本信息, 例如进程名称; (2) 恢复休眠进程所占用的系统资源; (3) 恢复休眠进程的用户空间; (4) 恢复运行时间上下文,

包括每个线程的上下文；(5)恢复其他的基本信息，例如父进程（如果可用的话）；以及(6)RRWD通过释放Wrapper所占用的地址区域而从用户空间去除Wrapper代码。然后，在成功实现所述恢复休眠进程上下文的步骤之后，在步骤S1010中，RRWD将唤醒该休眠进程。以及，最后在步骤S1015中，RRWD将显示休眠进程的所有窗口。由此，成功实现了对休眠进程的唤醒。

附图11、12示意性地示出了根据本发明的进程备份/恢复过程的流程图。该进程备份/恢复过程优选地是进程休眠过程的附加和扩展功能，其在进行进程休眠的过程中，自动将在进程休眠过程中转存到硬盘上的内容按一定的时间间隔进行自动备份，以及在计算机系统出现故障或异常关闭的情况下，能利用所保存的进程的最近上下文来恢复进程或休眠进程的内容。

参见附图11，进程备份的简单工作过程如下所示：

首先，在步骤S1105中，执行对进程休眠过程的调用。在该步骤中，实际上是对进程执行了进程休眠全部过程（图9描述）。接下来，在步骤S1110中，BRC将在硬盘上保存的进程转存文件和由进程打开的所有文件备份到一个非易失性存储器中。以及，在计算机运行出现故障或异常关闭的情况下，在步骤S1115中，调用进程唤醒过程，即图10描述的过程，由此成功实现备份。

