(12) 按照专利合作条约所公布的国际申请

(19) 世界知识产权组织
国 际 局

(43) 国际公布日
2017 年 9 月 21 日 (21.09.2017)

WIPO | PCT

(10) 国际公布号
WO 2017/156683 A1

(51) 国际专利分类号:
G06F 12/0871 (2016.01)

(21) 国际申请号: PCT/CN2016/076296

(22) 国际申请日: 2016 年 3 月 14 日 (14.03.2016)

(25) 申请语言: 中文

(26) 公布语言: 中文

(71) 申请人: 深圳创维-RGB 电子有限公司 (SHENZHEN SKYWORTH-RGB ELECTRONIC CO., LTD.) [CN/CN]; 中国广东省深圳市南山区深南大道创维大厦 A 座 13-16 楼, Guangdong 518057 (CN)。

(72) 发明人: 何锐 (HE, Rui); 中国广东省深圳市南山区深南大道创维大厦 A 座 13-16 楼, Guangdong 518057 (CN)。

(74) 代理人: 深圳市世纪恒程知识产权代理事务所 (CENFO INTELLECTUAL PROPERTY AGENCY); 中国广东省深圳市南山区南山大道 3838 号设计产业园金栋二层 210-212（原南头城工业村 11 栋）, Guangdong 518052 (CN)。

(81) 指定国 (除另有指明，要求每一种可提供的国家保护): 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, DK, DM, DO, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, GT, HN, HR, HU, ID, IL, IN, IR, IS, JP, KE, KG, KN, KP, KR, 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) 指定国 (除另有指明，要求每一种可提供的地区保护): ARIPO (BW, GH, GM, KE, LR, LS, MW, MZ, NA, RW, SD, SL, ST, SZ, TZ, UG, ZM, ZW), 欧亚 (AM, AZ, BY, KG, KZ, RU, TJ, TM), 欧洲 (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)。

本国际公布:
— 包括国际检索报告(条约第 21 条(3))。

(54) Title: LINKED LIST-BASED APPLICATION CACHE MANAGEMENT METHOD AND DEVICE

(54) 发明名称 : 基于链表的应用缓存管理方法及装置

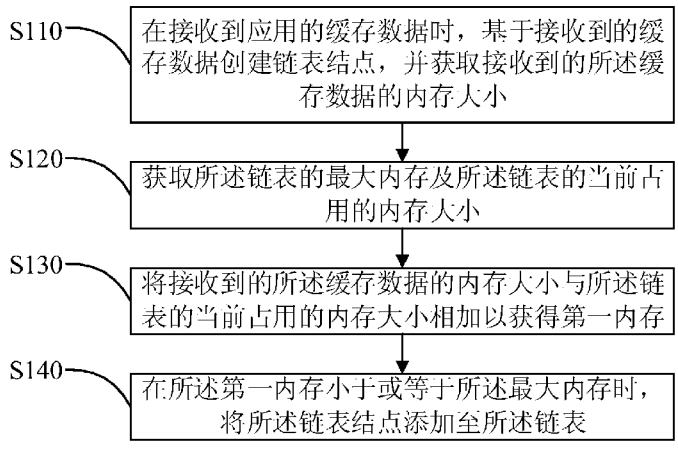

图 1

S110 When receiving cache data of an application, create a linked list node on the basis of the received cache data, and obtain the memory size of the received cache data

S120 Obtain the maximum memory of the linked list and the size of the currently occupied memory of the linked list

S130 Add the memory size of the received cache data and the size of the currently occupied memory of the linked list to obtain a first memory

S140 When the first memory is smaller than or equal to the maximum memory, add the linked list node into the linked list

(57) Abstract: The present invention discloses a linked list-based application cache management method, comprising: when the cache data of an application are received, a linked list node is created on the basis of the received cache data, and the memory size of the received cache data is obtained; the maximum memory of the linked list and the size of the currently occupied memory of the linked list are obtained; the memory size of the received cache data and the size of the currently occupied memory of the liked list are added to obtain a first memory; and when the first memory is smaller than or equal to the maximum memory, the linked list node is added into the linked list. The present invention further discloses a linked list-based application cache management device. The present invention implements that the received cache data is added into the linked list when the first memory is smaller than or equal to the maximum memory, thereby preventing the cache data in the linked list during a new cache data storage process from exceeding the maximum memory of the linked list, and improving the storage speed and efficiency of the cache data.

(57) 摘要:

[见续页]

WO 2017/156683 A1

本发明公开了一种基于链表的应用缓存管理方法，包括：在接收到应用的缓存数据时，基于接收到的缓存数据创建链表结点，并获取接收到的所述缓存数据的内存大小；获取所述链表的最大内存及所述链表的当前占用的内存大小；将接收到的所述缓存数据的内存大小与所述链表的当前占用的内存大小相加以获得第一内存；在所述第一内存小于或等于所述最大内存时，将所述链表结点添加至所述链表。本发明还公开了一种基于链表的应用缓存管理装置。本发明实现了在第一内存小于或等于所述最大内存时将接收到的缓存数据添加至链表中，避免了在新的缓存数据存储过程中链表中缓存数据超过链表的最大内存，提高了缓存数据的存储速度与效率。

发明名称: 基于链表的应用缓存管理方法及装置

[1] 技术领域

[2] 本发明涉及存储技术领域，尤其涉及一种基于链表的应用缓存管理方法及装置。

[3] 背景技术

[4] 随着移动通讯技术的快速发展，移动终端的应用也越来越丰富。在采用iOS开发应用的过程中，会用到缓存实现各种数据的存储以达到数据高速存取的目的。

[5] 目前，采用NSCache（NSCache是iOS引入的一个方便的缓存某些object的类）内存缓存机制的iOS平台，没有严格限制缓存数据占用的内存大小。iOS平台只能在存储新的缓存数据后，若当前存储的缓存数据总的内存超过内存限制，则删除其中最终存储的缓存数据。采用这种存储方式在新的缓存数据存储过程中，缓存数据的总容量可能超出系统的内存限制，导致缓存数据的存储速度及效率低。

[6] 发明内容

[7] 本发明提供一种基于链表的应用缓存管理方法及装置，旨在解决采用NSCache的iOS平台在新的缓存数据存储过程中，由于缓存数据的总容量可能超出系统的内存限制而导致 缓存数据的存储速度及效率低的技术问题。

[8] 为实现上述目的，本发明提供的一种基于链表的应用缓存管理方法，所述基于链表的应用缓存管理方法包括以下步骤:

