US 20140025868A1

(19) **United States**

(12) **Patent Application Publication**

(10) **Pub. No.: US 2014/0025868 A1**

(43) **Pub. Date: Jan. 23, 2014**

**Furuhjelm**

(54) **SYSTEM AND METHOD FOR MANAGING STORAGE DEVICE CACHING**

(71) Applicant: **Seagate Technology LLC**, Cupertino, CA (US)

(72) Inventor: **Martin R. Furuhjelm**, Auburn, CA (US)

(73) Assignee: **Seagate Technology LLC**, Cupertino, CA (US)

(21) Appl. No.: **13/953,395**

(22) Filed: **Jul. 29, 2013**

**Related U.S. Application Data**

(63) Continuation of application No. 12/508,895, filed on Jul. 24, 2009, now Pat. No. 8,499,120.

**Publication Classification**

(51) **Int. Cl.**

***G06F 12/08*** (2006.01)

***G06F 12/02*** (2006.01)

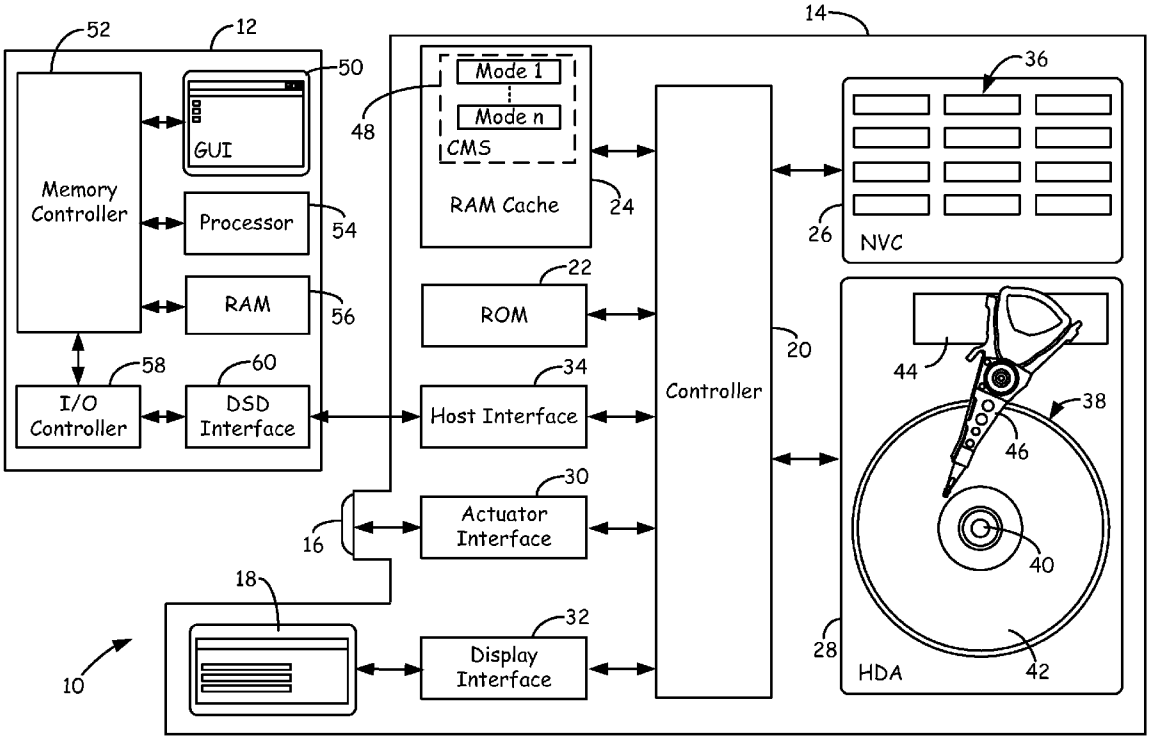
(52) **U.S. Cl.**

CPC ..... ***G06F 12/0868*** (2013.01); ***G06F 12/0246*** (2013.01)

USPC ..... **711/103**

(57) **ABSTRACT**

Systems and methods are disclosed for caching data according to user-selected caching modes. In an embodiment, an apparatus may comprise a controller configured to pin data from a non-volatile storage medium to a data cache based on at least one user-selectable caching mode. In another embodiment, a device may comprise a processor configured to display a graphical user interface (GUI) at a display, and receive, from the GUI, an indication of a user selection of a data caching mode that determines which data to pin from a non-volatile memory of a data storage device to a data cache of the data storage device. The processor may be configured to then send an indication of the user selection of the data caching mode to the data storage device.