附图12示意性地示出了根据本发明的进程恢复过程的流程图。参见附图12，进程恢复的简单进程过程如下所示：

首先在步骤S1205中，BRC将从备份数据库中复制所有文件到原始位置，其将覆盖现有文件。也就是说，恢复硬盘上的进程转存文件和该进程打开的所有文件。

然后，在步骤S1210中，BRC通过调用进程唤醒过程，即图10描述的过程，来恢复进程。

以上结合附图，对根据本发明针对单个进程的进程休眠/唤醒的方法和系统的各个实施例进行了详细描述，但是以上这些实施例仅是

示例的，本领域技术人员可以在本发明的精神和范围内作出各种变化和修改。因此，本发明不限于这些实施例，本发明的范围由随附权利要求限定为准。

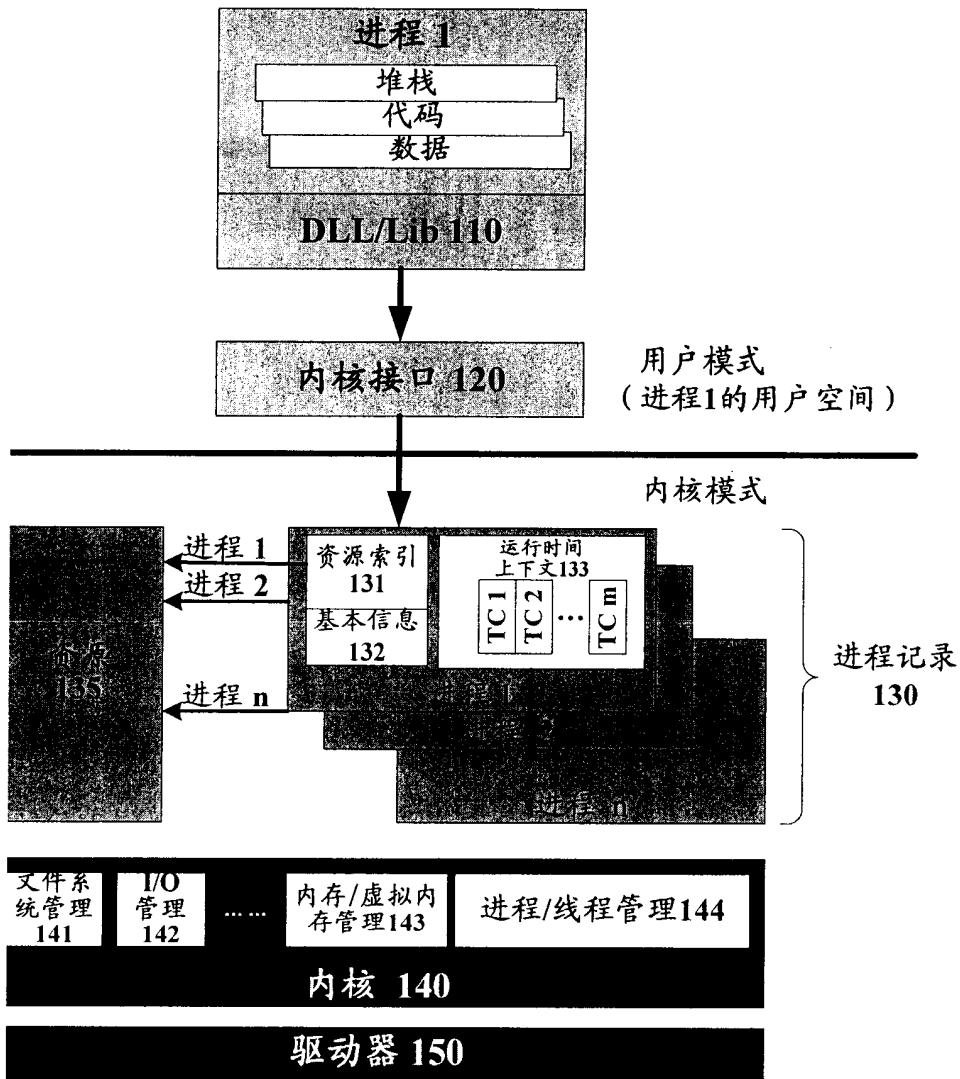


图 1

