



(51) International Patent Classification:
G06F 16/00 (2019.01)

(21) International Application Number:
PCT/CN2019/096857

(22) International Filing Date:
19 July 2019 (19.07.2019)

(25) Filing Language: English

(26) Publication Language: English

(71) Applicant: **ALIBABA GROUP HOLDING LIMITED**
[—/CN]; Fourth Floor, One Capital Place, P.O. Box 847,
George Town, Grand Cayman (KY).

(72) Inventors: **FU, Chen**; 10322 Alpine Dr. Apt B, Cupertino,
CA 95014 (US). **LUAN, Xiaofan**; Building 8, No.969 West
Wen Yi Road, Yu Hang District, Hangzhou (CN).

(74) Agent: **BEIJING TSINGYUANHUI INTELLECTUAL
PROPERTY LAW FIRM**; Room 101, 1st Floor, Build-
ing 1, No. C-18, Zhichun Road, Haidian District, Beijing
100190 (CN).

(81) Designated States (*unless otherwise indicated, for every
kind of national protection available*): AE, AG, AL, AM,
AO, AT, AU, AZ, BA, BB, BG, BH, BN, BR, BW, BY, BZ,
CA, CH, CL, CN, CO, CR, CU, CZ, DE, DJ, DK, DM, DO,
DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, GT, HN,

HR, HU, ID, IL, IN, IR, IS, JO, JP, KE, KG, KH, KN, KP,
KR, KW, KZ, LA, LC, LK, LR, LS, LU, LY, MA, MD, ME,
MG, MK, MN, MW, MX, MY, MZ, NA, NG, NI, NO, NZ,
OM, PA, PE, PG, PH, PL, PT, QA, RO, RS, RU, RW, SA,
SC, SD, SE, SG, SK, SL, SM, ST, SV, SY, TH, TJ, TM, TN,
TR, TT, TZ, UA, UG, US, UZ, VC, VN, ZA, ZM, ZW.

(84) Designated States (*unless otherwise indicated, for every
kind of regional protection available*): ARIPO (BW, GH,
GM, KE, LR, LS, MW, MZ, NA, RW, SD, SL, ST, SZ, TZ,
UG, ZM, ZW), Eurasian (AM, AZ, BY, KG, KZ, RU, TJ,
TM), European (AL, AT, BE, BG, CH, CY, CZ, DE, DK,
EE, ES, FI, FR, GB, GR, HR, HU, IE, IS, IT, LT, LU, LV,
MC, MK, MT, NL, NO, PL, PT, RO, RS, SE, SI, SK, SM,
TR), OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW,
KM, ML, MR, NE, SN, TD, TG).

Published:
— with international search report (Art. 21(3))

(54) Title: HOT-COLD DATA SEPARATION METHOD FOR REDUCING WRITE AMPLIFICATION IN KEY-VALUE STORES

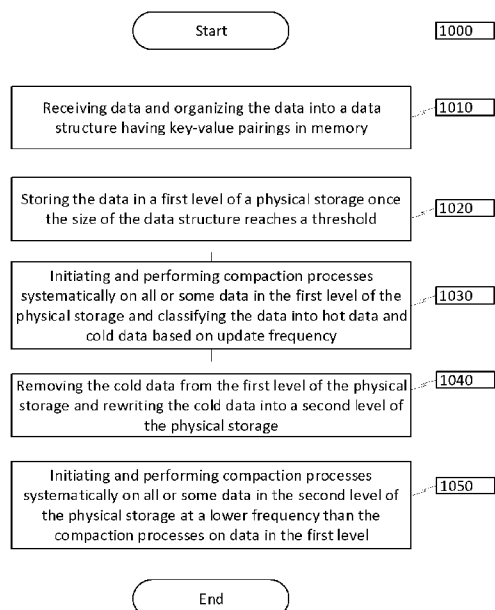


FIG. 6

(57) Abstract: A method incorporating a hot-cold data separation during compaction processes in data storage systems is disclosed. The method comprises storing data in into a first level of a physical storage; initiating compaction processes on data in the first level of the physical storage; classifying, during the compaction processes, the data into a first group of data and a second group of data based on update frequency; removing the second group of data from the first level of the physical storage; and rewriting the second group of data into a second level of the physical storage, wherein a frequency of initiating compaction processes on data in the first level of the physical storage is higher than a frequency of initiating compaction processes on data in the second level of the physical storage. A data storage system incorporating hot-cold data separation during compaction processes, and a non-transitory computer readable medium that stores a set of instructions that are executable by one or more processors of an apparatus to perform a method incorporating hot-cold data separation during compaction processes are also disclosed.

HOT-COLD DATA SEPARATION METHOD FOR REDUCING WRITE AMPLIFICATION IN KEY-VALUE STORES

BACKGROUND

[001] In data storage systems, key-value stores are widely used to power applications
5 from web searches to e-commerce. Key-value store implementation often organizes data using
rooted tree structures (e.g., log structured merge trees (LSM trees)). Rooted tree structures
require periodic compaction processes, which rewrite all or part of the data files in the key-value
store to improve read performance and remove obsolete records of data. The compaction
processes, however, can occupy a large amount of bandwidth resources available to clients of the
10 data storage systems, and hence raising the operational cost of the data storage system.
Accordingly, conventional data storage systems need improvements.

SUMMARY

[002] Embodiments of the present disclosure provide a method in data storage systems
incorporating a hot-cold data separation during compaction processes. The method comprises
15 storing data in into a first level of a physical storage; initiating and performing compaction
processes on data in the first level of the physical storage; classifying, during the compaction
processes, the data into a first group of data and a second group of data based on update
frequency; removing the second group of data from the first level of the physical storage; and
rewriting the second group of data into a second level of the physical storage, wherein a
20 frequency of initiating and performing compaction processes on data in the first level of the
physical storage is higher than a frequency of initiating and performing compaction processes on
data in the second level of the physical storage.

[003] Embodiments of the present disclosure provides a data storage system incorporating hot-cold data separation during compaction processes. The data storage system comprises a physical storage comprising a first level and a second level, a memory, and a processor configured to store data in key-value stores into the first level of the physical storage, perform compaction processes on data in the first level of the physical storage, classify, during the compaction processes, the data into a first type of data and a second type of data based on update frequency, remove the second type of data from the first level of the physical storage, and rewrite the second type of data into the second level of the physical storage, wherein a frequency of initiating performing compaction processes on data in the first level of the physical storage is higher than a frequency of initiating and performing compaction processes on data in the second level of the physical storage.

[004] Embodiments of the present disclosure also provide a non-transitory computer readable medium that stores a set of instructions that are executable by one or more processors of an apparatus to perform a method incorporating hot-cold data separation during compaction processes. The method comprises storing data in key-value stores into a first level of a physical storage; initiating and performing compaction processes on data in the first level of the physical storage; classifying, during the compaction processes, the data into a first type of data and a second type of data based on update frequency; removing the second type of data from the first level of the physical storage; and rewriting the second type of data into a second level of the physical storage, wherein a frequency of initiating and performing compaction processes on data in the first level of the physical storage is higher than a frequency of initiating and performing compaction processes on data in the second level of the physical storage..

BRIEF DESCRIPTION OF THE DRAWINGS

[005] Embodiments and various aspects of the present disclosure are illustrated in the following detailed description and the accompanying figures. Various features shown in the figures are not drawn to scale.

5 [006] **FIG. 1** is a schematic diagram illustrating an exemplary server of a data storage system, according to some embodiments of the present disclosure.

[007] **FIG. 2** is a schematic diagram illustrating an exemplary data storage system performing a saving operation into a physical storage, according to some embodiments of the present disclosure.

10 [008] **FIG. 3** is a schematic diagram illustrating an exemplary data storage system incorporating hot-cold data separation, according to some embodiments of the present disclosure.

[009] **FIG. 4** is a schematic diagram illustrating an exemplary data storage system incorporating hot-cold data separation using eviction spaces, according to some embodiments of the present disclosure.

15 [010] **FIG. 5** is a schematic diagram illustrating an exemplary data storage system incorporating hot-cold data separation using multiple levels, according to some embodiments of the present disclosure.

[011] **FIG. 6** illustrates a flow diagram of an exemplary method incorporating hot-cold data separation, according to some embodiments of the present disclosure.

20 [012] **FIG. 7** illustrates a flow diagram of an exemplary method for hot-cold data separation in more than two levels of physical storage, according to some embodiments of the present disclosure.

[013] **FIG. 8** illustrates a flow diagram of an exemplary method for hot-moderate-cold data separation in more than two levels of physical storage, according to some embodiments of the present disclosure.

DETAILED DESCRIPTION

5 [014] The following description refers to the accompanying drawings in which the same numbers in different drawings represent the same or similar elements unless otherwise represented. The implementations set forth in the following description of exemplary embodiments do not represent all implementations consistent with the disclosure. Instead, they are merely examples of apparatuses and methods consistent with aspects related to the disclosure
10 as recited in the appended claims.

[015] In data storage systems, key-value stores are a popular form of data storage engines. Key-value stores is a data structure designed for storing, retrieving, and managing data in a form of associative arrays, and is more commonly known as a dictionary or a hash table. Key-value stores contain a collection of objects or records, which in turn have many different
15 fields within them, each containing data. These records are stored and retrieved using a key that uniquely identifies the record. The key is used to quickly find a requested data within the data storage systems.

[016] Rooted tree structures, such as the LSM trees, is standard for key-value stores. The rooted tree structures do not perform update operations on data records directly in place.
20 Instead, the rooted tree structures insert updates into the key-value stores as a new version of the same key. For example, when a delete operation is performed, the rooted tree structures can insert delete operations as updates with keys and a delete marker. New updates can render old versions of the same key obsolete. Due to the nature of the rooted tree structures, the updates of

the same key naturally fall into locations that are close with each other. When a read operation is performed, the rooted tree structures can trace from the youngest version to the oldest version of the key and return version(s) that are still valid.

[017] Over time, data volume of the data storage systems grow indefinitely. To prevent
5 the data storage systems from running out of space, a process called compaction is performed periodically. The compaction process is a background process that reads some or all data stores, and then combines them into one or more new data stores using a sorting process (e.g., merge sort). The compaction process brings different versions of the same key together during the sorting process and discards obsolete versions. The compaction process then writes valid
10 versions of each key into a new data store.