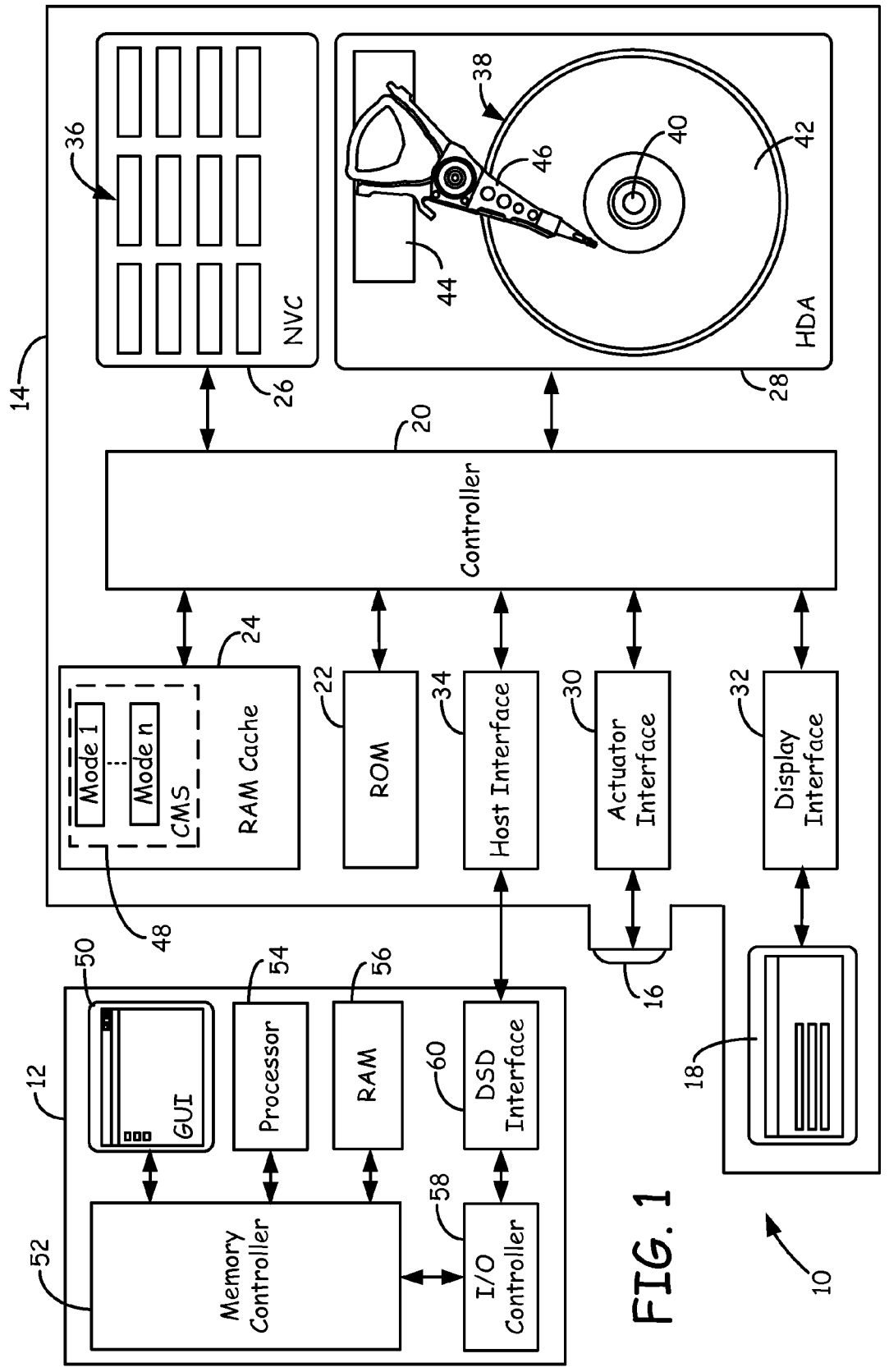

14
36
NVC
26
38
40
42
46
44
HDA
28
20
Controller
24
Mode 1
Mode n
CMS
RAM Cache
22
ROM
34
Host Interface
30
Actuator Interface
32
Display Interface
48
50
GUI
54
Processor
56
RAM
60
DSD Interface
16
18
12
52
Memory Controller
58
I/O Controller
10