[9] 在接收到应用的缓存数据时，基于接收到的缓存数据创建链表结点，并获取接收到的所述缓存数据的内存大小，

[10] 获取所述链表的最大内存及所述链表的当前占用的内存大小；

[11] 将接收到的所述缓存数据的内存大小与所述链表的当前占用的内存大小相加以获得第一内存；

[12] 在所述第一内存小于或等于所述最大内存时，将所述链表结点添加至所述链表

。

[13] 优选地，将接收到的所述缓存数据的内存大小与所述链表的当前占用的内存大小相加以获得第一内存的步骤之后，所述基于链表的应用缓存管理方法包括：

[14] 在所述第一内存大于所述最大内存时，获取所述链表中各个结点的上一次访问时间与当前时间的时间间隔；

[15] 删除所述链表中时间间隔最大的结点；

[16] 将接收到的所述缓存数据的内存大小与结点删除后所述链表的当前占用的内存大小相加以获得第二内存；

[17] 在所述第二内存小于或等于所述最大内存时，将所述链表结点添加至所述链表。

[18] 优选地，在所述第一内存小于或等于所述最大内存时，将所述链表结点添加至所述链表的步骤之后，所述基于链表的应用缓存管理方法还包括：

[19] 在接收到缓存数据的访问请求时，获取所述访问请求对应的缓存数据的标识信息；

[20] 基于所述标识信息遍历所述链表以访问所述访问请求对应的缓存数据。

[21] 优选地，基于所述标识信息遍历所述链表以访问所述访问请求对应的缓存数据的步骤之后，所述基于链表的应用缓存管理方法还包括：

[22] 将遍历到的缓存数据对应的结点设置为所述链表的头结点。

[23] 优选地，所述链表结点的数据域包含所述链表结点的存活时间，在所述第一内存小于或等于所述最大内存时，将所述链表结点添加至所述链表的步骤之后，所述基于链表的应用缓存管理方法还包括：

[24] 基于所述链表结点的创建时间确定所述链表结点当前是否达到所述存活时间；

[25] 在所述链表结点当前达到所述存活时间时，删除所述链表结点。

[26] 此外，为实现上述目的，本发明还提供一种基于链表的应用缓存管理装置，所述基于链表的应用缓存管理装置包括：

[27] 创建模块，用于在接收到应用的缓存数据时，基于接收到的缓存数据创建链表结点，并获取接收到的所述缓存数据的内存大小，

[28] 第一获取模块，用于获取所述链表的最大内存及所述链表的当前占用的内存大

小；

[29] 第一计算模块，用于将接收到的所述缓存数据的内存大小与所述链表的当前占用的内存大小相加以获得第一内存；

[30] 第一添加模块，用于在所述第一内存小于或等于所述最大内存时，将所述链表结点添加至所述链表。

[31] 优选地，所述基于链表的应用缓存管理装置还包括：

[32] 第二获取模块，用于在所述第一内存大于所述最大内存时，获取所述链表中各个结点的上一次访问时间与当前时间的时间间隔；

[33] 第一删除模块，用于删除所述链表中时间间隔最大的结点；

[34] 第二计算模块，用于将接收到的所述缓存数据的内存大小与结点删除后所述链表的当前占用的内存大小相加以获得第二内存；

[35] 第二添加模块，用于在所述第二内存小于或等于所述最大内存时，将所述链表结点添加至所述链表。

[36] 优选地，所述基于链表的应用缓存管理装置还包括：

[37] 第三获取模块，用于在接收到缓存数据的访问请求时，获取所述访问请求对应的缓存数据的标识信息；

[38] 遍历模块，用于基于所述标识信息遍历所述链表以访问所述访问请求对应的缓存数据。

[39] 优选地，所述基于链表的应用缓存管理装置还包括：

[40] 设置模块，用于将遍历到的缓存数据对应的结点设置为所述链表的头结点。

[41] 优选地，所述链表结点的数据域包含所述链表结点的存活时间，所述基于链表的应用缓存管理装置还包括：

[42] 确定模块，用于基于所述链表结点的创建时间确定所述链表结点当前是否达到所述存活时间；

[43] 第二删除模块，用于在所述链表结点当前达到所述存活时间时，删除所述链表结点。

[44] 本发明通过在接收到应用的缓存数据时，基于接收到的缓存数据创建链表结点，并获取接收到的所述缓存数据的内存大小；接着获取所述链表的最大内存及

所述链表的当前占用的内存大小；然后将接收到的所述缓存数据的内存大小与所述链表的当前占用的内存大小相加以获得第一内存；最后在所述第一内存小于或等于所述最大内存时，将所述链表结点添加至所述链表，实现了在第一内存小于或等于所述最大内存时将接收到的缓存数据添加至链表中，避免了在新的缓存数据存储过程中链表中缓存数据超过链表的最大内存，提高了缓存数据的存储速度与效率。

[45] 附图说明

[46] 图1为本发明基于链表的应用缓存管理方法第一实施例的流程示意图；

[47] 图2为本发明基于链表的应用缓存管理方法第二实施例的流程示意图；

[48] 图3为本发明基于链表的应用缓存管理方法第三实施例的流程示意图；

[49] 图4为本发明基于链表的应用缓存管理方法第四实施例的流程示意图；

[50] 图5为本发明基于链表的应用缓存管理装置第一实施例的功能模块示意图；

[51] 图6为本发明基于链表的应用缓存管理装置第二实施例的功能模块示意图；

[52] 图7为本发明基于链表的应用缓存管理装置第三实施例的功能模块示意图；

[53] 图8为本发明基于链表的应用缓存管理装置第四实施例的功能模块示意图。

[54] 本发明目的的实现、功能特点及优点将结合实施例，参照附图做进一步说明。

[55] 具体实施方式

[56] 应当理解，此处所描述的具体实施例仅仅用以解释本发明，并不用于限定本发明。

[57] 本发明提供一种基于链表的应用缓存管理方法。参照图1，图1为本发明基于链表的应用缓存管理方法第一实施例的流程示意图。

[58] 在本实施例中，该基于链表的应用缓存管理方法包括：

[59] 步骤S110，在接收到应用的缓存数据时，基于接收到的缓存数据创建链表结点，并获取接收到的所述缓存数据的内存大小；

[60] 本实施例中，链表可以为双向链表或单链表，创建链表结点是指将缓存数据存储至新建的链表结点的数据域，其中，新建链表结点的直接后继为当前链表的头结点。同时，获取接收到的缓存数据的内存大小，譬如接收到的缓存数据的内存为50M等。其中，本实施例基于链表的应用缓存管理方法可以应用与iOS平

台，譬如开发移动终端各种应用的iOS平台。

[61] 步骤S120，获取所述链表的最大内存及所述链表的当前占用的内存大小；

[62] 其中，所述链表的最大内存是指在建立链表时设置的链表的最大容量，所述链表的当前占用的内存大小是指链表当前存储的所有缓存数据的内存大小。

[63] 步骤S130，将接收到的所述缓存数据的内存大小与所述链表的当前占用的内存大小相加以获得第一内存；

[64] 步骤S140，在所述第一内存小于或等于所述最大内存时，将所述链表结点添加至所述链表。

[65] 在所述第一内存小于或等于所述最大内存时，表示链表当前存储的缓存数据未超过其最大容量，将所述链表结点添加至所述链表，本实施例中，具体是指将链表结点作为所述链表的头结点。

[66] 本实施例中，通过在接收到应用的缓存数据时，基于接收到的缓存数据创建链表结点，并获取接收到的所述缓存数据的内存大小；接着获取所述链表的最大内存及所述链表的当前占用的内存大小；然后将接收到的所述缓存数据的内存大小与所述链表的当前占用的内存大小相加以获得第一内存；最后在所述第一内存小于或等于所述最大内存时，将所述链表结点添加至所述链表，实现了在第一内存小于或等于所述最大内存时将接收到的缓存数据添加至链表中，避免了在新的缓存数据存储过程中链表中缓存数据超过链表的最大内存，提高了缓存数据的存储速度与效率。