[018] The compaction process is performed periodically on the data storage system to remove obsolete records and keep the data storage system from running out of space. In addition, the sorting process within the compaction process can realign data to improve read performance. Therefore, the compaction process repeatedly reads and rewrites data that has already been
15 written to a physical storage, causing write amplification. Write amplification is a phenomenon where a volume of writes is many times a volume of updates requested by a client of the data storage system (e.g., an application or a user). For example, each time a compaction process is performed, a record is read and rewritten at least once. Therefore, if the compaction process is performed 100 times per hour, the record would be read and rewritten at least 100 times, even if
20 the client may have never updated the record in the same time period. As a result, the constant reads and rewrites performed by the compaction process can consume a vast majority of an input/output (I/O) bandwidth provided by the physical storage, which competes with the client's operations and greatly reduces the throughput of the entire system.

[019] One source of data amplification is an uneven distribution in data updates. For example, a few records may have a much higher probability of being read or updated than other records. Records that experience more frequent reads and updates can be categorized as “hot keys,” whereas records that experience less frequent reads and updates can be categorized as “cold keys.” The compaction process, however, makes no distinction between both types of records. The compaction process mixes the hot keys and the cold keys and rewrites all of them, even though the cold keys may have never been updated.

[020] To mitigate this problem, some systems separate hot keys and cold keys to reduce the amount of rewrites on data that is designated by cold keys. These systems, however, require a separate memory data structure to keep track of records’ access history. In large scale systems, this data structure often takes up a large amount of memory. Some systems may even require a deployment of a dedicated data storage for this purpose specifically.

[021] Another disadvantage with conventional systems is that these systems implement the separate memory data structure to keep track of both read operations and update operations of a record in order to determine whether the record is “hot.” In many of these systems, the number of read operations can be an order of magnitude larger than the number of update operations. For example, for every 10 read operations on a record, there may be only 1 update operation on the same record. If a solution can rely on only the update operations, the systems would no longer have to keep an access history of read operations, hence greatly reduce the total number of operations to keep track of.

[022] Embodiments of the present disclosure resolve these issues using a new hot-cold data separation method for reducing write amplifications. **FIG. 1** is a schematic diagram illustrating an exemplary server 110 of a data storage system 100, according to some

embodiments of the present disclosure. According to **FIG. 1**, server 110 comprises a bus 112 or other communication mechanism for communicating information, and one or more processors 116 communicatively coupled with bus 112 for processing information. Processors 116 can be, for example, one or more microprocessors.

5 [023] Server 110 can transmit data to or communicate with another server 130 through a network 122. Network 122 can be a local network, an internet service provider, internet, or any combination thereof. Communication interface 118 of server 110 is connected to network 122. In addition, server 110 can be coupled via bus 112 to peripheral devices 140, which comprises displays (e.g., cathode ray tube (CRT), liquid crystal display (LCD), touch screen, etc.) and input
10 devices (e.g., keyboard, mouse, soft keypad, etc.).

 [024] Server 110 can be implemented using customized hard-wired logic, one or more ASICs or FPGAs, firmware, or program logic that in combination with the server causes server 110 to be a special-purpose machine.

 [025] Server 110 further comprises storage devices 114, which may include memory
15 161 and physical storage 164 (e.g., hard drive, solid-state drive, etc.). Memory 161 may include random access memory (RAM) 162 and read only memory (ROM) 163. Storage devices 114 can be communicatively coupled with processors 116 via bus 112. Storage devices 114 may include a main memory, which can be used for storing temporary variables or other intermediate information during execution of instructions to be executed by processors 116. Such instructions,
20 after being stored in non-transitory storage media accessible to processors 116, render server 110 into a special-purpose machine that is customized to perform operations specified in the instructions. The term “non-transitory media” as used herein refers to any non-transitory media storing data or instructions that cause a machine to operate in a specific fashion. Such non-

transitory media can comprise non-volatile media and/or volatile media. Non-transitory media include, for example, optical or magnetic disks, dynamic memory, a floppy disk, a flexible disk, hard disk, solid state drive, magnetic tape, or any other magnetic data storage medium, a CD-ROM, any other optical data storage medium, any physical medium with patterns of holes, a RAM, a PROM, and EPROM, a FLASH-EPROM, NVRAM, flash memory, register, cache, any
5 other memory chip or cartridge, and networked versions of the same.

[026] Various forms of media can be involved in carrying one or more sequences of one or more instructions to processors 116 for execution. For example, the instructions can initially be carried out on a magnetic disk or solid state drive of a remote computer. The remote computer
10 can load the instructions into its dynamic memory and send the instructions over a telephone line using a modem. A modem local to server 110 can receive the data on the telephone line and use an infra-red transmitter to convert the data to an infra-red signal. An infra-red detector can receive the data carried in the infra-red signal and appropriate circuitry can place the data on bus 112. Bus 112 carries the data to the main memory within storage devices 114, from which
15 processors 116 retrieves and executes the instructions.

[027] In some embodiments, when the data storage system receives data, the data storage system stores data in a memory and then saves the data into a physical storage after the amount of data reaches a threshold. **FIG. 2** is a schematic diagram illustrating an exemplary data storage system performing a save operation into a physical storage, according to some
20 embodiments of the present disclosure. It is appreciated that the save operation illustrated in **FIG. 2** can be performed by data storage system 100 of **FIG. 1** or server 110 of **FIG. 1**.

[028] As shown in **FIG. 2**, received data is first saved into a memory store M0 in memory (e.g., memory 161 of **FIG. 1**). In some embodiments, each entry in the received data

comprises a time stamp, indicating the time that the entry was saved into store M0. In some embodiments, memory store M0 is a key-value store organized as a rooted tree structure (e.g., LSM tree structure). When the size of memory store M0 reaches a certain threshold, memory store M0 is sealed. In other words, memory store M0 no longer accepts and saves update

5 operations. When memory store M0 is sealed, it becomes memory store M1, and a new memory store M0 is created to accept and save new update operations. Memory store M1 is then written into a physical storage (e.g., physical storage 164 of **FIG. 1**), shown as data file F1. In some embodiments, data file F1 is a key-value store organized as a rooted tree structure (e.g., B+ tree structure, LSM tree structure, etc.) or other sorted data structures.

10 [029] After data has been written into the physical storage, the data storage system may perform a hot-cold data separation using compaction processes. **FIG. 3** is a schematic diagram illustrating an exemplary data storage system incorporating hot-cold data separation, according to some embodiments of the present disclosure. According to **FIG. 3**, the physical storage comprises two levels, L1 and L2. Each level may host one or more data files, depicted as

15 triangles on **FIG. 3**. In some embodiments, on the basis of **FIG. 2**, after data file F1 in **FIG. 3** has been saved into the physical storage, the data storage system saves data file F1 into level L1 of the physical storage.

[030] The data storage system performs compaction processes systematically on some or all data in level L1. In some embodiments, the data storage system performs compaction

20 processes periodically, or according to a plan that is initiated by the system or a user of the system. During the compaction process, the data storage system combines records, which are identified by their keys, using a sorting process (e.g., merge sort). As a result, different versions of records that are identified by the same key is naturally brought together. The data storage

system then classifies the key according to the key's update frequency using a categorizer algorithm.

[031] In some embodiments, the categorizer algorithm can comprise a simple comparison process. For example, the data storage system may incorporate a categorizer threshold that may be a predefined number, or a number that can be set by a user of the data storage system. The categorizer algorithm reviews the time stamps on each version of the same key, and identifies how many times the record identified by the key has been updated within a time period. If the number of updates exceeds the categorizer threshold, the data storage system classifies the key as a first type of data (e.g., hot keys). If the number of updates is below the categorizer threshold, the data storage system classifies the key as a second type of data (e.g., cold keys). As a result, the second type of data is updated less frequently than the first type of data on average. In some embodiments, the time period used by the categorizing algorithm is equal to a time between each compaction process performed on level L1.

[032] In some embodiments, having determined whether a particular key is hot or cold, the data storage system then creates or finds two data files, one for writing the valid version(s) of records identified by hot keys, and the other for writing the valid version(s) of records identified by cold keys. After the compaction process finishes, the data file having cold keys are removed and rewritten into level L2. The data storage system performs compaction processes systematically on some or all data in level L2.

[033] In some embodiments, the data storage system performs compaction processes on some or all data in level L2 at a frequency that is lower than a frequency at which the data storage system performs compaction processes on some or all data in level L1. For example, the data storage system may perform compaction processes on some or all data in level L1 100 times

per hour, and perform compaction processes on some or all data in level L2 only 40 times per hour. Because the data storage system performs compaction processes at a lower frequency on some or all data in level L2, the data storage system reads and rewrites data files in L2 in and out of the physical storage less frequently. As a result, the data storage system can preserve valuable I/O bandwidth provided by the physical storage for any users or clients of the system. At the same time, since records in level L2 are not updated as frequently, the data storage system does not necessarily have to perform compaction processes on data in level L2 as frequently to create more storage space.

[034] The data storage system according to the embodiments of the present disclosure offer many other advantages over the existing systems. For example, the data storage system does not necessarily need a separate in-memory data structure to keep track of access patterns of each key. While the compaction process is being performed, the compaction process brings together different version of the same key naturally, and the compaction process can analyze the time stamps on the fly without having to save all the time stamps in a separate data structure. Moreover, the data storage system according to some embodiments of the present disclosure combines hot-key and cold-key classification and hot-cold separation in one process, further reducing the extra I/O bandwidth needed.

[035] In some embodiments, the data storage system monitors the categorizer threshold continuously and increases or decreases the categorizer threshold based on the efficiency of the system. For example, if the data storage system determines that the physical storage has been running out of spaces on a consistent basis, the data storage system may choose to decrease the categorizer threshold. After the categorizer threshold has been decreased, the categorizer algorithm classifies more keys as hot, leading to more records being rewritten into level L1.