FIG. 1

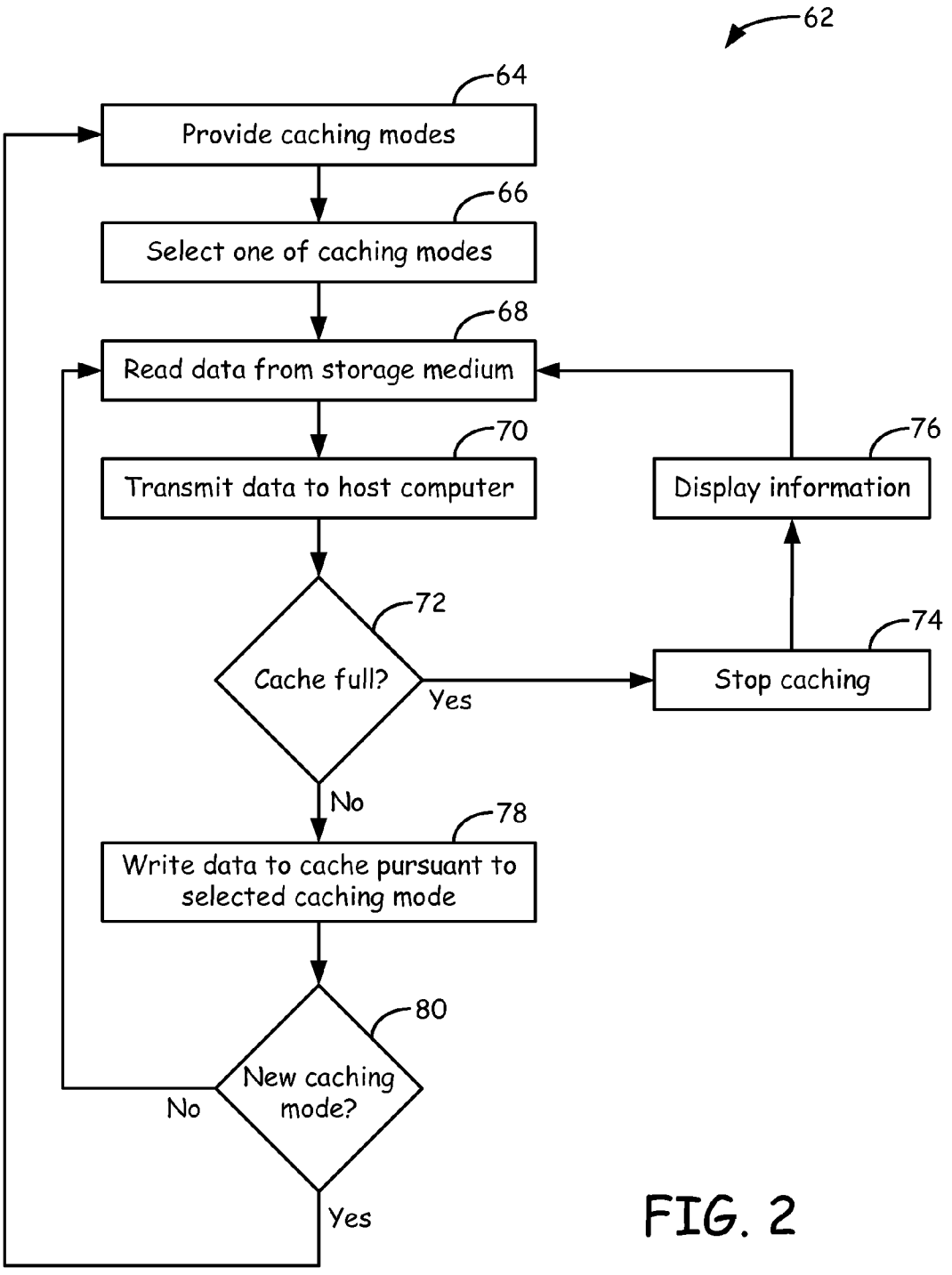

62
64
Provide caching modes
66
Select one of caching modes
68
Read data from storage medium
70
Transmit data to host computer
72
Cache full?
Yes
No
74
Stop caching
76
Display information
78
Write data to cache pursuant to selected caching mode
80
New caching mode?
No
Yes

FIG. 2

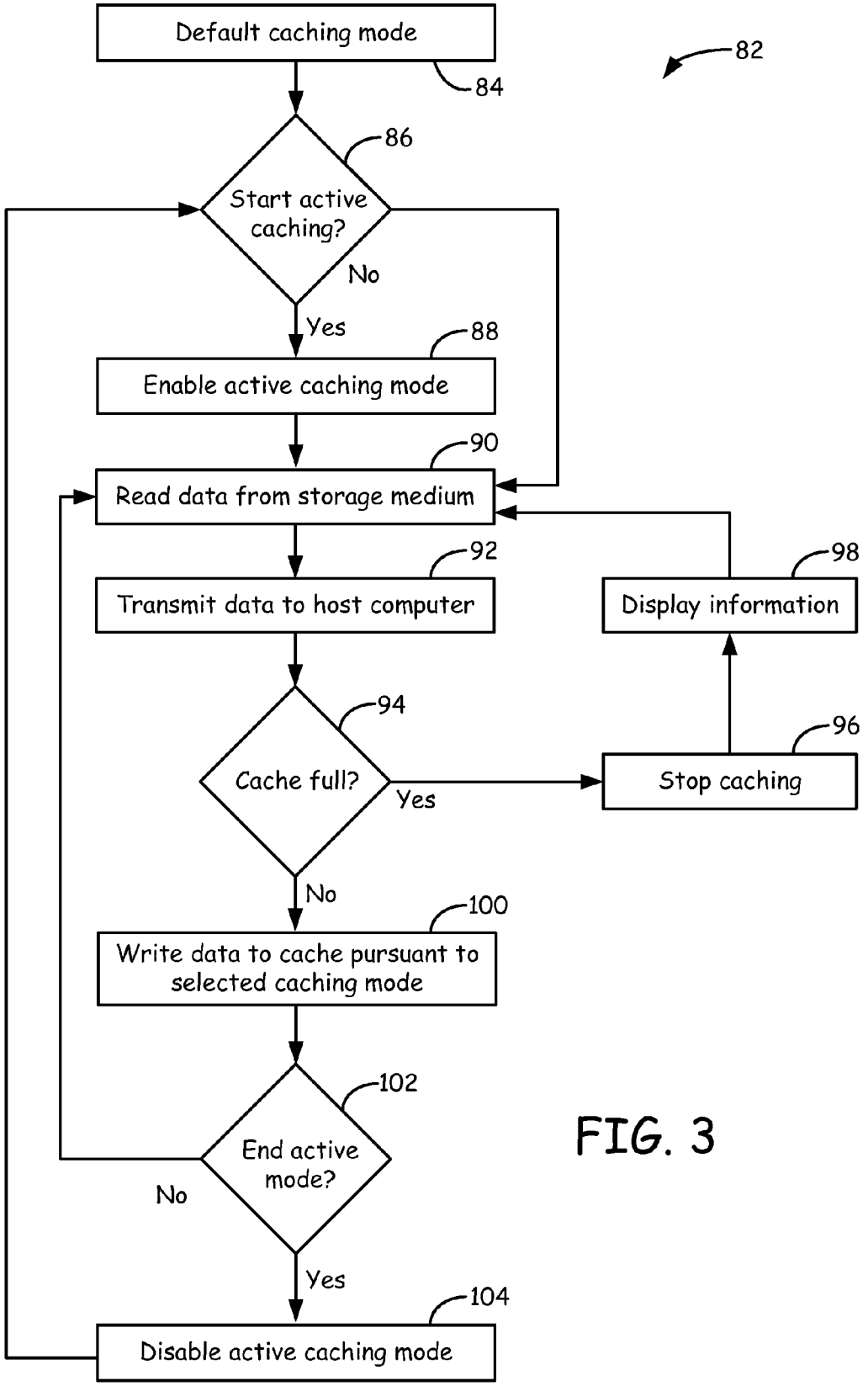

Default caching mode
84
82
86
Start active caching?
No
Yes
88
Enable active caching mode
90
Read data from storage medium
92
Transmit data to host computer
98
Display information
94
Cache full?
Yes
96
Stop caching
No
100
Write data to cache pursuant to selected caching mode
102
End active mode?
No
Yes
104
Disable active caching mode

FIG. 3

# SYSTEM AND METHOD FOR MANAGING STORAGE DEVICE CACHING

## CROSS-REFERENCE TO RELATED APPLICATIONS

**[0001]** The present application is a continuation of and claims priority to pending U.S. patent application Ser. No. 12/508,895, entitled SYSTEM AND METHOD FOR MANAGING STORAGE DEVICE CACHING, filed on Filed Jul. 24, 2009, the contents of which are hereby incorporated by reference in their entirety.