[67] 基于第一实施例提出本发明基于链表的应用缓存管理方法的第二实施例，参照图2，在本实施例中，在步骤S130之后，该基于链表的应用缓存管理方法还包括：

[68] 步骤S150，在所述第一内存大于所述最大内存时，获取所述链表中各个结点的上一次访问时间与当前时间的时间间隔；

[69] 其中，链表中各个结点的上一次访问时间是指链表中各个结点最后一次被访问的访问时间。

[70] 步骤S160，删除所述链表中时间间隔最大的结点；

[71] 删除所述链表中时间间隔最大的结点是指删除链表中最后一次被访问的访问时

间距当前时间最久的结点，即删除链表中最久没有被访问的结点。

[72] 步骤S170，将接收到的所述缓存数据的内存大小与结点删除后所述链表的当前占用的内存大小相加以获得第二内存；

[73] 步骤S180，在所述第二内存小于或等于所述最大内存时，将所述链表结点添加至所述链表。

[74] 链表中时间间隔最大的结点被删除以后，在所述第二内存小于或等于所述最大内存时，即链表的当前占用的内存大小与接收到的所述缓存数据的内存大小之和小于最大内存时，将所述链表结点添加至所述链表，本实施例中，具体是指将链表结点作为所述链表的头结点。

[75] 本实施例中，通过在所述第一内存大于所述最大内存时，获取所述链表中各个结点的上一次访问时间与当前时间的时间间隔；接着删除所述链表中时间间隔最大的结点；然后将接收到的所述缓存数据的内存大小与结点删除后所述链表的当前占用的内存大小相加以获得第二内存；最后在所述第二内存小于或等于所述最大内存时，将所述链表结点添加至所述链表，实现了在第一内存大于所述最大内存时，首先删除链表中时间间隔最大的结点，以保证第二内存小于或等于所述最大内存，最后将所述链表结点添加至所述链表，进一步避免了在新的缓存数据存储过程中链表中缓存数据超过链表的最大内存，提高了缓存数据的存储速度与效率。

[76] 基于第一实施例提出本发明基于链表的应用缓存管理方法的第三实施例，参照图3，在本实施例中，在步骤S40之后，该基于链表的应用缓存管理方法还包括：

[77] 步骤S190，在接收到缓存数据的访问请求时，获取所述访问请求对应的缓存数据的标识信息；

[78] 其中，在采用iOS平台开发移动终端各种应用的过程中，需要调用链表内的缓存数据时，生成缓存数据的访问请求，该访问请求携带有缓存数据的标识信息，在接收到缓存数据的访问请求时，解析所述访问请求以获取所述访问请求对应的缓存数据的标识信息。

[79] 步骤S200，基于所述标识信息遍历所述链表以访问所述访问请求对应的缓存数

据。

[80] 在获得缓存数据的标识信息之后，基于所述标识信息遍历所述链表，以访问所述访问请求对应的缓存数据。现有采用NSCache的iOS平台在缓存数据访问时，需要将标识信息与缓存数据的key进行一一匹配，在各个缓存数据key相似时，即具有大量相似的key时，系统会消耗大量的时间在key的匹配对比上，导致缓存数据的读取性能较低即缓存数据的读取速度及效率较低。

[81] 在其他实施例中，在步骤S200之后，该基于链表的应用缓存管理方法还包括：将遍历到的缓存数据对应的结点设置为所述链表的头结点。

[82] 通过将遍历到的缓存数据对应的结点设置为所述链表的头结点，使得在后续接收到新的缓存数据时，若所述第一内存大于所述最大内存，则删除所述链表的尾结点；将接收到的所述缓存数据的内存大小与结点删除后所述链表的当前占用的内存大小相加以获得第二内存；在所述第二内存小于或等于所述最大内存时，将所述链表结点添加至所述链表。减少了第一内存大于所述最大内存时存储缓存数据的流程，提高了缓存数据的存储速度及效率。

[83] 本实施例通过在接收到缓存数据的访问请求时，获取所述访问请求对应的缓存数据的标识信息；接着基于所述标识信息遍历所述链表以访问所述访问请求对应的缓存数据，通过根据缓存数据的标识信息遍历所述链表实现了缓存数据的访问，提高了缓存数据读取的速度及效率。

[84] 基于第一实施例提出本发明基于链表的应用缓存管理方法的第四实施例，参照图4，在本实施例中，所述链表结点的数据域包含所述链表结点的存活时间，在步骤S40之后，该基于链表的应用缓存管理方法还包括：

[85] 步骤S210，基于所述链表结点的创建时间确定所述链表结点当前是否达到所述存活时间；

[86] 本实施例中，在存储缓存数据时，创建的链表结点的数据域包含所述链表结点的存活时间，在链表结点添加至链表后开始计时，以确定所述链表结点当前是否达到所述存活时间。

[87] 步骤S220，在所述链表结点当前达到所述存活时间时，删除所述链表结点。

[88] 本实施例通过基于所述链表结点的创建时间确定所述链表结点当前是否达到所

述存活时间，接着在所述链表结点当前达到所述存活时间时，删除所述链表结点，实现了根据链表结点的存活时间删除链表结点，即实现了缓存数据的定时存储，进一步提高了缓存数据的存取效率。

[89] 本发明进一步提供一种基于链表的应用缓存管理装置。参照图5，图5为本发明基于链表的应用缓存管理装置第一实施例的功能模块示意图。

[90] 在本实施例中，该基于链表的应用缓存管理装置包括：

[91] 创建模块110，用于在接收到应用的缓存数据时，基于接收到的缓存数据创建链表结点，并获取接收到的所述缓存数据的内存大小，

[92] 本实施例中，链表可以为双向链表或单链表，创建链表结点是指将缓存数据存储至新建的链表结点的数据域，其中，新建链表结点的直接后继为当前链表的头结点。同时，获取接收到的缓存数据的内存大小，譬如接收到的缓存数据的内存为50M等。其中，本实施例基于链表的应用缓存管理方法可以应用与iOS平台，譬如开发移动终端各种应用的iOS平台。

[93] 第一获取模块120，用于获取所述链表的最大内存及所述链表的当前占用的内存大小；

[94] 其中，所述链表的最大内存是指在建立链表时设置的链表的最大容量，所述链表的当前占用的内存大小是指链表当前存储的所有缓存数据的内存大小。

[95] 第一计算模块130，用于将接收到的所述缓存数据的内存大小与所述链表的当前占用的内存大小相加以获得第一内存；

[96] 第一添加模块140，用于在所述第一内存小于或等于所述最大内存时，将所述链表结点添加至所述链表。

[97] 在所述第一内存小于或等于所述最大内存时，表示链表当前存储的缓存数据未超过其最大容量，第一添加模块140将所述链表结点添加至所述链表，本实施例中，具体是指将链表结点作为所述链表的头结点。

[98] 本实施例中，通过在接收到应用的缓存数据时，创建模块110基于接收到的缓存数据创建链表结点，并获取接收到的所述缓存数据的内存大小；第一获取模块120接着获取所述链表的最大内存及所述链表的当前占用的内存大小；第一计算模块130然后将接收到的所述缓存数据的内存大小与所述链表的当前占用的内

存大小相加以获得第一内存；最后在所述第一内存小于或等于所述最大内存时，第一添加模块140将所述链表结点添加至所述链表，实现了在第一内存小于或等于所述最大内存时将接收到的缓存数据添加至链表中，避免了在新的缓存数据存储过程中链表中缓存数据超过链表的最大内存，提高了缓存数据的存储速度与效率。