Because compaction processes are performed in level L1 more frequently, obsolete versions in level L1 are removed more quickly before they can accumulate. As a result, the data storage system runs a lower risk of running out of physical storage space when the categorizer threshold is decreased. In another example, if the data storage system determines that the compaction processes are still taking up too much I/O bandwidth provided by the physical storage, the data storage system may choose to increase the categorizer threshold, leading to more records being rewritten in level L2. Because compaction processes are conducted in level L2 less frequently, the data storage system takes up less I/O bandwidth provided by the physical storage

[036] In some embodiments, the categorizer algorithm can comprise more sophisticated algorithms, such as logistic regression or artificial intelligence (e.g., neural networks). The categorizer algorithm can be updated constantly when the data storage system performs compaction processes systematically. For example, the categorizer algorithm may collect, in real time, information on the time stamps of the newest updates and use the information as training sets to adjust parameters of the logistic regressions or weights of the neural networks.

[037] In some embodiments, the ratio of frequencies in initiating and performing the compaction processes at different levels of the physical storage is associated with the categorizer algorithm. Categorizer thresholds can depend on the frequency at which a particular key is updated, which is directly associated with how often compaction processes may need to be performed on the data identified by the particular key. For example, if the data storage system performs compaction processes in level L1 at a frequency that is twice the frequency at which the data storage system performs compaction processes in level L2, the categorizer threshold used in the categorizer algorithm for L1 may be set at 2 times the categorizer threshold used in the categorizer algorithm for L2. In another example, the categorizer threshold used in the

categorizer algorithm for L1 may be set so that hot data is, on average, twice as likely to get updated in a given period of time than cold data.

[038] In some embodiments, each level of the physical storage can comprise an eviction space to store newly arrived data files. **FIG. 4** is a schematic diagram illustrating an exemplary data storage system incorporating hot-cold data separation using eviction spaces, according to some embodiments of the present disclosure. According to **FIG. 4**, each level of physical storage (e.g., level L1 and level L2) can have an eviction space. The eviction spaces host newly arrived data files. For example, when data file F1 is saved into level L1 of the physical storage, it is first saved into the eviction space of level L1. When compaction processes are performed on some or all data in level L1, the compaction processes combine data file F1, together with other data files in the eviction space, into one or more new data files. The compaction processes then write the new data files that are reserved for level L1 into a space outside of the eviction space in level L1. In another example, when data files comprising of cold data in level L1 is rewritten into level L2, these data files are first rewritten into the eviction space of level L2. When compaction processes are performed on some or all data in level L2, the compaction processes combine these data files into one or more new data files. The compaction processes then write the new data files that are reserved for level L2 into a space outside of the eviction space in level L2.

[039] One advantage of having eviction spaces in each level of the physical storage is that the eviction spaces contain newly updated data files. When the data storage system needs to perform a read operation, the data storage system can search the data files in the eviction spaces first. This allows the data storage system to perform read operations more efficiently.

[040] In some embodiments, the data storage system comprises more than 2 levels in the physical storage. **FIG. 5** is a schematic diagram illustrating an exemplary data storage system

incorporating hot-cold data separation using multiple levels, according to some embodiments of the present disclosure. According to **FIG. 5**, the physical storage comprises a plurality of levels L1 to Ln. Each level may host one or more data files, depicted as triangles on **FIG. 5**. In some embodiments, on the basis of **FIG. 2** and **FIG. 3**, after data file F1 in **FIG. 5** has been saved into the physical storage, the data storage system saves data file F1 into level L1 of the physical storage.

[041] In some embodiments, when the data storage system performs the compaction process on some or all data in level L2, the data storage system further identifies whether the key is hot or cold using a categorizer algorithm through the compaction process. In some embodiments, the categorizer algorithm used for data files in level L2 differs from the categorizer algorithm used for data files in level L1. For example, the categorizer algorithm used for data files in level L2 may have a different categorizer threshold than the categorizer threshold used in the categorizer algorithm for data files in level L1. The categorizer threshold used for level L2 may be lower than the categorizer threshold used for level L1, because data in data files in level L2 may be updated less frequently than data in data files in level L1.

[042] In some embodiments, having determined whether a particular key in level L2 is hot or cold, the data storage system then creates or finds two data files, one for writing the valid version(s) of records identified by hot keys, and the other for writing the valid version(s) of records identified by cold keys. After the compaction process finishes, the data file having cold keys are moved into level L3. The data storage system performs compactions processes systematically on some or all data in level L3.

[043] In some embodiments, the data storage system performs compaction processes on some or all of data files in level L3 at a frequency that is lower than the frequencies at which the

data storage system performs compaction processes on some or all of data files in level L1 or in level L2. For example, the data storage system may perform compaction processes on some or all data files in level L2 40 times per hour and may perform compaction processes on some or all of data files in level L3 only 25 times per hour. Because the data storage system performs

5 compaction processes at a lower frequency on some or all of data files in level L3, the data storage system reads and rewrites data files in L3 on the physical storage less frequently. As a result, the data storage system can preserve valuable I/O bandwidth provided by the physical storage for users or clients of the system. At the same time, since the records in data files in level L3 are not updated nearly as frequently as those in L1 and L2, the data storage system does not

10 necessarily need to perform compaction processes on data files in level L3 as frequently to prevent the system from running out of storage space.

[044] In some embodiments, when the data storage system performs the compaction process on some or all of data files in level L2, the data storage system further identifies whether the key is hot, cold, or moderate using a categorizer algorithm through the compaction process.

15 For example, the categorizer algorithm used for data files in L2 may use two different categorizer thresholds, with one categorizer threshold larger than the other. The compaction process reviews the time stamps on each version of the same key and identifies how many times the record identified by the key has been updated within a time period. If the number of updates exceed the larger categorizer threshold, the data storage system would identify the key as hot. If

20 the number of updates is in between the larger categorizer threshold and the smaller categorizer threshold, the data storage system would identify the key as moderate. If the number of updates is below the smaller categorizer threshold, the data storage system would identify the key as

cold. In some embodiments, the time period is equal to the time in between each compaction process on level L2.

[045] In some embodiments, having determined whether a particular key is hot, moderate, or cold, the data storage system then creates or finds three data files, one for writing the valid version(s) of records identified by hot keys, one for writing the valid version(s) of records identified by moderate keys, and one for writing the valid version(s) of records identified by cold keys. After the compaction process finishes, the data file having cold keys is moved into level L3, and the data file having hot keys is moved into level L1.

[046] In some embodiments, the data storage system performs compaction processes on some or all data files in levels L3 to L_n in a similar fashion as the compaction processes performed on some or all data files in level L2. The data storage system may perform compaction processes in a particular level at a frequency that is lower than the frequency at which the system performs compaction processes in the level above. For example, if the compaction processes are performed 15 times per hour at level L_m, the data storage system may perform compaction processes at a frequency that is smaller than 15 times per hour at level L_{m+1} (e.g., 10 times per hour). Because the frequency of initiating and performing compaction processes decreases in each level, the data storage system reads and rewrites data on the physical storage less frequently. As a result, the data storage system can preserve valuable I/O bandwidth provided by the physical storage for any users or clients of the system. At the same time, since the records in data files are updated less frequently in each level, the data storage system does not need to perform compaction processes on data files in each level very frequently to prevent the system from running out of storage space.

[047] In some embodiments, the categorizer algorithms being performed at each level may be of the same type (e.g., a simple comparison), but the categorizer thresholds are different among different levels. For example, the categorizer threshold for level L1 may be 50 updates per minute, and the categorizer threshold for level L2 is most likely smaller than 50 (e.g., 10 updates per minute) because records in level L2 are generally updated less frequently than records in level L1.

[048] Embodiments of the present disclosure further provides a method for hot-cold separation in data storage systems. **FIG. 6** illustrates a flow diagram of an exemplary method 1000 for hot-cold data separation, according to some embodiments of the present disclosure. It is appreciated that method 1000 can be performed by a data storage system (e.g., data storage system 100 of **FIG. 1**) or by one or more servers (e.g., exemplary server 110 of **FIG. 1**).

[049] In step 1010, a data storage system receives data and organizes the data into a data structure having key-value pairings in memory. For example, as shown in **FIG. 2**, data is received and first saved into memory store M0 in memory.

[050] In step 1020, the data storage system stores the data into a first level of a physical storage after the size of the data structure reaches a threshold. For example, as shown in **FIG. 2**, memory store M0 is sealed after the size of memory store M0 reaches a certain threshold. Memory store M0 then becomes memory store M1, which is written into the physical storage. When memory store M1 is written into physical storage, it becomes data file F1. As shown in **FIG.3**, data file F1 is stored into level L1 of the physical storage. In some embodiments, when data is written into the first level of the physical storage, it is written into an eviction space of the first level of the physical storage. For example, when memory store M1 is written into physical

storage, it becomes data file F1. As shown in **FIG. 4**, data file F1 is stored into the eviction space of level L1.

[051] In step 1030, the data storage system performs compaction processes systematically on all or some data in the first level of the physical storage and classifies the data into a first type of data (e.g., hot data) and a second type of data (e.g., cold data) based on update frequency. For example, as shown in **FIG. 3**, compaction processes are performed periodically on some or all data files in level L1. The categorizer algorithm can comprise a simple comparison process or a more sophisticated process involving logistic regressions or artificial intelligence (e.g., neural networks).

[052] In some embodiments, the categorizer algorithm incorporates a categorizer threshold. For example, as shown in **FIG. 3** and **FIG. 4**, the categorizer algorithm reviews and identifies how many times the record identified by a particular key has been updated within a time period. If the number of updates is larger than the categorizer threshold, the data storage system classifies the key as hot. If the number of updates is below the categorizer threshold, the data storage system classifies the key as cold. In some embodiments, as shown in **FIG. 3** and **FIG. 4**, the data storage system monitors the categorizer threshold continuously, and increases or decreases the categorizer threshold based on the efficiency of the system. In some embodiments, as shown in **FIG. 3** and **FIG. 4**, the categorizer algorithm can comprise more sophisticated algorithms, such as logistic regression or artificial intelligence.

[053] Referring back to **FIG. 6**, in step 1040, the data storage system removes the cold data from the first level of the physical storage and rewrites the cold data into a second level of the physical storage. For example, as shown in **FIG. 3**, the data storage system creates or finds two data files, one for writing the valid version(s) of records identified by hot keys, and the other

for writing the valid version(s) of records identified by cold keys. The data file having cold keys are rewritten into level L2.