## BACKGROUND

**[0002]** The present disclosure is directed to data storage devices. In particular, the present disclosure is directed to cache management systems for data storage devices.

**[0003]** Data storage devices, such as magnetic disk drives, optical disk drives, hybrid drives, and solid state drives are used to store data for subsequent retrieval. Such devices employ data caches, which are typically high-speed semiconductor memory chips that enable the devices to rapidly manage the data and commands received from a host computer. Without caching, all read and write commands from the host computer would result in an access to the mass storage medium, such as a magnetic disk. This may result in significant time latencies due to mechanical positioning of the head relative to the magnetic disk. By caching, the storage device can buffer data that is likely to be accessed by the host computer so that when the data is actually accessed, the data is made available more quickly.

## SUMMARY

**[0004]** In an embodiment, an apparatus may comprise a controller configured to pin data from a non-volatile storage medium to a data cache based on at least one user-selectable caching mode.

**[0005]** In another embodiment, a device may comprise a processor configured to display a graphical user interface (GUI) at a display, and receive, from the GUI, an indication of a user selection of a data caching mode that determines which data to pin from a non-volatile memory of a data storage device to a data cache of the data storage device. The processor may be configured to then send an indication of the user selection of the data caching mode to the data storage device.

**[0006]** In yet another embodiment, a data storage memory device may store instructions that, when executed by a processor, perform a method comprising pinning data from a non-volatile memory to a data cache based on at least one user-selectable caching mode.

## BRIEF DESCRIPTION OF THE DRAWINGS

**[0007]** FIG. **1** is a schematic illustration of a data storage device in use with a host computer, where the data storage device includes a cache management system having user-selectable caching modes.

**[0008]** FIG. **2** is a flow diagram of a method of using a data storage device that includes user-selectable caching modes.

**[0009]** FIG. **3** is a flow diagram of a method of using a data storage device that includes an active caching mode.

## DETAILED DESCRIPTION

**[0010]** The present disclosure is directed to a data storage device having a caching management system (CMS) that allows a user to interactively control and monitor the data caching operations of the device. As discussed below, the CMS may include a plurality of user-selectable caching modes, where each caching mode includes an algorithm for caching data. This allows the user to modify performance of the data storage device and an associated host computer system based on the particular needs and desires of the user. This is in comparison to caching selections that rely solely on system-controlled caching algorithms. Such algorithms are typically based on statistical results (e.g., cache hit rates) rather than particular needs of individual users.

**[0011]** FIG. **1** shows storage device **10** in use with host computer **12**, where host computer **12** may be one or more computer-based systems configured to engage with storage device **10**. Storage device **10** is an example of a suitable data storage device of the present disclosure, and includes housing **14**, actuator **16**, display unit **18**, controller **20**, ROM module **22**, RAM cache **24**, non-volatile cache (NVC) **26**, and head disk assembly (HDA) **28**. Housing **14** is an exterior casing for storage device **10**, which desirably encases and retains the interior components, such as one or more of controller **20**, ROM module **22**, RAM cache **24**, NVC **26**, and HDA **28**. Accordingly, storage device **10** may be internal or external to a casing of host computer **12**.

**[0012]** Actuator **16** is a user interface (e.g., a push-button user interface) for storage device **10** that is retained by housing **14** and communicates with controller **20** over actuator interface **30**. As discussed below, actuator **16** may be activated (e.g., pressed) by a user to select different caching modes for operating storage device **10**. This allows the user to interact directly with storage device **10** to select which caching mode the user prefers storage device **10** to operate under. While actuator **16** is illustrated in FIG. **1** as a single press-button, storage device **10** may include a variety of different user interfaces for allowing a user to select different caching modes.

**[0013]** These user interfaces (e.g., actuator **16**) desirably operate independently of host computer **12**. This allows the user to select caching modes for storage device **10** while access to host computer **12** is unavailable, such as during early stages of a startup sequence. For example, as discussed below, during the initial boot sequence for starting host computer **12** (e.g., after the power-on self test), the user may select a caching mode with actuator **16** that directs controller **20** to cache all data read from HDA **28**. This allows data read from a startup sequence of HDA **28** (e.g., including a boot sector) to be cached for subsequent use. Pinning this cached data in a non-volatile cache (e.g., NVC **26**) may reduce subsequent startup times by allowing controller **20** to read the startup sequence data from NVC **26** rather than from the slower HDA **28**. Furthermore, this choice of whether to cache the startup sequence data may be determined entirely by the user.

**[0014]** Display unit **18** is a visual display unit retained by housing **14** in a manner that desirably allows the user to view information relating to the data caching in storage device **10**. As shown, display unit **16** communicates with controller **20** over display interface **32**. Display unit **18** may include a variety of different units for displaying information to the user, such as liquid-crystal display (LCD) units, light-emitting diode (LED) units, and the like. In one embodiment, display unit **18** may be separate from housing **14**, thereby

allowing display unit **18** be disposed at a remote location from storage device **10**. In this embodiment, display interface **32** may include a communication cable (not shown) that interconnects display unit **18** and controller **20**, and/or may include a wireless transmitter/receiver assembly (not shown).

**[0015]** Display unit **18** may provide a variety of different information relating to the data caching in storage device **10**, such as percentages of cache used, percentages of cache hits, cache hit rates, free and/or used cache volumes, available caching modes, selected caching modes, and the like. Because this information is viewable directly from storage device **10**, display unit **18** may display the information independently of host computer **12**. This allows the user to receive caching information even when such information may be unavailable through host computer **12**, such as during startup and shut down sequences.