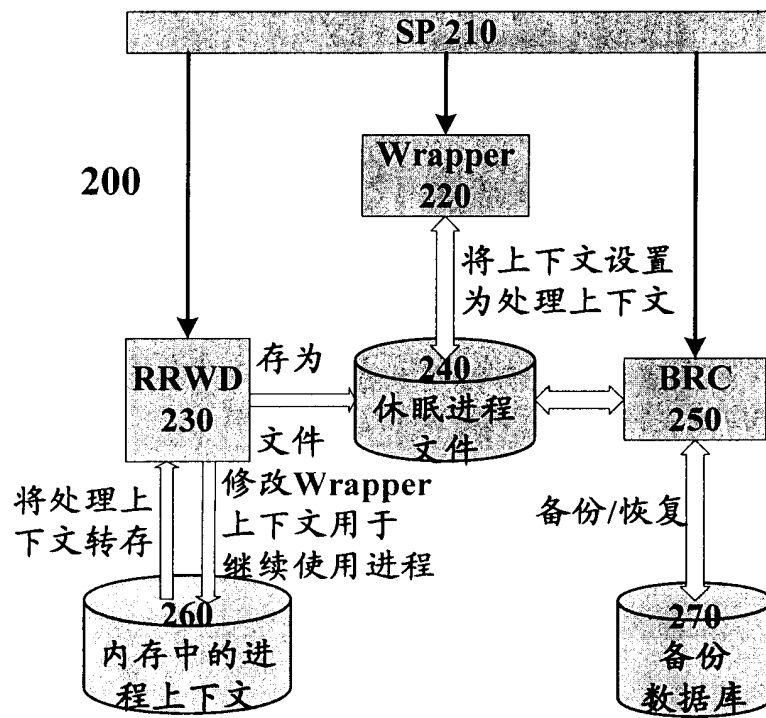


图 2

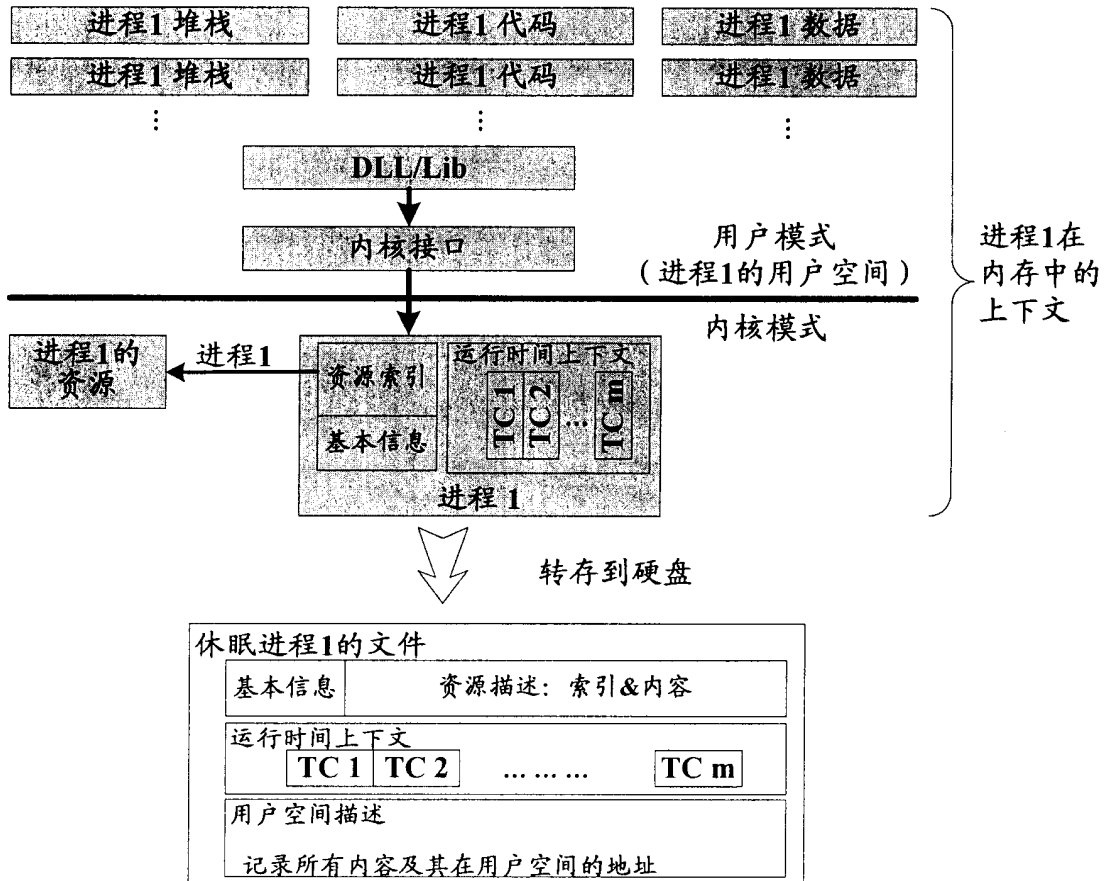


图 3

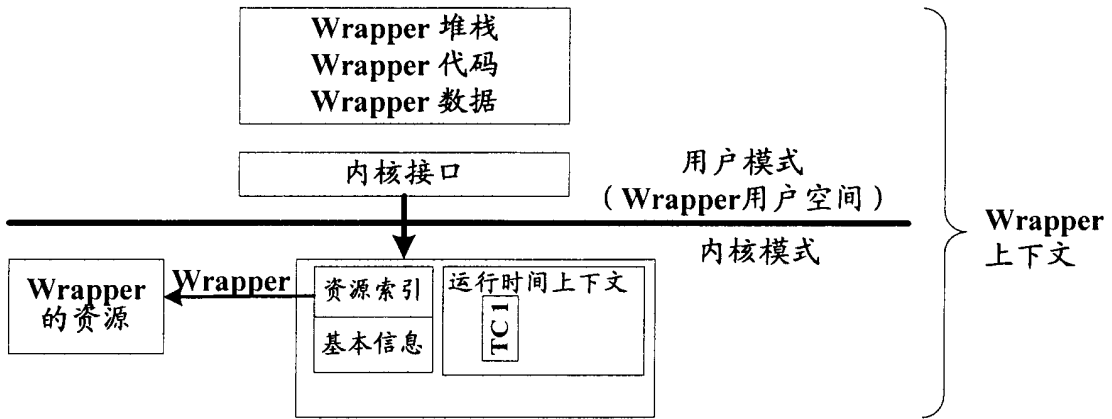


图 4

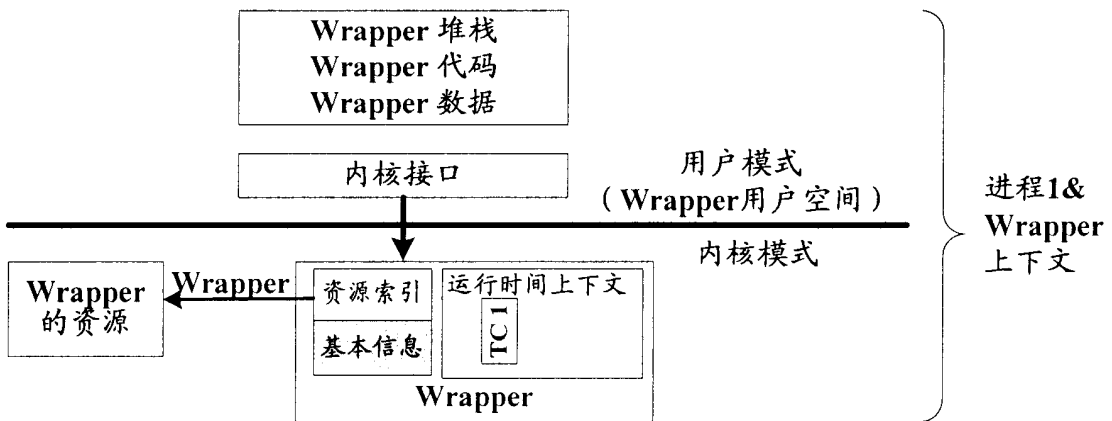