[054] In step 1050, the data storage system performs compaction processes systematically on all or some data in the second level of the physical storage at a lower frequency than the compaction processes on data in the first level. For example, as shown in **FIG. 3**, the data storage system performs compaction processes periodically on some or all of data files in level L2 at a frequency that is lower than a frequency at which the data storage system performs compaction processes on some or all of data files in level L1.

[055] In some embodiments, the data storage system classifies the data in the second level into a third type of data (e.g., hot data) and a fourth type of data (e.g., cold data) based on update frequency during the compaction processes and removes the hot data from the second level and rewrite the hot data into the first level of the physical storage. In some embodiments, the ratio of frequencies in initiating and performing the compaction processes in different levels of the physical storage is associated with the categorizer algorithm. In some embodiments, the data storage system removes the hot data in the second level and rewrites the hot data into the first level.

[056] In some embodiments, method 1000 further comprises additional steps involving more than two levels of physical storage. **FIG. 7** illustrates a flow diagram of an exemplary method 1000 for hot-cold data separation in more than two levels of physical storage, according to some embodiments of the present disclosure. On the basis of **FIG. 6**, method 1000 further comprises steps 1060, 1070 and 1080. It is appreciated that method 1000 can be performed by a data storage system (e.g., data storage system 100 of **FIG. 1**) or by one or more servers (e.g., exemplary server 110 of **FIG. 1**).

[057] In step 1060, the data storage system separates data in the second level of the physical storage into a third type of data (e.g., hot data) and a fourth type of data (e.g., cold data) based on update frequency. For example, as shown in **FIG. 5**, the data storage system identifies whether keys of records in data files in level L2 are hot or cold using a categorizer algorithm.

5 [058] In some embodiments, the categorizer algorithms being performed at each level may be of the same type (e.g., a simple comparison), but the categorizer thresholds are different among different levels. For example, as shown in **FIG. 5**, the categorizer threshold for level L1 may be 50 updates since the last compaction, and the categorizer threshold for level L2 is most likely smaller than 50 (e.g., 10 updates since the last compaction) because records in level L2 are
10 generally updated less frequently than records in level L1.

[059] Referring back to **FIG. 7**, in step 1070, the data storage system removes cold data from the second level of the physical storage and rewrites the cold data into a third level of the physical storage. For example, as shown in **FIG. 5**, the data storage system creates or finds two data files, one for writing the valid version(s) of records identified by hot keys, and the other for
15 writing the valid version(s) of records identified by cold keys. The data file having cold keys are moved into level L3.

[060] In step 1080, the data storage system performs steps 1050, 1060, and 1070 for some or all other levels of the physical storage, wherein the frequency at which compaction processes are performed in a particular level is lower than the level above. For example, as
20 shown in **FIG. 5**, the data storage system performs compaction processes on some or all data files in level L3 to Ln in similar fashion as the compaction processes performed on some or all data files in level L2. In some embodiments, the physical storage comprises N levels. The first level of the physical storage is an M level of the physical storage, and the second level of the

physical storage is an $M+1$ level of the physical storage. M and N are integers, and $0 < M < N$. The data storage system performs step 1050, 1060 and 1070 on the M level of the physical storage in the same manner as the second level of the physical storage.

[061] In some embodiments, method 1000 further comprises additional steps involving
5 a more sophisticated separation of data. **FIG. 8** illustrates a flow diagram of an exemplary method 1000 for hot-moderate-cold data separation in more than two levels of physical storage, according to some embodiments of the present disclosure. On the basis of **FIG. 6**, method 1000 further comprises steps 1065, 1070, 1075 and 1085. It is appreciated that method 1000 can be performed by a data storage system (e.g., data storage system 100 of **FIG. 1**) or by one or more
10 servers (e.g., exemplary server 110 of **FIG. 1**).

[062] In step 1065, the data storage system separates data in the second level of the physical storage into a third type of data (e.g., hot data), a fourth type of data (e.g., cold data) or a fifth type of data (e.g., moderate data) based on update frequency. For example, as shown in **FIG. 5**, when the data storage system performs the compaction process on some or all of data
15 files in level L2, the data storage system further identifies whether a key for each record is hot, cold, or moderate using a categorizer algorithm.

[063] In some embodiments, the categorizer algorithm uses two categorizer thresholds to separate data into hot, moderate, or cold. For example, as shown in **FIG. 5**, the categorizer algorithm has two categorizer thresholds, a larger categorizer threshold and a smaller categorizer
20 threshold. The compaction process identifies a key as hot if the number of updates on the key is larger than the larger threshold. The compaction process identifies a key as moderate if the number of updates on the key is in between the larger threshold and the smaller threshold. The

compaction process identifies a key as cold if the number of updates on the key is smaller than the smaller threshold.

[064] Referring back to **FIG. 8**, in step 1070, the data storage system removes cold data from the second level of the physical storage and rewrites the cold data into a third level of the physical storage. For example, as shown in **FIG. 5**, the data storage system creates or finds three data files, one for writing the valid version(s) of records identified by hot keys, one for writing the valid version(s) of records identified by moderate keys, and the one for writing the valid version(s) of records identified by cold keys. The data file having cold keys are moved into level L3.

[065] In step 1075, the data storage system removes hot data from the second level of the physical storage and rewrites the hot data into the first level of the physical storage. For example, as shown in **FIG. 5**, the data storage system creates or finds three data files, one for writing the valid version(s) of records identified by hot keys, one for writing the valid version(s) of records identified by moderate keys, and one for writing the valid version(s) of records identified by cold keys. After the compaction finishes, the data file having hot keys are moved and rewritten into level L1, and the data file having cold keys are moved and rewritten into level L3.

[066] In step 1085, the data storage system performs steps 1050, 1065, 1070 and 1075 for some or all other levels of the physical storage, wherein the frequency at which compaction processes are performed in a particular level is smaller than the level above. For example, as shown in **FIG. 5**, the data storage system performs compaction processes on some or all data files in level L3 to Ln in similar fashion as the compaction processes performed on some or all data files in level L2.

[067] The various example embodiments described herein are described in the general context of method steps or processes, which may be implemented in one aspect by a computer program product, embodied in a computer-readable medium, including computer-executable instructions, such as program code, executed by computers in networked environments. In some
5 embodiments, a data storage system (e.g., data storage system 100 of **FIG. 1**) may instruct components (e.g., physical storage 164 of **FIG. 1**) of computer system (e.g., server 110 of **FIG. 1**) to perform various functions described above, such as establishing categorizer algorithms, initiating and performing compaction processes, etc. A computer-readable medium may include removeable and nonremoveable storage devices including, but not limited to, Read Only Memory
10 (e.g., ROM 163 of **FIG. 1**), Random Access Memory (e.g., RAM 162 of **FIG. 1**), compact discs (CDs), digital versatile discs (DVD), etc. Generally, program modules may include routines, programs, objects, components, data structures, etc. that perform particular tasks or implement particular abstract data types. Computer-executable instructions, associated data structures, and program modules represent examples of program code for executing steps of the methods
15 disclosed herein. The particular sequence of such executable instructions or associated data structures represents examples of corresponding acts for implementing the functions described in such steps or processes.

[068] In the foregoing specification, embodiments have been described with reference to numerous specific details that can vary from implementation to implementation. Certain
20 adaptations and modifications of the described embodiments can be made. Other embodiments can be apparent to those skilled in the art from consideration of the specification and practice of the embodiments disclosed herein. It is intended that the specification and examples be considered as exemplary only, with a true scope and spirit of the invention being indicated by the

following claims. It is also intended that the sequence of steps shown in figures are only for illustrative purposes and are not intended to be limited to any particular sequence of steps. As such, those skilled in the art can appreciate that these steps can be performed in a different order while implementing the same method.

5 [069] As used herein, unless specifically stated otherwise, the term “or” encompasses all possible combinations, except where infeasible. For example, if it is stated that a database may include A or B, then, unless specifically stated otherwise or infeasible, the database may include A, or B, or A and B. As a second example, if it is stated that a database may include A, B, or C, then, unless specifically stated otherwise or infeasible, the database may include A, or B,
10 or C, or A and B, or A and C, or B and C, or A and B and C.

 [070] In the drawings and specification, there have been disclosed exemplary embodiments. Many variations and modifications, however, can be made to these embodiments. Accordingly, although specific terms are employed, they are used in a generic and descriptive sense only and not for purposes of limitation, the scope of the embodiments being defined by the
15 following claims.

WHAT IS CLAIMED IS:

1. A method for data storage system, comprising:
storing data into a first level of a physical storage;
initiating compaction processes on data in the first level of the physical storage;
5 classifying, during the compaction processes, the data into a first group of data and a
second group of data based on update frequency;
removing the second group of data from the first level of the physical storage; and
rewriting the second group of data into a second level of the physical storage,
wherein a frequency of initiating compaction processes on data in the first level of the
10 physical storage is higher than a frequency of initiating compaction processes on data in the
second level of the physical storage.
2. The method according to claim 1, wherein the second group of data is updated
less frequently on average than the first group of data.
- 15 3. The method according to claim 1 or 2, further comprising initiating compaction
processes on data in the second level of the physical storage.
4. The method according to claim 3, wherein initiating compaction processes on data
20 in the second level of the physical storage further comprises:
classifying, during the compaction processes, the data in the second level into a third
group of data and a fourth group of data based on updating frequency;
removing the third group of data from the second level of the physical storage; and

rewriting the third group of data into the first level of the physical storage.

5. The method according to any one of claims 1-4, wherein the physical storage comprises N levels, the first level of the physical storage is an M level of the physical storage, and the second level of the physical storage is an M+1 level of the physical storage, wherein M and N are integers and $0 < M < N$.

6. The method according to any one of claims 1-5, wherein classifying, during the compaction processes, the data into a first group of data and a second group of data based on update frequency further comprises:

determining if a number of write operations on a particular key that is associated with the data satisfies a categorizer threshold;

classifying the data associated with the particular key as the first group of data in response to the determination that the number of write operations on the particular key satisfies the categorizer threshold; or

classifying the data associated with the particular key as the second group of data in response to the determination that the number of write operations on the particular key does not satisfy the categorizer threshold.