**[0016]** Controller **20** is a circuit assembly that directs the reading and writing operations for storage device **10**, and is configured to respectively communicate with ROM module **22**, RAM cache **24**, NVC **26**, and HDA **28**. Controller **20** may also communicate with host computer **12** via host interface **34**, where host interface **34** may be any suitable interface, such as a universal serial bus (USB) interface, a Serial Advanced Technology Attachment (SATA) interface, an External SATA (eSATA) interface, a Parallel Advanced Technology Attachment (PATA) interface, an IEEE 1394 interface, a Small Computer System Interface (SCSI), a Serial Attached SCSI (SAS) interface, an Integrated Drive Electronics (IDE) interface, a Fiber Channel interface, and the like.

**[0017]** ROM module **22** is one or more non-volatile memory modules (e.g., read-only memory) for storing information such as firmware. RAM cache **24** is one or more volatile memory modules (e.g., dynamic random access memory) that may function as a volatile data cache during reading and/or writing operations with HDA **28** (i.e., a volatile disk buffer). NVC **26** is one or more solid state, non-volatile memory modules, such as flash memory, magnetic random access memory (MRAM), electrically erasable programmable read-only memory (EEPROM), ferroelectric random access memory (FeRAM), and combinations thereof. NVC **26** includes a plurality of erasable data blocks (referred to as data blocks **36**) for storing data, and desirably functions as a non-volatile data cache during reading and/or writing operations with HDA **28** (i.e., a non-volatile disk buffer).

**[0018]** HDA **28** is one or more head disk assemblies, and is the primary storage medium/media for storage device **10**. HDA **28** includes storage disk **38** and spindle motor **40**, where spindle motor **40** rotates storage disk **38** in a rotational direction during operation. Storage disk **38** includes recordable surface **42**, which is a surface of storage disk **38** having multiple data tracks (not shown) for storing data (e.g., startup sequence data). HDA **28** further includes actuation motor **44** (e.g., a voice coil motor) and actuator arm assembly **46**, where actuator arm assembly **46** may include an actuator arm, suspension assembly, and slider to carry a transducing head for writing data to, and reading data from, the data tracks in recordable surface **42**. Actuation motor **44** is configured to pivot actuator arm assembly **46** about an axis to sweep the transducing head in an arc over recordable surface **42**. The operation of HDA **28** is directed by controller **20**, where controller **20** may communicate with servo and spindle controls to read from, and write data to, recordable surface **42**.

**[0019]** Reading data from, and writing data to, HDA **28** is generally slower than the respective operations with RAM cache **24** and/or NVC **26**. This is due in part to the mechanical arrangement of HDA **28**, which requires storage disk **38** to spin up and actuator arm assembly **46** to seek the desired data tracks in recordable surface **42**. As such, pinning data to RAM cache **24** and/or NVC **26** may increase data retrieval efficiencies by allowing controller **20** to read the cached data directly from RAM cache **24** and/or NVC **26** rather than from the slower HDA **28**. Allowing the user to determine which data is to be cached correspondingly provides greater user control over storage device **10** and host computer **12**, and allows the performance of storage device **10** to be modified to suit the particular needs and desires of the user.

**[0020]** Storage device **10** also includes CMS **48**, which desirably includes a plurality of user-selectable caching modes (e.g., caching mode **1**, caching mode **2**, . . . , caching mode n) for directing how controller **20** performs cache writing operations from HDA **28** to RAM cache **24** and/or NVC **26**. Instructions for instantiating CMS **48** may be stored on any suitable storage medium/media in storage device **10**, such as in ROM module **22**, NVC **26**, and/or HDA **28**. When storage device **10** is powered up, CMS **48** is desirably copied from the given storage medium to RAM cache **24**. CMS **48** may then direct the cache writing operations that controller **20** performs based on a selected caching mode.

**[0021]** The caching modes of CMS **48** may include a variety of different algorithms for directing controller **20** to perform cache writing operations. Furthermore, the caching modes may be preset caching modes, user-defined caching modes, system-modified caching modes, and combinations thereof. The preset caching modes are predefined caching modes that provide suitable default algorithms for performing cache writing operations, and may be stored in any suitable storage medium of storage device **10**, such as ROM module **22**, NVC **26**, and/or HDA **28**. User-defined caching modes are caching modes that users may generate with host computer **12**, as discussed below. System-modified caching modes are caching modes that may be modified by storage device **10** and/or host computer **12** based on cache statistical results. Examples of suitable techniques for generating system-modified caching modes based on cache statistical results include those disclosed in Herbst et al., U.S. Patent Application Publication No. 2003/0061444. When generated, the user-defined caching modes and the system-modified caching modes may each be stored in any suitable storage medium of storage device **10**, such as NVC **26** and/or HDA **28**. The caching modes may then be copied to RAM cache **24** with CMS **48** when storage device **10** is powered up.

**[0022]** As discussed above, the caching modes may define a variety of different algorithms for directing controller **20** to perform cache writing operations from HDA **28** to RAM cache **24** and/or NVC **26**. For example, the caching modes may define usable cache volumes, may define optimal operating characteristics for one or more benchmarks, may apply different algorithms for writing data to the data caches (e.g., first-in-last-out algorithms), may allocate portions of the data caches to specific algorithms (e.g., volatile vs. non-volatile storage), and the like.

**[0023]** In one embodiment, the caching modes of CMS **48** may include an active caching mode. As used herein, the term "active caching mode" refers to a caching mode in which a controller (e.g., controller **20**) copies and pins all data read from a storage medium (e.g., HDA **28**) to one or more data caches of the storage device (e.g., RAM cache **24** and/or NVC **26**). As such, in the embodiment shown in FIG. **1**, when the

user selects the active caching mode, all data that controller **20** reads from recordable surface **42** of HDA **28** for transmission to host computer **12**, is also copied and pinned to RAM cache **24** and/or NVC **26**. In this embodiment, while the active caching mode in CMS **48** remains selected, controller **20** continues to copy and pin all data read from HDA **28** to RAM cache **24** and/or NVC **26**. This may continue until either RAM cache **24** and/or NVC **26** are full, or until the user disables the active caching mode (e.g., the user selects a different caching mode).