[99] 基于第一实施例提出本发明基于链表的应用缓存管理装置的第二实施例，参照图6，在本实施例中，该基于链表的应用缓存管理装置还包括：

[100] 第二获取模块150，用于在所述第一内存大于所述最大内存时，获取所述链表中各个结点的上一次访问时间与当前时间的时间间隔；

[101] 其中，链表中各个结点的上一次访问时间是指链表中各个结点最后一次被访问的访问时间。

[102] 第一删除模块160，用于删除所述链表中时间间隔最大的结点；

[103] 第一删除模块160删除所述链表中时间间隔最大的结点是指删除链表中最后一次被访问的访问时间距当前时间最久的结点，即删除链表中最久没有被访问的结点。

[104] 第二计算模块170，用于将接收到的所述缓存数据的内存大小与结点删除后所述链表的当前占用的内存大小相加以获得第二内存；

[105] 第二添加模块180，用于在所述第二内存小于或等于所述最大内存时，将所述链表结点添加至所述链表。

[106] 链表中时间间隔最大的结点被删除以后，在所述第二内存小于或等于所述最大内存时，即链表的当前占用的内存大小与接收到的所述缓存数据的内存大小之和小于最大内存时，第二添加模块180将所述链表结点添加至所述链表，本实施例中，具体是指将链表结点作为所述链表的头结点。

[107] 本实施例中，通过在所述第一内存大于所述最大内存时，第二获取模块150获取所述链表中各个结点的上一次访问时间与当前时间的时间间隔；接着第一删除模块160删除所述链表中时间间隔最大的结点；然后第二计算模块170将接收到的所述缓存数据的内存大小与结点删除后所述链表的当前占用的内存大小相加以获得第二内存；最后在所述第二内存小于或等于所述最大内存时，第二添

加模块180将所述链表结点添加至所述链表，实现了在第一内存大于所述最大内存时，首先删除链表中时间间隔最大的结点，以保证第二内存小于或等于所述最大内存，最后将所述链表结点添加至所述链表，进一步避免了在新的缓存数据存储过程中链表中缓存数据超过链表的最大内存，提高了缓存数据的存储速度与效率。

[108] 基于第一实施例提出本发明基于链表的应用缓存管理装置的第三实施例，参照图7，在本实施例中，该基于链表的应用缓存管理装置还包括：

[109] 第三获取模块190，用于在接收到缓存数据的访问请求时，获取所述访问请求对应的缓存数据的标识信息；

[110] 其中，在采用iOS平台开发移动终端各种应用的过程中，需要调用链表内的缓存数据时，生成缓存数据的访问请求，该访问请求携带有缓存数据的标识信息，在接收到缓存数据的访问请求时，第三获取模块190解析所述访问请求以获取所述访问请求对应的缓存数据的标识信息。

[111] 遍历模块200，用于基于所述标识信息遍历所述链表以访问所述访问请求对应的缓存数据。

[112] 在获得缓存数据的标识信息之后，基于所述标识信息遍历所述链表，以访问所述访问请求对应的缓存数据。现有采用NSCache的iOS平台在缓存数据访问时，需要将标识信息与缓存数据的key进行一一匹配，在各个缓存数据key相似时，即具有大量相似的key时，系统会消耗大量的时间在key的匹配对比上，导致缓存数据的读取性能较低即缓存数据的读取速度及效率较低。

[113] 在其他实施例中，该基于链表的应用缓存管理装置还包括：设置模块，用于将遍历到的缓存数据对应的结点设置为所述链表的头结点。

[114] 通过设置模块将遍历到的缓存数据对应的结点设置为所述链表的头结点，使得在后续接收到新的缓存数据时，若所述第一内存大于所述最大内存，则删除所述链表的尾结点；将接收到的所述缓存数据的内存大小与结点删除后所述链表的当前占用的内存大小相加以获得第二内存；在所述第二内存小于或等于所述最大内存时，将所述链表结点添加至所述链表。减少了第一内存大于所述最大内存时存储缓存数据的流程，提高了缓存数据的存储速度及效率。

[115] 本实施例通过在接收到缓存数据的访问请求时，第三获取模块190获取所述访问请求对应的缓存数据的标识信息；接着遍历模块200基于所述标识信息遍历所述链表以访问所述访问请求对应的缓存数据，通过根据缓存数据的标识信息遍历所述链表实现了缓存数据的访问，提高了缓存数据读取的速度及效率。

[116] 基于第一实施例提出本发明基于链表的应用缓存管理装置的第四实施例，参照图8，在本实施例中，所述链表结点的数据域包含所述链表结点的存活时间，该基于链表的应用缓存管理装置还包括：

[117] 确定模块210，用于基于所述链表结点的创建时间确定所述链表结点当前是否达到所述存活时间；

[118] 本实施例中，在存储缓存数据时，创建的链表结点的数据域包含所述链表结点的存活时间，在链表结点添加至链表后开始计时，以确定所述链表结点当前是否达到所述存活时间。

[119] 第二删除模块220，用于在所述链表结点当前达到所述存活时间时，删除所述链表结点。

[120] 本实施例通过确定模块210基于所述链表结点的创建时间确定所述链表结点当前是否达到所述存活时间，接着在所述链表结点当前达到所述存活时间时，第二删除模块220删除所述链表结点，实现了根据链表结点的存活时间删除链表结点，即实现了缓存数据的定时存储，进一步提高了缓存数据的存取效率。

[121] 需要说明的是，在本文中，术语“包括”、“包含”或者其任何其它变体意在涵盖非排他性的包含，从而使得包括一系列要素的过程、方法、物品或者装置不仅包括那些要素，而且还包括没有明确列出的其它要素，或者是还包括为这种过程、方法、物品或者装置所固有的要素。在没有更多限制的情况下，由语句“包括一个……”限定的要素，并不排除在包括该要素的过程、方法、物品或者装置中还存在另外的相同要素。

[122] 上述本发明实施例序号仅仅为了描述，不代表实施例的优劣。

[123] 通过以上的实施方式的描述，本领域的技术人员可以清楚地了解到上述实施例方法可借助软件加必需的通用硬件平台的方式来实现，当然也可以通过硬件，但很多情况下前者是更佳的实施方式。基于这样的理解，本发明的技术方案本

质上或者说对现有技术做出贡献的部分可以以软件产品的形式体现出来，该计算机软件产品存储在一个存储介质（如ROM/RAM、磁碟、光盘）中，包括若干指令用以使得一台终端设备（可以是手机，计算机，服务器，空调器，或者网络设备等）执行本发明各个实施例所述的方法。

[124] 以上仅为本发明的优选实施例，并非因此限制本发明的专利范围，凡是利用本发明说明书及附图内容所作的等效结构或等效流程变换，或直接或间接运用在其他相关的技术领域，均同理包括在本发明的专利保护范围内。