7. The method according to claim 5, further comprising:
separating the data in the M level of the physical storage into the first group of data, the second group of data or a fifth group of data based on update frequency during the compaction processes;

removing the second data from the M level of the physical storage;
rewriting the second data into an M+1 level of the physical storage;
removing the first data from the M level of the physical storage; and
rewriting the first data into an M-1 level of the physical storage.

5

8. The method according to claim 7, further comprising:

determining if a number of write operations on a particular key that is associated with the data satisfies a first categorizer threshold;

classifying the data associated with the particular key as the first group of data in

10 response to a determination that the number of write operations on the particular key larger than the first categorizer threshold;

determining if a number of write operations on the particular key is between the first categorizer threshold and a second categorizer threshold, wherein the first categorizer threshold is larger than the second categorizer threshold;

15 classifying the data associated with the particular key as the fifth group of data in response to a determination that the number of write operations on the particular key is between the first categorizer threshold and the second categorizer threshold; and

classifying the data associated with the particular key as the second group of data in response to a determination that the number of write operations on the particular key is below the
20 second categorizer threshold.

9. The method according to any one of claims 1-8, wherein storing data into the first level of the physical storage further comprises:

organizing the data into a data structure having key-value pairings in memory; and
storing the data into the first level of the physical storage after the size of the data reaches
a threshold.

5 10. The method according to any one of claims 1-9, wherein the ratio of frequencies
in initiating the compaction processes at different levels of the physical storage is associated with
the categorizing thresholds.

 11. The method according to any one of claims 1-10, further comprising:
10 storing newly arrived data in each level of the physical storage into an eviction space
within the level.

 12. The method according to any one of claims 1-11, wherein removing the second
group of data from the first level of the physical storage and rewriting the second group of data
15 into a second level of the physical storage further comprises:

 creating a new data file for the first group of data and another data file for the second
group of data; and
writing the data file for the second group of data into the second level of the physical
storage.

20

 13. A data storage system, comprising:
a physical storage comprising a first level and a second level;
a memory storing a set of instructions; and

a processor configured to execute the set of instructions to cause the data storage system to:

store data in into the first level of the physical storage;

initiate compaction processes on data in the first level of the physical storage;

5 classify, during the compaction processes, the data into a first group of data and a second group of data based on update frequency;

remove the second group of data from the first level of the physical storage; and

rewrite the second group of data into the second level of the physical storage;

10 wherein a frequency of initiating compaction processes on data in the first level of the physical storage is higher than a frequency of initiating compaction processes on data in the second level of the physical storage.

14. The data storage system according to claim 13, wherein the second group of data is updated less frequently on average than the first group of data.

15

15. The data storage system according to claim 13 or 14, wherein the processor is further configured to initiate compaction processes on data in the second level of the physical storage.

20

16. The data storage system according to claim 15, wherein the processor is further configured to cause the data storage system to:

classify, during the compaction processes, the data in the second level into a third group of data and a fourth group of data based on updating frequency;

remove the third group of data from the second level of the physical storage; and

rewrite the third group of data into the first level of the physical storage.

17. The data storage system according to any one of claims 13-16, wherein the physical storage comprises N levels, the first level of the physical storage is an M level of the physical storage, and the second level of the physical storage is an M+1 level of the physical storage, wherein M and N are integers and $0 < M < N$.

18. The data storage system according to any one of claims 13-17, wherein the processor is further configured to cause the data storage system to:

10 determine if a number of write operations on a particular key that is associated with the data satisfies a categorizer threshold;

classify the data associated with the particular key as the first group of data in response to the determination that the number of write operations on the particular key satisfies the categorizer threshold; or

15 classify the data associated with the particular key as the second group of data in response to the determination that the number of write operations on the particular key does not satisfy the categorizer threshold.

19. The data storage system according to claim 17, wherein the processor is further

20 configured to cause the data storage system to:

separate the data in the M level of the physical storage into the first group of data, the second group of data or a fifth group of data based on update frequency during the compaction processes;

remove the second data from the M level of the physical storage;
rewrite the second data into an M+1 level of the physical storage;
remove the first data from the M level of the physical storage; and
rewrite the first data into an M-1 level of the physical storage.

5

20. The data storage system according to claim 19, wherein the processor is further configured to cause the data storage system to:

determine if a number of write operations on a particular key that is associated with the data satisfies a first categorizer threshold;

10 classify the data associated with the particular key as the first group of data in response to a determination that the number of write operations on the particular key larger than the first categorizer threshold;

15 determine if a number of write operations on the particular key is between the first categorizer threshold and a second categorizer threshold, wherein the first categorizer threshold is larger than the second categorizer threshold;

 classify the data associated with the particular key as the fifth group of data in response to a determination that the number of write operations on the particular key is between the first categorizer threshold and the second categorizer threshold; and

20 classify the data associated with the particular key as the second group of data in response to a determination that the number of write operations on the particular key is below the second categorizer threshold.

21. The data storage system according to any one of claims 13-20, wherein the processor is further configured to cause the data storage system to:

organizing the data into a data structure having key-value pairings in the memory; and
store the data into the first level of the physical storage after the size of the data reaches a
5 threshold.

22. The data storage system according to any one of claims 13-21, wherein the processor is further configured to cause the data storage system to:

store newly arrived data in each level of the physical storage into an eviction space within
10 the level.

23. The data storage system according to any one of claims 13-22, wherein the processor is further configured to cause the data storage system to:

create a new data file for the first group of data and another data file for the second type
15 of data; and

write the data file for the second group of data into the second level of the physical
storage.

24. A non-transitory computer readable medium that stores a set of instructions that is
20 executable by one or more processors of an apparatus to cause the apparatus to initiate a method
comprising:

storing data in into a first level of the physical storage;

initiating compaction processes on data in the first level of the physical storage;

classifying, during the compaction processes, the data into a first group of data and a second group of data based on update frequency;

removing the second group of data from the first level of the physical storage; and

rewriting the second group of data into a second level of the physical storage;

5 wherein a frequency of initiating compaction processes on data in the first level of the physical storage is higher than a frequency of initiating compaction processes on data in the second level of the physical storage.

25. The non-transitory computer readable medium according to claim 24, wherein the
10 second group of data is updated less frequently on average than the first group of data.

26. The non-transitory computer readable medium according to claim 24 or 25, wherein the set of instructions that is executable by one or more processors of the apparatus to cause the apparatus to further perform:

15 initiating compaction processes on data in the second level of the physical storage.

27. The non-transitory computer readable medium according to claim 26, wherein initiating compaction processes on data in the second level of the physical storage further comprises:

20 classifying, during the compaction processes, the data in the second level into a third group of data and a fourth group of data based on updating frequency;

removing the third group of data from the second level of the physical storage; and

rewriting the third group of data into the first level of the physical storage.

28. The non-transitory computer readable medium according to any one of claims 24-27, wherein the physical storage comprises N levels, the first level of the physical storage is an M level of the physical storage, and the second level of the physical storage is an M+1 level of the physical storage, wherein M and N are integers and $0 < M < N$.

29. The non-transitory computer readable medium according to any one of claims 24-28, wherein classifying, during the compaction processes, the data into a first group of data and a second group of data based on update frequency further comprises:

determining if a number of write operations on a particular key that is associated with the data satisfies a categorizer threshold;

classifying the data associated with the particular key as the first group of data in response to the determination that the number of write operations on the particular key satisfies the categorizer threshold; or

classifying the data associated with the particular key as the second group of data in response to the determination that the number of write operations on the particular key does not satisfy the categorizer threshold.

30. The non-transitory computer readable medium according to claim 29, wherein the set of instructions that is executable by one or more processors of the apparatus to cause the apparatus to further perform:

separating the data in the M level of the physical storage into the first group of data, the second group of data or a fifth group of data based on update frequency during the compaction processes;

removing the second data from the M level of the physical storage;

5 rewriting the second data into an M+1 level of the physical storage;

removing the first data from the M level of the physical storage; and

rewriting the first data into an M-1 level of the physical storage.

31. The non-transitory computer readable medium according to claim 30, wherein the
10 set of instructions that is executable by one or more processors of the apparatus to cause the apparatus to further perform:

determining if a number of write operations on a particular key that is associated with the data satisfies a first categorizer threshold;

15 classifying the data associated with the particular key as the first group of data in response to a determination that the number of write operations on the particular key satisfies the first categorizer threshold;

determining if a number of write operations on the particular key is between the first categorizer threshold and a second categorizer threshold, wherein the first categorizer threshold is larger than the second categorizer threshold;

20 classifying the data associated with the particular key as the fifth group of data in response to a determination that the number of write operations on the particular key is between the first categorizer threshold and the second categorizer threshold; and

classifying the data associated with the particular key as the second group of data in response to a determination that the number of write operations on the particular key is below the second categorizer threshold.

5 32. The non-transitory computer readable medium according to any one of claims 24-31, wherein storing data into the first level of the physical storage further comprises:

 organizing the data into a data structure having key-value pairings in memory; and
 storing the data into the first level of the physical storage after the size of the data reaches
a threshold.

10 33. The non-transitory computer readable medium according to any one of claims 24-32, wherein the set of instructions that is executable by one or more processors of the apparatus to cause the apparatus to further perform:

 storing newly arrived data in each level of the physical storage into an eviction space
15 within the level.

 34. The non-transitory computer readable medium according to any one of claims 24-33, wherein removing the second group of data from the first level of the physical storage and rewriting the second group of data into a second level of the physical storage further comprises:

20 creating a new data file for the first group of data and another data file for the second group of data; and

 writing the data file for the second group of data into the second level of the physical storage.