**[0024]** In this embodiment, actuator **16** may function as an on/off switch for enabling and disabling the active caching mode. Accordingly, when the user presses actuator **16**, controller **20** may be directed to operate under the active caching mode, as discussed above. At a subsequent point in time, when the user desires to disable the active caching mode, the user may press actuator **16** again to select another non-active caching mode, thereby disabling the active caching mode. The non-active caching mode may be one of a variety of caching modes, such as a default preset caching mode. Controller **20** may then perform cache writing operations pursuant to the non-active caching mode.

**[0025]** In addition to functioning as an on/off switch, actuator **16** may also provide additional data caching functions, such as data purging. In this embodiment, after the active caching mode has been enabled and disabled, when the user presses actuator **16** again to re-enable the active caching mode, one or more portions of the previously pinned data may be purged from RAM cache **24** and/or NVC **26**. The data purging may empty entire physical regions of RAM cache **24** and/or NVC **26**, or may selectively purge data based on criteria. Suitable criteria for data purging include the number of hits, the residence time in RAM cache **24** and/or NVC **26**, and the like. The purging of RAM cache **24** and/or NVC **26** in this manner may accordingly free up storage space for subsequent cache writing operations. In one embodiment, the purging operation may be performed by pressing and holding actuator **16** for a predetermined time period (e.g., a few seconds).

**[0026]** Display unit **18** may also indicate whether controller **20** is operating under the active caching mode. For example, when controller **20** is operating under the active caching mode, display unit **18** may provide one or more visual and/or audible indicators, such as a textual indicator stating "ACTIVE CACHING ON", an illuminated LED indicator, an audible sound, and the like. When RAM cache **24** and/or NVC **26** become full, the active caching mode may be disabled, and display unit **18** may provide a corresponding indication, such as "CACHE FULL". Likewise, when the user disables the active caching mode, the display unit **18** may provide a corresponding indication, such as "ACTIVE CACHING OFF".

**[0027]** The active caching mode is particularly suitable for use with hybrid drives (e.g., storage device **10**) and solid state drives, each of which desirably includes one or more large-volume, non-volatile caches (e.g., NVC **26**). A substantial amount of data may be cached and retained using non-volatile caches having large storage volumes. For example, in embodiments in which NVC **26** has a storage volume of about 32 gigabytes (GB), and controller **20** reads data from HDA **28** at a rate of about 100 megabytes/second, the active caching mode may run for about 5 minutes before data blocks **36** of NVC **26** become full. Similarly, a 250 GB volume for NVC **26** would allow the caching operation under the active caching mode to continue for about 40 minutes before NVC **26** becomes full. Because data is typically re-read from previously cached sectors, the active caching mode may actually continue for greater durations than these examples listed above. As such, providing the user the means to select the active caching mode in combination with large caching volumes allows substantial amounts of data to be pinned to data caches based on the particular needs and desires of the user.

**[0028]** This combination is also suitable for caching startup sequence data, as discussed above. For example, during the initial startup sequence for starting host computer **12** (e.g., after the power-on self test), the user may select the active caching mode with actuator **16**, thereby directing controller **20** to cache all data read from HDA **28**. This desirably directs controller **20** to copy and pin all startup sequence data to data blocks **36** of NVC **26**. When the startup sequence is complete, the user may then press actuator **16** again to select a non-active caching mode (e.g., a default caching mode), thereby disabling the active caching mode. Due to the non-volatile nature of NVC **26**, the retained startup sequence data may then be read from NVC **26** during subsequent startup sequences of host computer **12**. As discussed above, this may reduce subsequent startup times by allowing controller **20** to read the startup sequence data from NVC **26** rather than from the slower HDA **28**.

**[0029]** In comparison, data cache managers that are based on software are generally unsuitable for selectively caching startup sequence data from data storage devices. Such data cache managers typically require the startup sequence of the host computer to be completed before they may be used. This effectively prevents these types of data cache managers from being selectively operated by a user during the startup sequences. Storage device **10** with CMS **48**, however, allows the user to selectively cache data during a variety of different power cycles of host computer **12**, including startup and shut down sequences.

**[0030]** In addition to actuator **16** and display unit **18**, in some embodiments, the user may also select caching modes for controller **20** through host computer **12**. In these embodiment, host computer **12** may provide software-based tools for allowing the user to select the caching modes for storage device **10**, as well as for reviewing caching statistics and generating user-defined caching modes for CMS **48**.

**[0031]** As discussed above, host computer **12** may be one or more computer-based systems configured to engage with storage device **10**. Examples of suitable systems for host computer **12** include desktop computers, laptops, server-based systems, personal digital assistants (PDA), telecommunication devices, multimedia players (e.g., portable multimedia players), and combinations thereof. In the embodiment shown in FIG. **1**, host computer **12** includes graphical user interface (GUI) **50**, memory controller **52**, processor **54**, RAM **56**, input/output (I/O) controller **58**, and data storage device (DSD) interface **60**. Host computer **12** may also include a variety of additional components that are typically contained in computer-based systems.

**[0032]** GUI **50** is a user interface, such as an operating system interface, that may be instantiated and stored in RAM **56** during operation. Memory controller **52** is a circuit assembly that interfaces the components of host computer **12** with RAM **56**, and may include logic for mapping addresses of the components of host computer **12** to particular areas of RAM **56**. Processor **54** is one or more processing units, and RAM **56** is one or more volatile random access memory modules. I/O controller **58** is a circuit assembly that interfaces memory

controller **52**, processor **54**, and RAM **56** with various input and output components of host computer **12**, including DSD interface **60**. DSD interface **60** is the reciprocating interface of host interface **34**, which allows storage device **10** to engage and communicate with host computer **12**. Suitable interfaces for DSD interface **60** include the reciprocating interfaces for those discussed above for host interface **34**. Furthermore, I/O controller **58** and/or the DSD interface **60** may include logic to support RAID technology, thereby allowing storage device **10** to operate using RAID technology.