图 5

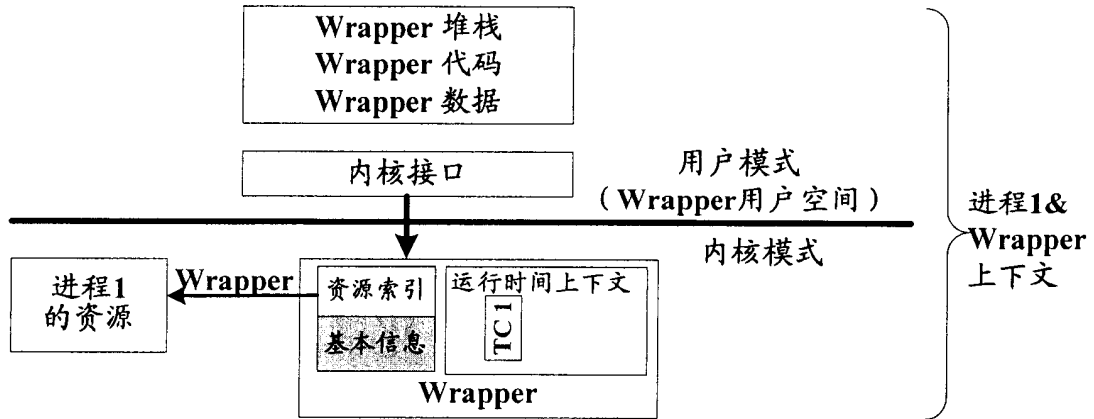


图 6

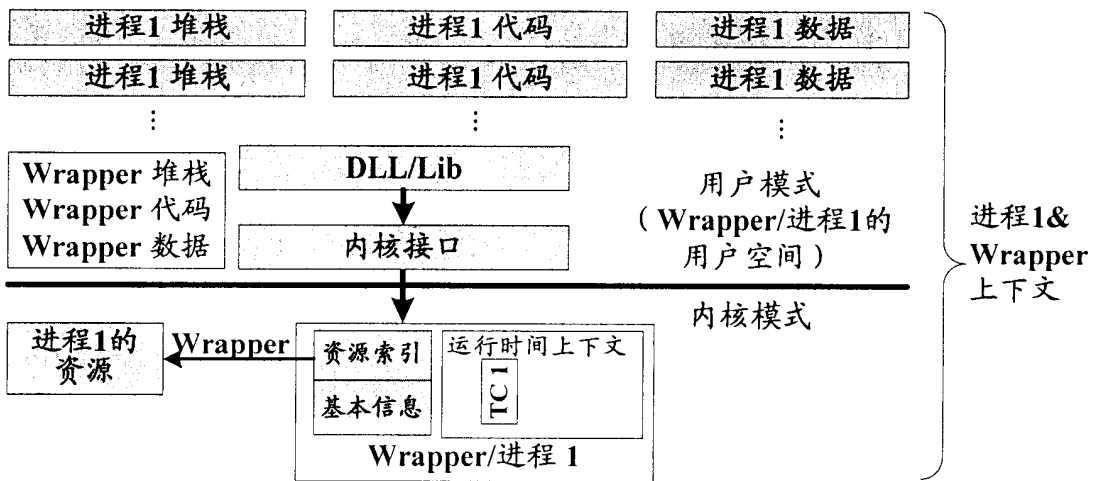


图 7

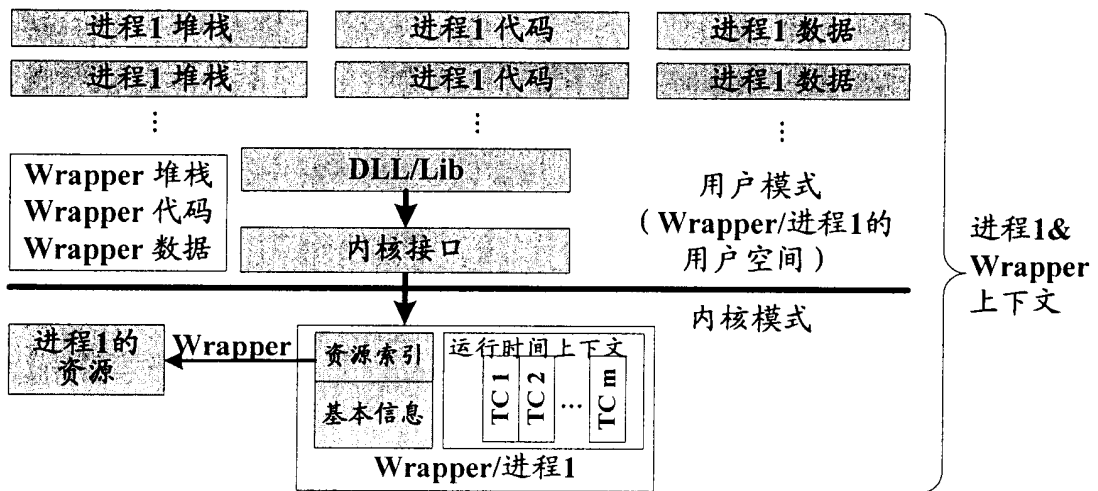


图 8