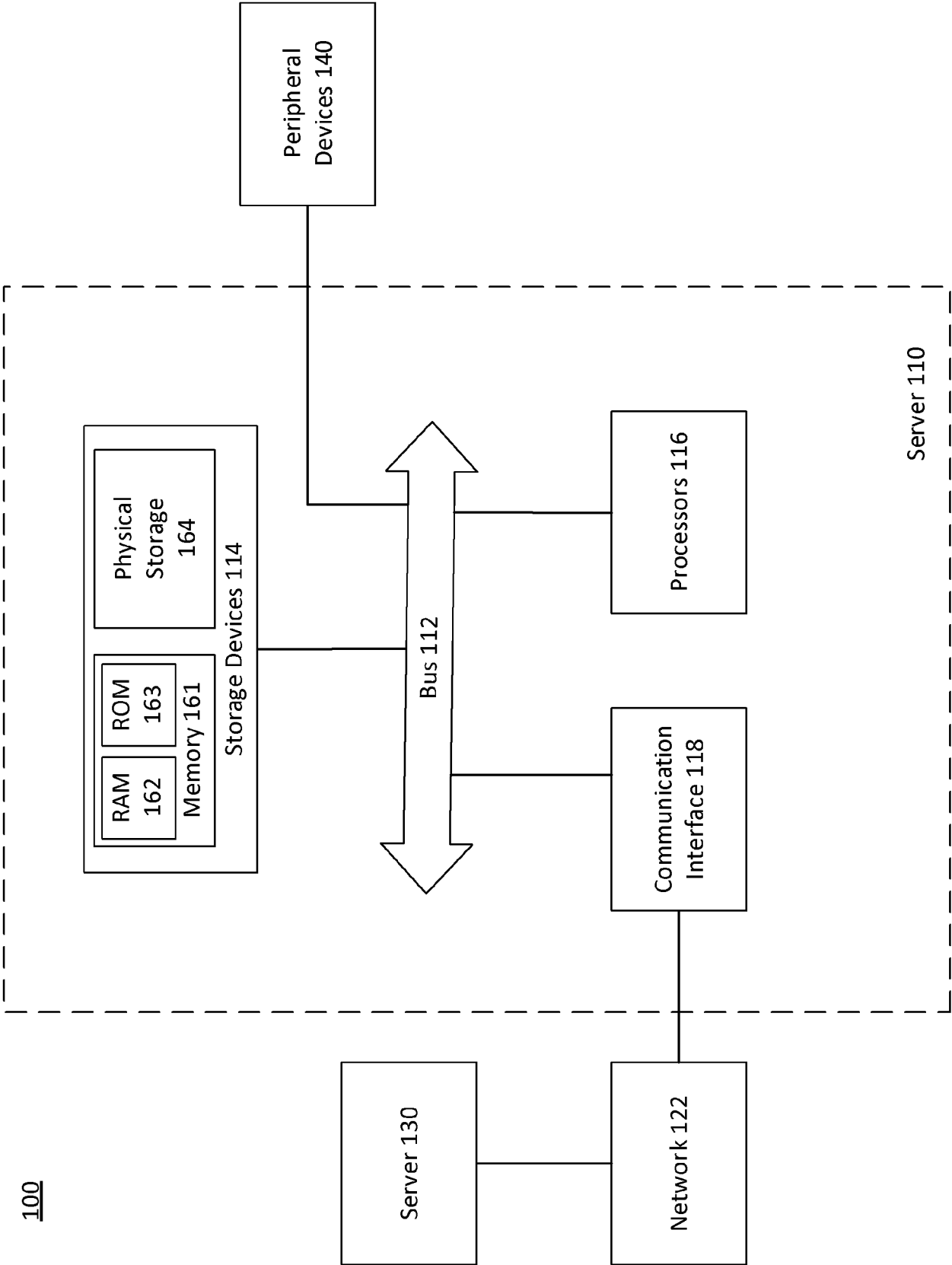


FIG. 1

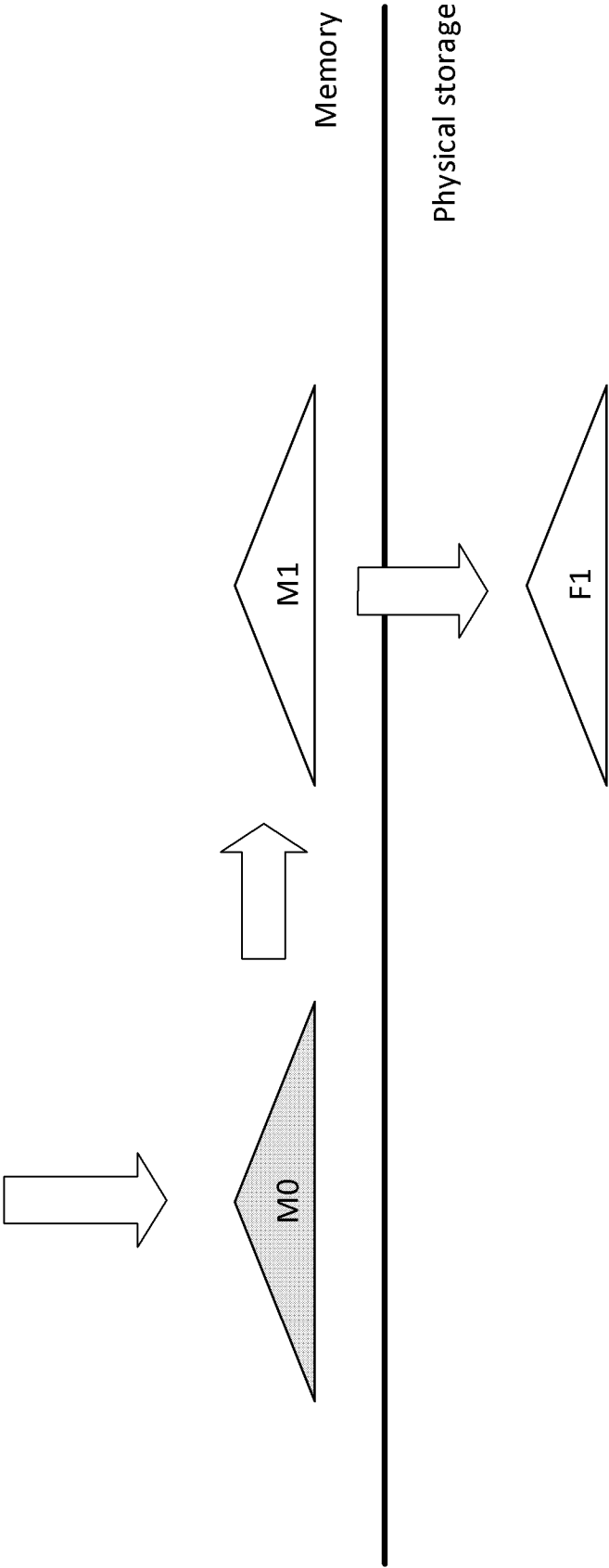


FIG. 2

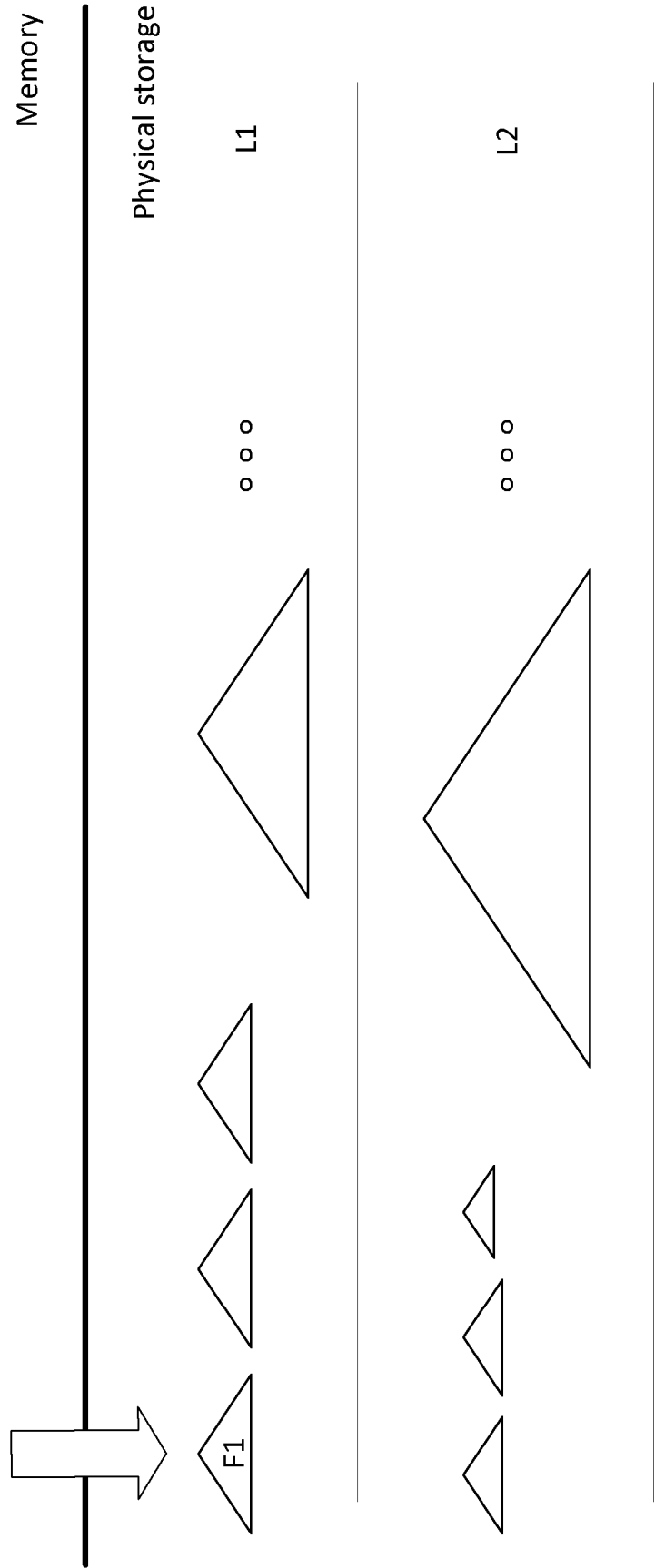


FIG. 3

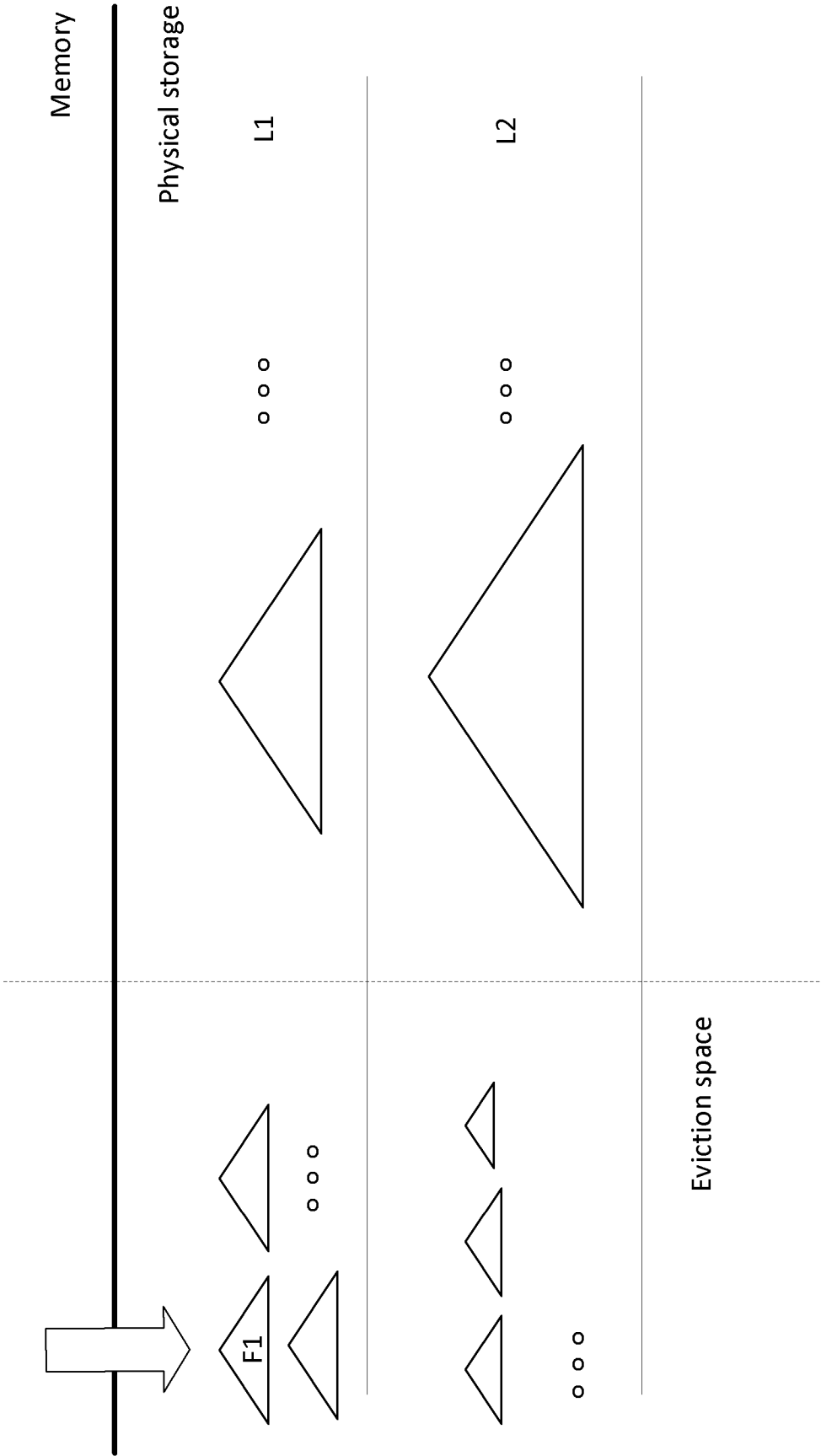


FIG. 4

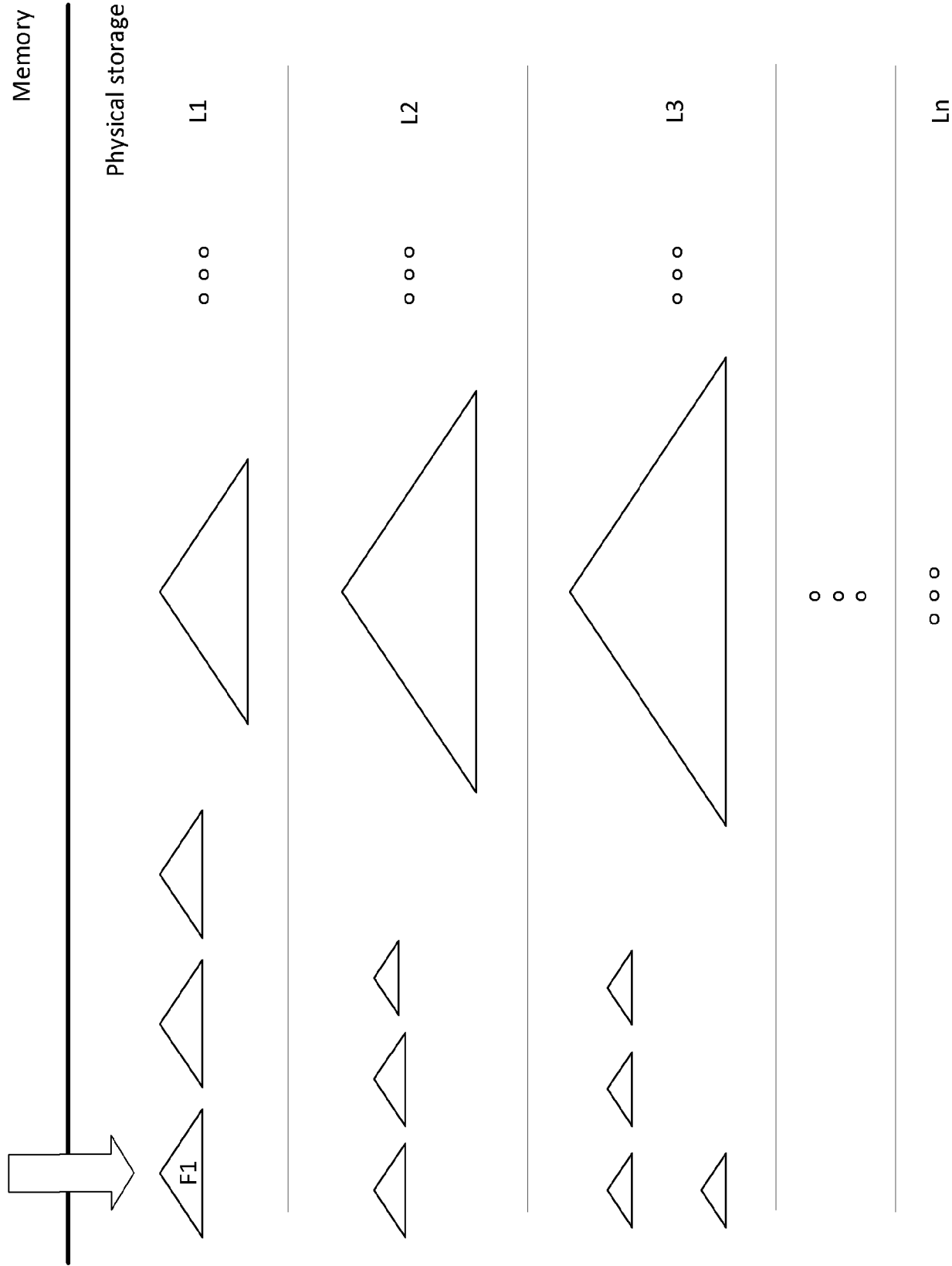


FIG. 5

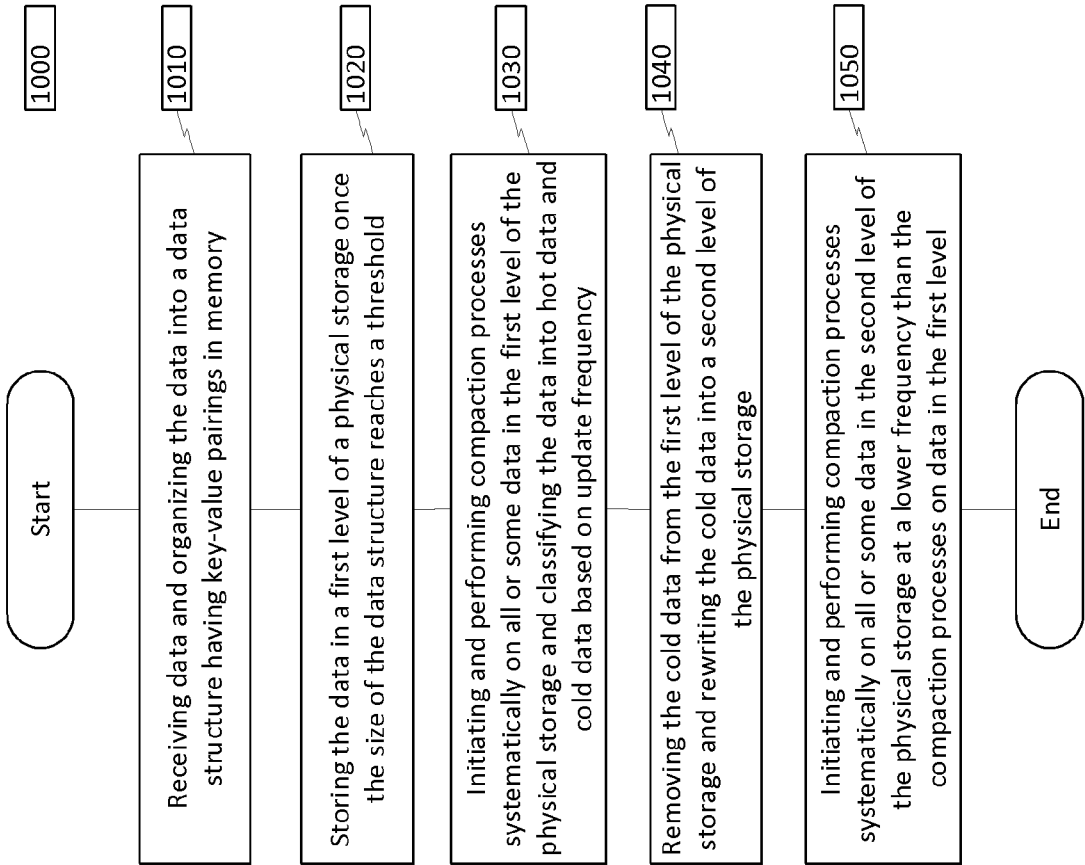


FIG. 6

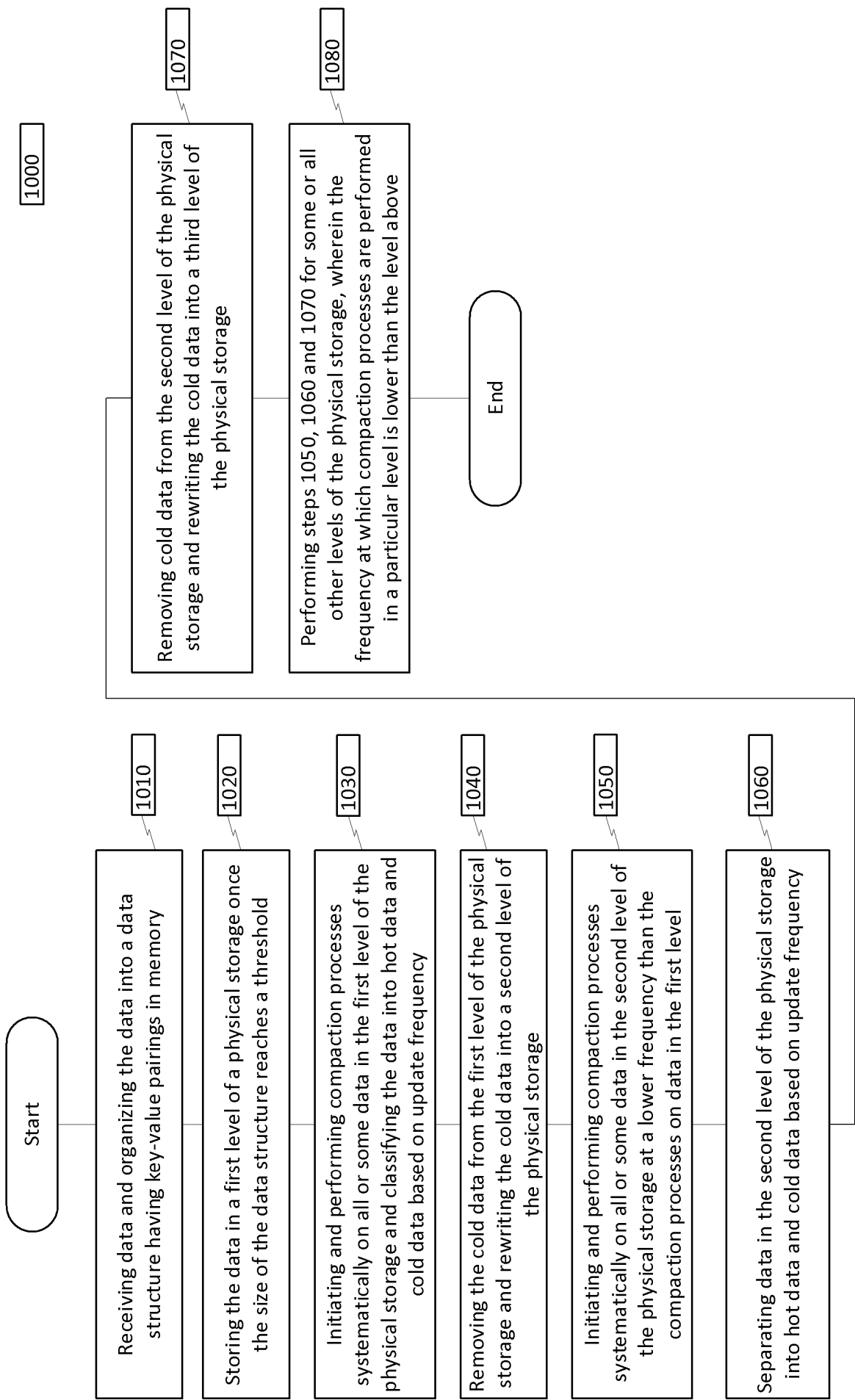


FIG. 7

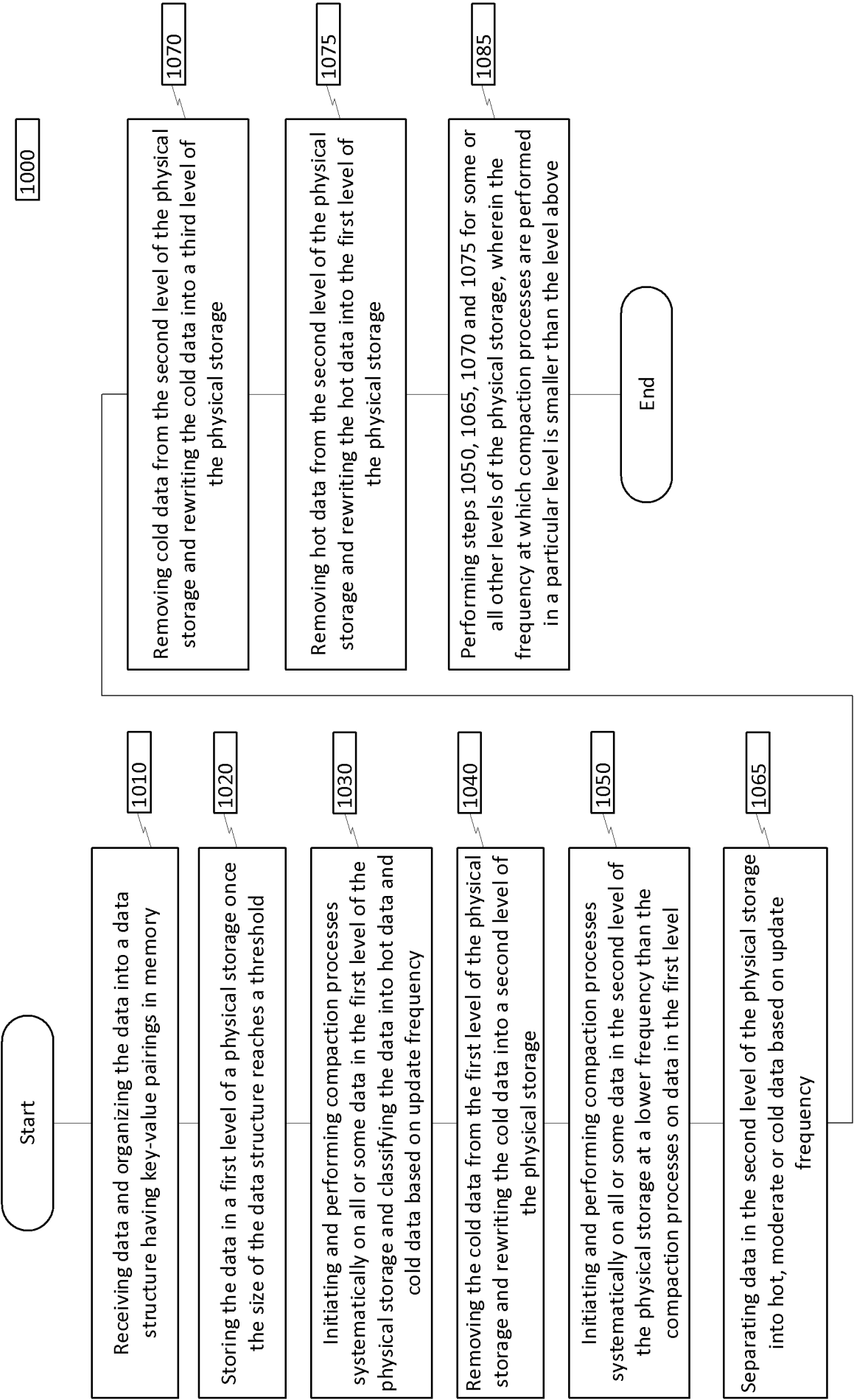


FIG. 8

INTERNATIONAL SEARCH REPORT

International application No.

PCT/CN2019/096857

A. CLASSIFICATION OF SUBJECT MATTER

G06F 16/00(2019.01)i

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

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)

G06F

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

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

CNPAT, WPI, EPODOC, CNKI, IEEE: LSM, log, structure, merge, key, value, compact, update, write, frequency, threshold, hot, cold, group, classify, move, reduce, amplification, level, layer

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
Y	CN 105447059 A (HUAWEI TECHNOLOGIES CO., LTD. et al.) 30 March 2016 (2016-03-30) description, paragraphs [0065]-[0090]	1-34