**[0033]** The user may interact with CMS **48** via host computer **12** through a software application that may be instantiated and stored in RAM **56**, and displayed in GUI **50**. The software application may be instantiated in host computer **12** by loading program instructions from NVC **26** and/or HDA **28** with controller **20**, and relaying the instructions to host computer **12**. The instructions may then be stored in RAM **64** during operation. The software application may then allow the user to interact with CMS **48** in a variety of manners.

**[0034]** In one embodiment, through the software application, the user may select one or more of the caching modes for directing the cache writing operations of storage device **10**. For example, the software application may provide a selectable menu in GUI **50**, providing a list of the caching modes that are available in CMS **48**. In this embodiment, when the software application is instantiated and stored in RAM **64**, the software application may query storage device **10** for the caching modes available in CMS **48**. Controller **20** may then transmit the list of the available caching modes to host computer **12** for display and selection by the user.

**[0035]** In an additional embodiment, the software application may allow the user to generate user-defined caching modes, as discussed above. Furthermore, the software application may allow the user to remove specific data from RAM cache **24** and/or NVC **26**, such as by file name, block numbers, or by specifying that blocks with zero hits (or some other number) be purged from the given data cache(s). The display on GUI **50** may also provide information relating to statistical information of RAM cache **24** and/or NVC **26**. For example, the program may display information such as the percentages of cache used, percentages of cache hits, cache hit rates, free and/or used cache volumes, available caching modes, selected caching modes, and the like. Moreover, in some embodiments, the program may allow the user to select applications and/or data to be cached. For example, GUI **50** may include menus for an application or data file that allow the user to selectively cache this application and/or data file. Additionally, the program may also identify applications and/or data files that are related to the application or data file already selected. The program may then prompt the user to cache the related data (e.g., dynamic link libraries).

**[0036]** As discussed above, storage device **10** allows a user to interactively control and monitor data caching operations. The inclusion of user-selectable caching modes accordingly allows the user to modify performance of storage device **10** and host computer **12** based on the particular needs and desires of the user. Furthermore, the use of a user interface (e.g., actuator **16**) that is independent of host computer **12** is beneficial for situations in which selective data caching through software programs is generally unavailable (e.g., startup and shut down sequences). This increases the versatility of storage device **10** in accommodating the particular needs of individual users.

**[0037]** FIG. **2** is a flow diagram of method **62** for using a data storage device. The following discussion of method **62** is made with reference to storage device **10** (shown in FIG. **1**) with the understanding that method **62** may be used with a variety of data storage devices. As shown, method **62** includes steps **64-80**, and initially involves providing a plurality of caching modes for selectively operating controller **20** (step **64**). For example, display unit **18** of storage device **10** and/or GUI **50** of host computer **12** may provide lists of available caching modes for the user to select from. When the user identifies a desired caching mode, the user then may select the given caching mode, such as with actuator **16** and/or through the software application in GUI **50** (step **66**).

**[0038]** After the desired caching mode has been selected, subsequent read operations may be performed, where controller **20** reads data from one or more storage media. For example, controller **20** may initially determine whether the intended data is already cached in RAM cache **24** and/or NVC **26**. If so, controller **20** may read the data from RAM cache **24** and/or NVC **26** and transmit one or more portions of the read data to host computer **12**. Alternatively, if the intended data is not currently cached, controller **20** may read the data from HDA **28** (step **68**) and transmit one or more portions of the read data to host computer **12** (step **70**). At this point, controller **20** may also verify whether the intended data cache, such as NVC **26**, is full (step **72**). In alternative embodiments, this verification under step **72** may be performed prior to, or concurrent with, one or more of steps **64-70**. In the event that NVC **26** is full, controller **20** may stop caching (step **74**) and display information to the user that NVC **26** is full via display unit **18** and/or GUI **50** (step **76**). The user may then either allow controller **20** to continue to read and transmit the data pursuant to steps **68-76** without caching, or have controller **20** purge one or more portions of the cached data in NVC **26**. After purging, controller **20** may then continue to read and transmit the data pursuant to steps **68** and **70**, and perform a cache writing operation, as discussed below under step **78**.

**[0039]** If, pursuant to step **72**, NVC **26** contains free data blocks **36**, controller **20** may then copy and pin one or more portions of the data read from HDA **28** to NVC **26** in conformance with the algorithm of the selected caching mode (step **78**). This data pinning allows the data to be retained in NVC **26** for subsequent reading operations. Steps **68-78** may then be repeated pursuant to the selected caching mode until the user decides to select a different caching mode (step **80**). At this point, the user may review the list of available caching modes (step **64**) and select a new caching mode for controller **20** to operate under (step **66**). These steps of method **62** may then be repeated as desired by the user, thereby allowing the user to cache data in NVC **26** based on the user's particular needs.

**[0040]** FIG. **3** is a flow diagram of method **82** for using a storage device, and is an example of a suitable aspect of method **62** (shown in FIG. **2**) that involves an active caching mode. As discussed above, the term "active caching mode" refers to a caching mode in which a controller copies and pins all data read from a storage medium to one or more data caches of the storage device.

**[0041]** The following discussion of method **82** is also made with reference to storage device **10** (shown in FIG. **1**) with the understanding that method **82** may be used with a variety of data storage devices. As shown in FIG. **3**, method **82** includes steps **84-104**, and initially involves operating controller **20** under a default caching mode (i.e., a non-active caching

mode) (step **84**). If the user desires to copy and pin all data read from HDA **28** to one or more data caches (e.g., NVC **26**) (step **86**), the user may enable the active caching mode (step **88**). As discussed above, the user may enable the active caching mode in a variety of manners, such as by pressing actuator **16**.