权利要求书

[权利要求 1] 一种基于链表的应用缓存管理方法，其特征在于，所述基于链表的应用缓存管理方法包括以下步骤：

在接收到应用的缓存数据时，基于接收到的缓存数据创建链表结点，并获取接收到的所述缓存数据的内存大小；

获取所述链表的最大内存及所述链表的当前占用的内存大小；

将接收到的所述缓存数据的内存大小与所述链表的当前占用的内存大小相加以获得第一内存；

在所述第一内存小于或等于所述最大内存时，将所述链表结点添加至所述链表。

[权利要求 2] 如权利要求1所述的基于链表的应用缓存管理方法，其特征在于，将接收到的所述缓存数据的内存大小与所述链表的当前占用的内存大小相加以获得第一内存的步骤之后，所述基于链表的应用缓存管理方法包括：

在所述第一内存大于所述最大内存时，获取所述链表中各个结点的上一次访问时间与当前时间的时间间隔；

删除所述链表中时间间隔最大的结点；

将接收到的所述缓存数据的内存大小与结点删除后所述链表的当前占用的内存大小相加以获得第二内存；

在所述第二内存小于或等于所述最大内存时，将所述链表结点添加至所述链表。

[权利要求 3] 如权利要求1所述的基于链表的应用缓存管理方法，其特征在于，在所述第一内存小于或等于所述最大内存时，将所述链表结点添加至所述链表的步骤之后，所述基于链表的应用缓存管理方法还包括：

在接收到缓存数据的访问请求时，获取所述访问请求对应的缓存数据的标识信息；

基于所述标识信息遍历所述链表以访问所述访问请求对应的缓存

数据。

[权利要求 4] 如权利要求3所述的基于链表的应用缓存管理方法，其特征在于，基于所述标识信息遍历所述链表以访问所述访问请求对应的缓存数据的步骤之后，所述基于链表的应用缓存管理方法还包括：

将遍历到的缓存数据对应的结点设置为所述链表的头结点。

[权利要求 5] 如权利要求1所述的基于链表的应用缓存管理方法，其特征在于，所述链表结点的数据域包含所述链表结点的存活时间，在所述第一内存小于或等于所述最大内存时，将所述链表结点添加至所述链表的步骤之后，所述基于链表的应用缓存管理方法还包括：

基于所述链表结点的创建时间确定所述链表结点当前是否达到所述存活时间；

在所述链表结点当前达到所述存活时间时，删除所述链表结点。

[权利要求 6] 如权利要求2所述的基于链表的应用缓存管理方法，其特征在于，所述链表结点的数据域包含所述链表结点的存活时间，在所述第一内存小于或等于所述最大内存时，将所述链表结点添加至所述链表的步骤之后，所述基于链表的应用缓存管理方法还包括：

基于所述链表结点的创建时间确定所述链表结点当前是否达到所述存活时间；

在所述链表结点当前达到所述存活时间时，删除所述链表结点。

[权利要求 7] 如权利要求3所述的基于链表的应用缓存管理方法，其特征在于，所述链表结点的数据域包含所述链表结点的存活时间，在所述第一内存小于或等于所述最大内存时，将所述链表结点添加至所述链表的步骤之后，所述基于链表的应用缓存管理方法还包括：

基于所述链表结点的创建时间确定所述链表结点当前是否达到所述存活时间；

在所述链表结点当前达到所述存活时间时，删除所述链表结点。

[权利要求 8] 如权利要求4所述的基于链表的应用缓存管理方法，其特征在于，所述链表结点的数据域包含所述链表结点的存活时间，在所述第

一内存小于或等于所述最大内存时，将所述链表结点添加至所述链表的步骤之后，所述基于链表的应用缓存管理方法还包括：

基于所述链表结点的创建时间确定所述链表结点当前是否达到所述存活时间；

在所述链表结点当前达到所述存活时间时，删除所述链表结点。

[权利要求 9] 一种基于链表的应用缓存管理装置，其特征在于，所述基于链表的应用缓存管理装置包括：

创建模块，用于在接收到应用的缓存数据时，基于接收到的缓存数据创建链表结点，并获取接收到的所述缓存数据的内存大小，

第一获取模块，用于获取所述链表的最大内存及所述链表的当前占用的内存大小；

第一计算模块，用于将接收到的所述缓存数据的内存大小与所述链表的当前占用的内存大小相加以获得第一内存；

第一添加模块，用于在所述第一内存小于或等于所述最大内存时，将所述链表结点添加至所述链表。

[权利要求 10] 如权利要求9所述的基于链表的应用缓存管理装置，其特征在于，所述基于链表的应用缓存管理装置还包括：

第二获取模块，用于在所述第一内存大于所述最大内存时，获取所述链表中各个结点的上一次访问时间与当前时间的时间间隔；

第一删除模块，用于删除所述链表中时间间隔最大的结点；

第二计算模块，用于将接收到的所述缓存数据的内存大小与结点删除后所述链表的当前占用的内存大小相加以获得第二内存；

第二添加模块，用于在所述第二内存小于或等于所述最大内存时，将所述链表结点添加至所述链表。

[权利要求 11] 如权利要求9所述的基于链表的应用缓存管理装置，其特征在于，所述基于链表的应用缓存管理装置还包括：

第三获取模块，用于在接收到缓存数据的访问请求时，获取所述访问请求对应的缓存数据的标识信息；

遍历模块，用于基于所述标识信息遍历所述链表以访问所述访问请求对应的缓存数据。

[权利要求 12] 如权利要求11所述的基于链表的应用缓存管理装置，其特征在于，所述基于链表的应用缓存管理装置还包括：

设置模块，用于将遍历到的缓存数据对应的结点设置为所述链表的头结点。

[权利要求 13] 如权利要求9所述的基于链表的应用缓存管理装置，其特征在于，所述链表结点的数据域包含所述链表结点的存活时间，所述基于链表的应用缓存管理装置还包括：

确定模块，用于基于所述链表结点的创建时间确定所述链表结点当前是否达到所述存活时间；

第二删除模块，用于在所述链表结点当前达到所述存活时间时，删除所述链表结点。

[权利要求 14] 如权利要求10所述的基于链表的应用缓存管理装置，其特征在于，所述链表结点的数据域包含所述链表结点的存活时间，所述基于链表的应用缓存管理装置还包括：

确定模块，用于基于所述链表结点的创建时间确定所述链表结点当前是否达到所述存活时间；

第二删除模块，用于在所述链表结点当前达到所述存活时间时，删除所述链表结点。

[权利要求 15] 如权利要求11所述的基于链表的应用缓存管理装置，其特征在于，所述链表结点的数据域包含所述链表结点的存活时间，所述基于链表的应用缓存管理装置还包括：