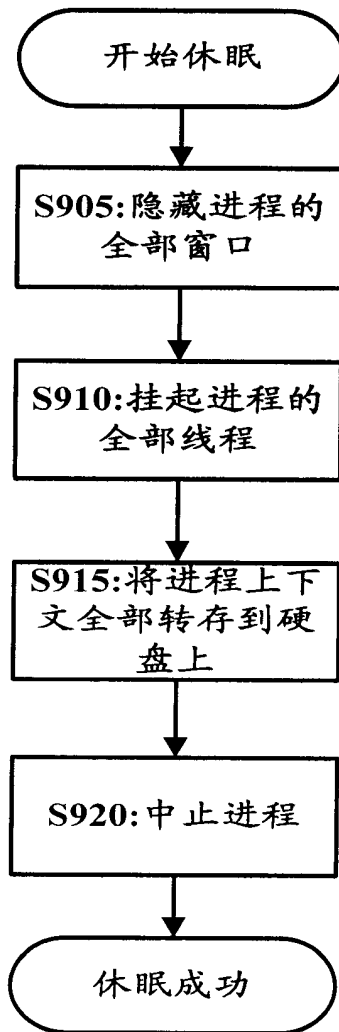


图 9

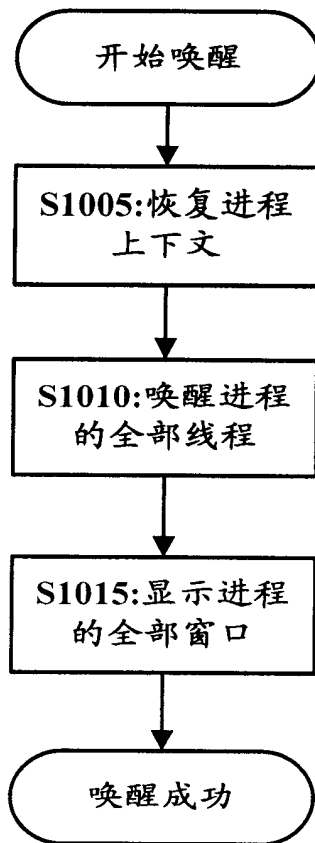


图 10

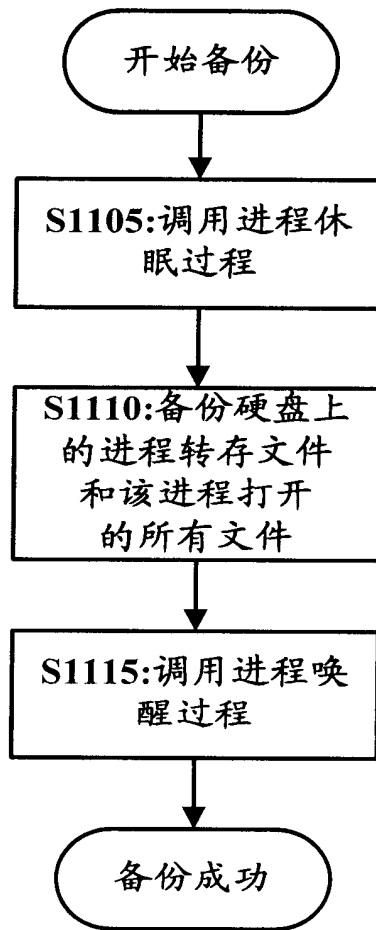


图 11

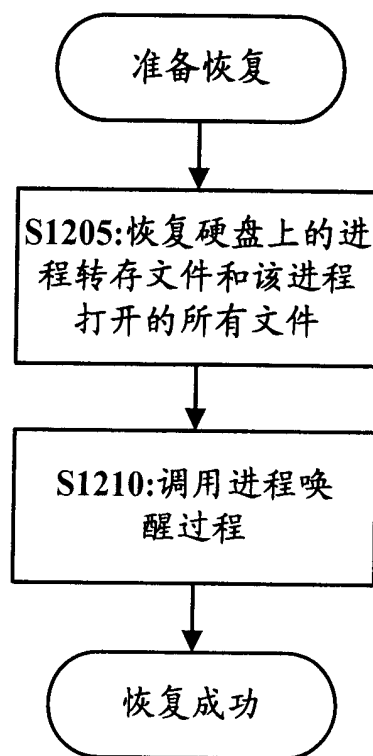


图 12