**[0042]** Steps **90-100** of method **82** generally correspond to steps **68-78** of method **62**. Accordingly, controller **20** may read data from one or more storage media. For example, controller **20** may initially determine whether the intended data is already cached in RAM cache **24** and/or NVC **26**. If so, controller **20** may read the data from RAM cache **24** and/or NVC **26** and transmit one or more portions of the read data to host computer **12**. Alternatively, if the intended data is not currently cached, controller **20** may read the data from HDA **28** (step **90**) and transmits one or more portions of the read data to host computer **12** (step **92**). At this point, controller **20** may also verify whether the intended data cache, such as NVC **26**, is full (step **94**). In alternative embodiments, this verification under step **94** may be performed prior to, or concurrent with, one or more of steps **84-92**. In the event that NVC **26** is full, controller **20** may stop caching (step **96**) and display information to the user via display unit **18** and/or GUI **50**, as discussed above (step **98**). The user may then either allow controller **20** to continue to read and transmit the data pursuant to steps **90-98** without caching, or have controller **20** purge one or more portions of the cached data in NVC **26**. After purging, controller **20** may then continue to read and transmit the data pursuant to steps **90** and **92**, and perform a cache writing operation pursuant to the active caching mode, as discussed below under step **100**.

**[0043]** If, pursuant to step **94**, NVC **26** contains free data blocks **36**, controller **20** may then copy and pin all data read from HDA **28** to NVC **26**, in conformance with the active caching mode (step **100**). This data pinning allows the data to be retained in NVC **26** for subsequent reading operations. Steps **90-100** may then be repeated pursuant to the active caching mode until the user desires to stop the active caching (step **102**). At this point, the user may disable the active caching mode, such as by pressing actuator **16** again (step **104**). This may switch operation of controller **20** back to the default caching mode, or another non-active caching mode as desired by the user.

**[0044]** While under the non-active caching mode (and if the user does not desire to re-enable the active caching mode, pursuant to step **86**), steps **90-104** may then be repeated under the selected non-active caching mode (e.g., a default caching mode). Method **82** accordingly allows cached data to be retained in NVC **26** for subsequent reading operations based on the particular needs of the user. As discussed above, this method is particularly suitable for caching startup sequence data when applied in combination with a user interface (e.g., actuator **16**) that is independent of host computer **12**.

**[0045]** Although the present invention has been described with reference to preferred embodiments, workers skilled in the art will recognize that changes may be made in form and detail without departing from the spirit and scope of the invention.

**1**. An apparatus comprising:
a controller configured to pin data from a non-volatile storage medium to a data cache based on at least one user-selectable caching mode.

**2**. The apparatus of claim **1**, further comprising the at least one user-selectable caching mode includes:
one or more caching modes that are modified based on cache statistical results.

**3**. The apparatus of claim **1**, further comprising:
the data cache includes a non-volatile cache; and
the at least one user-selectable caching mode includes an active caching mode configured to pin all data requested by a host from the non-volatile memory to the data cache.

**4**. The apparatus of claim **1**, further comprising the controller configured to:
implement the at least one user-selectable caching mode based on input received at a user interface; and
purge data from the data cache based on input received at the user interface.

**5**. The apparatus of claim **4**, further comprising the user interface includes a display unit configured to display information relating to cache pinning operations performed by the controller.

**6**. The apparatus of claim **1**, further comprising:
the data cache;
a user interface configured to allow a user to select the at least one user-selectable caching mode; and
a housing for retaining the data cache, the controller, and the user interface.

**7**. The apparatus of claim **1**, further comprising:
the controller further configured to pin data based on read commands received from a host device; and
the at least one user-selectable caching mode is selectable in a manner that is independent of the host device.

**8**. The apparatus of claim **7**, further comprising:
the controller configured to receive data from the host including an indication of a selection of a user-selectable caching mode.

**9**. The apparatus of claim **7**, further comprising:
the controller configured to transmit statistics relating to cache pinning operations to the host for display on a graphical user interface.

**10**. A device comprising:
a processor configured to:
initiate a display of a graphical user interface (GUI);
receive, from the GUI, an indication of a user selection of a data caching mode that determines which data to pin from a non-volatile memory of a data storage device to a data cache of the data storage device; and
send an indication of the user selection of the data caching mode to the data storage device.

**11**. The device of claim **10**, further comprising the processor configured to:
display a list of available data caching modes via the GUI.

**12**. The device of claim **10**, further comprising the processor configured to:
provide tools allowing a user to generate user-defined caching modes via the GUI; and
send the user-defined caching modes to the data storage device for storage to the non-volatile memory.

**13**. The device of claim **10**, further comprising the processor configured to:
display caching statistics received from the data storage device via the GUI.

**14**. The device of claim **10**, further comprising the processor configured to:
receive instructions from a user via the GUI to remove specific data from the data cache; and

send the instructions to remove specific data to the data storage device.

**15**. The device of claim **14**, further comprising:

the specific data is specified based on file name, block numbers, or based on a threshold number of cache hits.

**16**. The device of claim **10**, the processor further configured to:

receive instructions from a user via the GUI to pin selected data to the data cache; and

send the instructions to pin the selected data to the data storage device.

**17**. The device of claim **16**, the processor further configured to:

display a list of data related to the selected data to the user via the GUI;

receive a selection to pin the related data from a user via the GUI; and

send the selected related data to the data storage device.

**18**. The device of claim **17**, further comprising:

the selected data includes an application, and the related data includes data files associated with the application.

**19**. The device of claim **17**, further comprising:

the selected data includes a data file, and the related data includes an application associated with the data file.

**20**. The device of claim **17**, further comprising:

the selected data includes a first data file, and the related data includes other data files related to the first data file.

**21**. A data storage memory device storing instructions that, when executed by a processor, perform a method comprising:

pinning data from a non-volatile memory to a data cache based on at least one user-selectable caching mode.

**22**. The data storage memory device of claim **21**, the method further comprising:

purging data selected by a user from the data cache.

**23**. The data storage memory device of claim **21**, the method further comprising:

pinning specific data identified by a user to the data cache.

* * * * *