确定模块，用于基于所述链表结点的创建时间确定所述链表结点当前是否达到所述存活时间；

第二删除模块，用于在所述链表结点当前达到所述存活时间时，删除所述链表结点。

[权利要求 16] 如权利要求12所述的基于链表的应用缓存管理装置，其特征在于

，所述链表结点的数据域包含所述链表结点的存活时间，所述基于链表的应用缓存管理装置还包括：

确定模块，用于基于所述链表结点的创建时间确定所述链表结点当前是否达到所述存活时间；

第二删除模块，用于在所述链表结点当前达到所述存活时间时，删除所述链表结点。

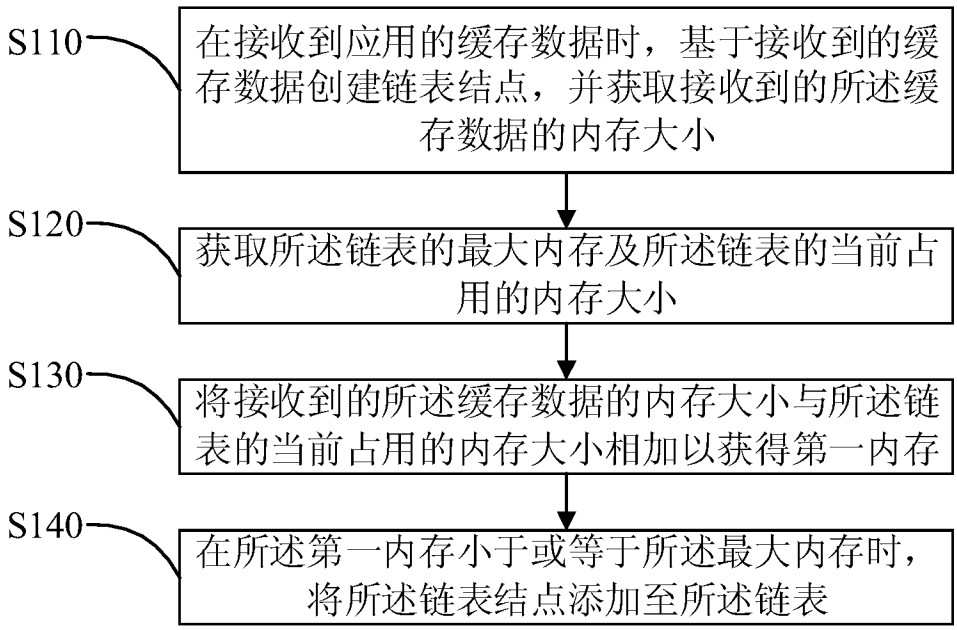
S110
在接收到应用的缓存数据时，基于接收到的缓存数据创建链表结点，并获取接收到的所述缓存数据的内存大小
S120
获取所述链表的最大内存及所述链表的当前占用的内存大小
S130
将接收到的所述缓存数据的内存大小与所述链表的当前占用的内存大小相加以获得第一内存
S140
在所述第一内存小于或等于所述最大内存时，将所述链表结点添加至所述链表

图 1

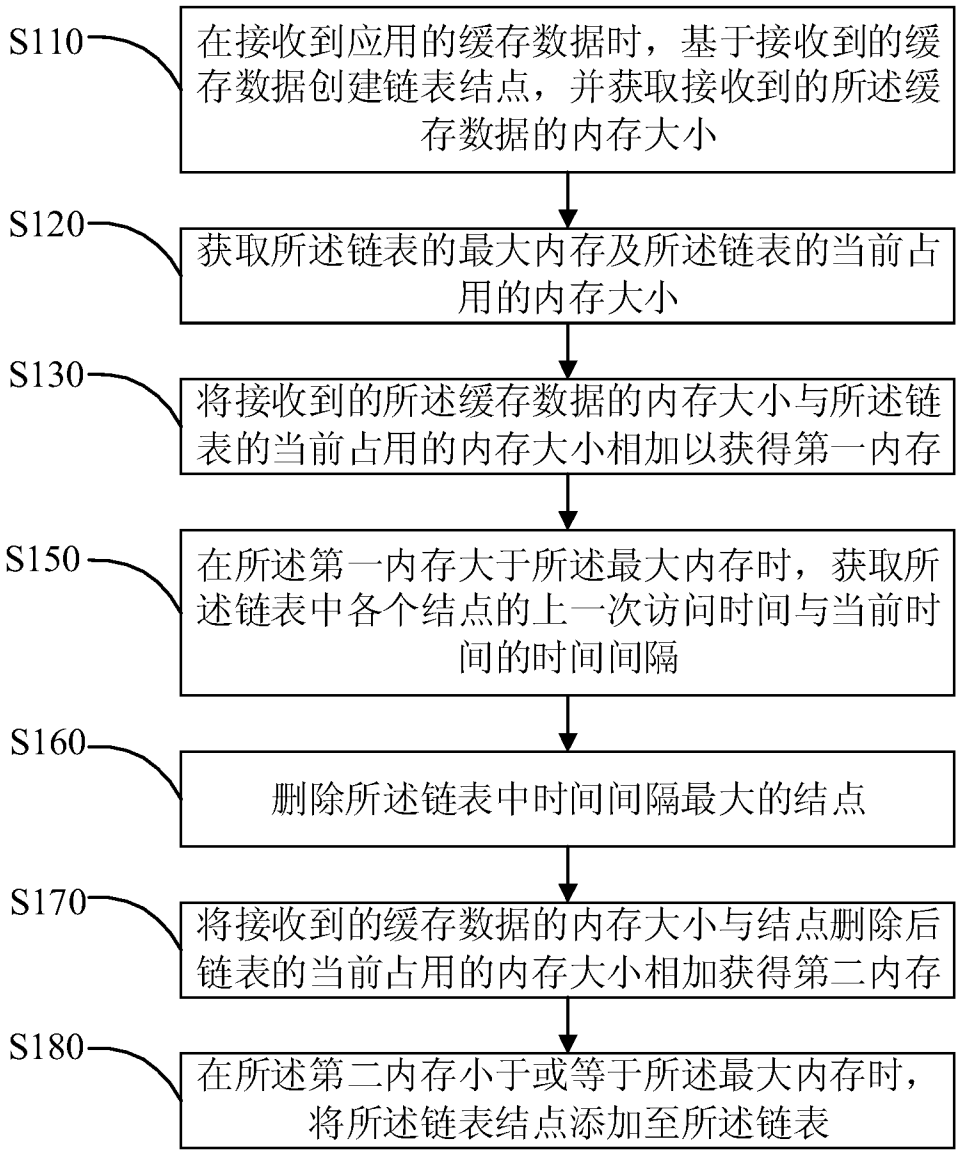
S110
在接收到应用的缓存数据时，基于接收到的缓存数据创建链表结点，并获取接收到的所述缓存数据的内存大小
S120
获取所述链表的最大内存及所述链表的当前占用的内存大小
S130
将接收到的所述缓存数据的内存大小与所述链表的当前占用的内存大小相加以获得第一内存
S150
在所述第一内存大于所述最大内存时，获取所述链表中各个结点的上一次访问时间与当前时间的时间间隔
S160
删除所述链表中时间间隔最大的结点
S170
将接收到的缓存数据的内存大小与结点删除后链表的当前占用的内存大小相加获得第二内存
S180
在所述第二内存小于或等于所述最大内存时，将所述链表结点添加至所述链表