Y	CN 109857725 A (BEIJING BAIDU NETCOM SCIENCE & TECHNOLOGY CO., LTD.) 07 June 2019 (2019-06-07) description, paragraphs [0067]-[0082]	1-34
A	CN 107908358 A (RAMAXEL TECHNOLOGY SHENZHEN CO., LTD.) 13 April 2018 (2018-04-13) the whole document	1-34
A	US 2013080689 A1 (SAMSUNG ELECTRONICS CO., LTD.) 28 March 2013 (2013-03-28) the whole document	1-34
A	US 2016011971 A1 (LEE, Jae-il et al.) 14 January 2016 (2016-01-14) the whole document	1-34



Further documents are listed in the continuation of Box C.



See patent family annex.

* Special categories of cited documents:

“A” document defining the general state of the art which is not considered to be of particular relevance

“E” earlier application or patent but published on or after the international filing date

“L” document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)

“O” document referring to an oral disclosure, use, exhibition or other means

“P” document published prior to the international filing date but later than the priority date claimed

“T” later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention

“X” document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone

“Y” document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art

“&” document member of the same patent family

Date of the actual completion of the international search

24 March 2020

Date of mailing of the international search report

17 April 2020

Name and mailing address of the ISA/CN

National Intellectual Property Administration, PRC
6, Xitucheng Rd., Jimen Bridge, Haidian District, Beijing
100088
China

Authorized officer

YANG, Jishuang

Facsimile No. (86-10)62019451

Telephone No. 86-(10)-53961357

INTERNATIONAL SEARCH REPORT
Information on patent family members

International application No.

PCT/CN2019/096857

Patent document cited in search report			Publication date (day/month/year)	Patent family member(s)			Publication date (day/month/year)
CN	105447059	A	30 March 2016	None			
CN	109857725	A	07 June 2019	None			
CN	107908358	A	13 April 2018	None			
US	2013080689	A1	28 March 2013	KR	20130032155	A	01 April 2013
US	2016011971	A1	14 January 2016	KR	20160008365	A	22 January 2016