图 2

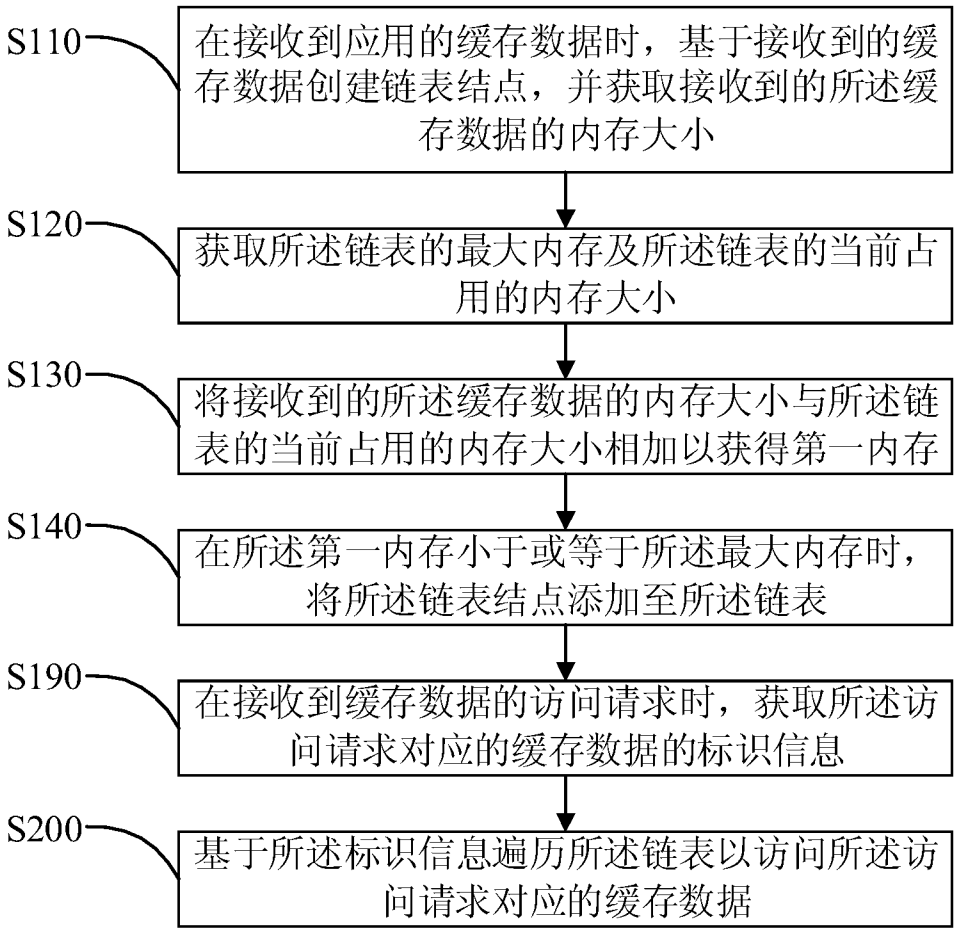
S110
在接收到应用的缓存数据时，基于接收到的缓存数据创建链表结点，并获取接收到的所述缓存数据的内存大小
S120
获取所述链表的最大内存及所述链表的当前占用的内存大小
S130
将接收到的所述缓存数据的内存大小与所述链表的当前占用的内存大小相加以获得第一内存
S140
在所述第一内存小于或等于所述最大内存时，将所述链表结点添加至所述链表
S190
在接收到缓存数据的访问请求时，获取所述访问请求对应的缓存数据的标识信息
S200
基于所述标识信息遍历所述链表以访问所述访问请求对应的缓存数据

图 3

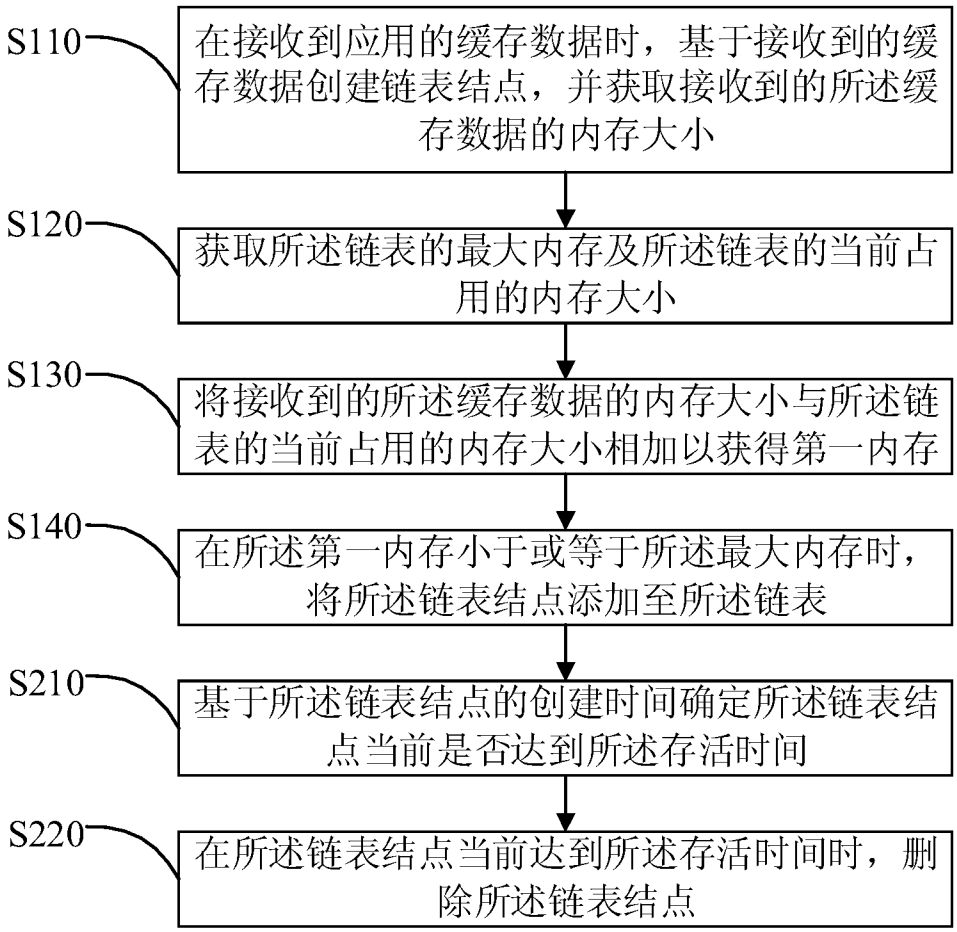
S110
在接收到应用的缓存数据时，基于接收到的缓存数据创建链表结点，并获取接收到的所述缓存数据的内存大小
S120
获取所述链表的最大内存及所述链表的当前占用的内存大小
S130
将接收到的所述缓存数据的内存大小与所述链表的当前占用的内存大小相加以获得第一内存
S140
在所述第一内存小于或等于所述最大内存时，将所述链表结点添加至所述链表
S210
基于所述链表结点的创建时间确定所述链表结点当前是否达到所述存活时间
S220
在所述链表结点当前达到所述存活时间时，删除所述链表结点

图 4

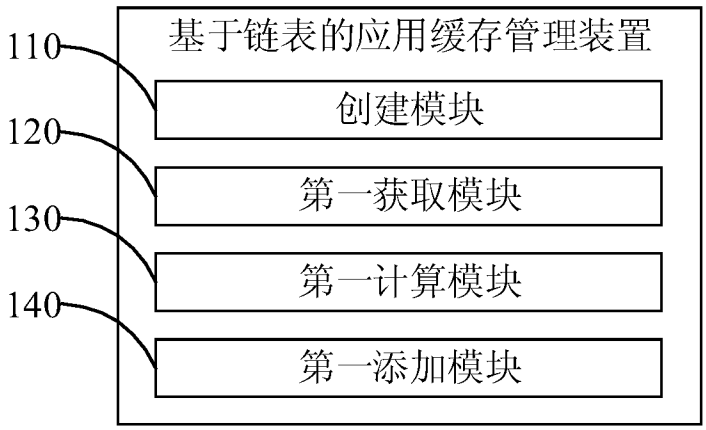
基于链表的应用缓存管理装置
110
创建模块
120
第一获取模块
130
第一计算模块
140
第一添加模块

图 5

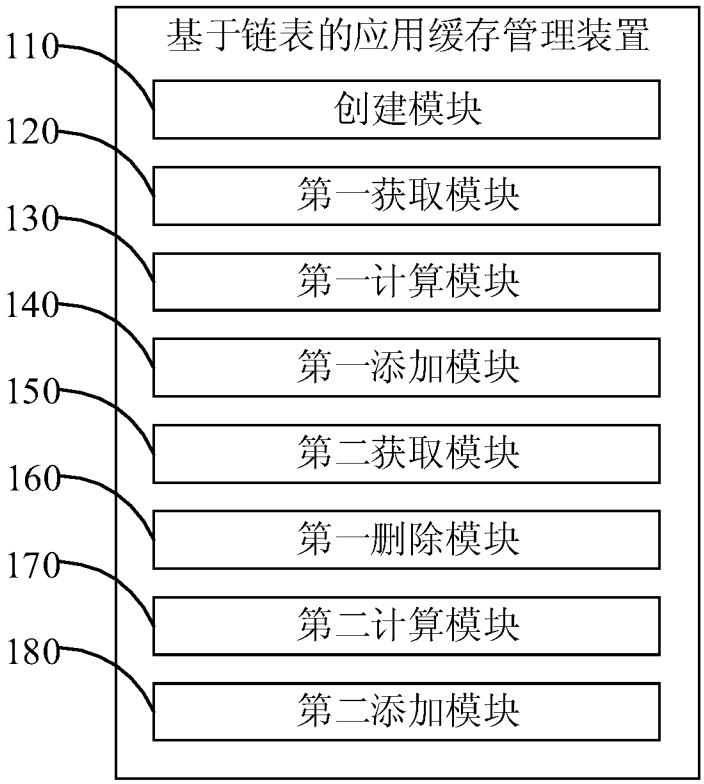
基于链表的应用缓存管理装置
110
创建模块
120
第一获取模块
130
第一计算模块
140
第一添加模块
150
第二获取模块
160
第一删除模块
170
第二计算模块
180
第二添加模块

图 6

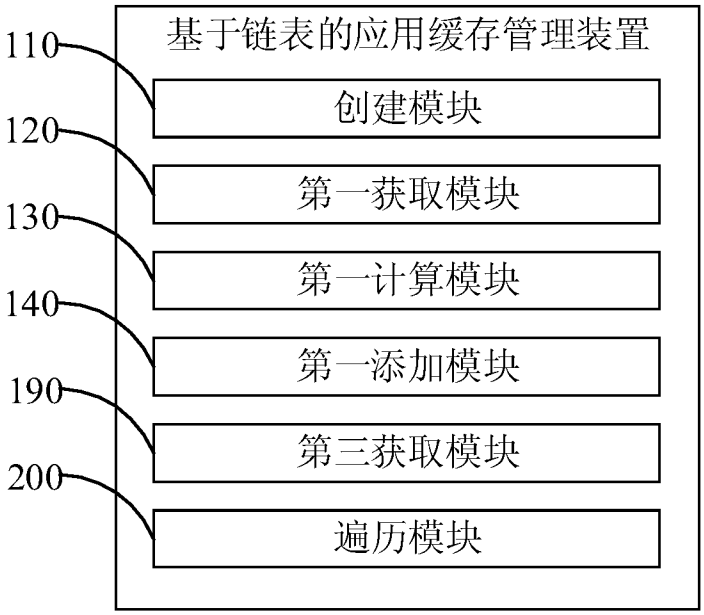
基于链表的应用缓存管理装置
110
创建模块
120
第一获取模块
130
第一计算模块
140
第一添加模块
190
第三获取模块
200
遍历模块

图 7

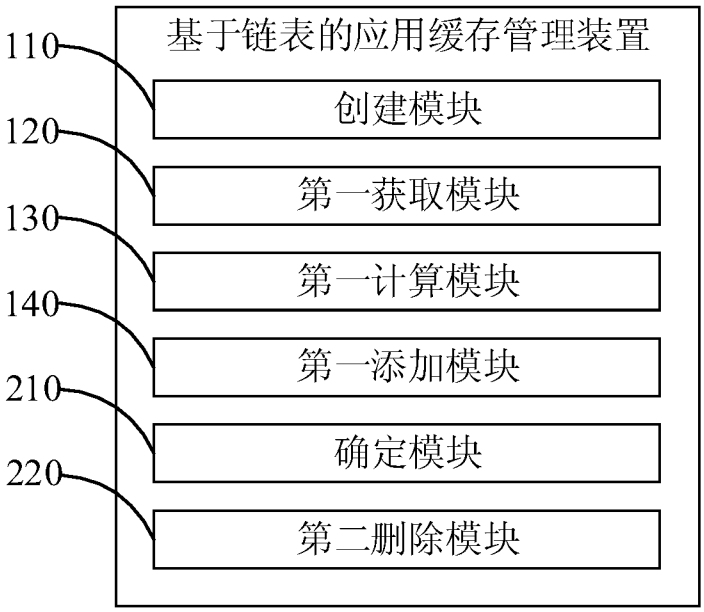
基于链表的应用缓存管理装置
110
创建模块
120
第一获取模块
130
第一计算模块
140
第一添加模块
210
确定模块
220
第二删除模块

图 8

INTERNATIONAL SEARCH REPORT

International application No.
PCT/CN2016/076296

A. CLASSIFICATION OF SUBJECT MATTER

G06F 12/0871 (2016.01) 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; H04L

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)

CPRSABS; CNTXT; VEN; USTXT; CNKI: buffer, memory, storage, cache, link?? w list, chain?? w list, link?? w table, chain?? w table, size, small+, higher, application, less, big+, great+, time, delet+, clear

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	CN 104850507 A (TENCENT TECHNOLOGY (SHENZHEN) CO., LTD.) 19 August 2015 (19.08.2015) description, paragraphs [0034]-[0066]	1-16
E	CN 105786723 A (SHENZHEN SKYWORTH RGB ELECTRONIC CO., LTD.) 20 July 2016 (20.07.2016) claims 1-10	1-16
A	CN 103984639 A (NINGBO SANXING ELECTRIC CO., LTD.) 13 August 2014 (13.08.2014) the whole document	1-16
A	US 7536529 B1 (AMERICAN MEGATRENDS INC) 19 May 2009 (19.05.2009) the whole document	1-16

☐ 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	Date of mailing of the international search report
25 November 2016	14 December 2016
Name and mailing address of the ISA State Intellectual Property Office of the P. R. China No. 6, Xitucheng Road, Jimenqiao Haidian District, Beijing 100088, China Facsimile No. (86-10) 62019451	Authorized officer FAN, Wenjing Telephone No. (86-10) 62411255

INTERNATIONAL SEARCH REPORT
Information on patent family members

International application No.
PCT/CN2016/076296

Patent Documents referred in the Report	Publication Date	Patent Family	Publication Date
CN 104850507 A	19 August 2015	None	
CN 105786723 A	20 July 2016	None	
CN 103984639 A	13 August 2014	None	
US 7536529 B1	19 May 2009	US 8549230 B1	01 October 2013
		US 8402209 B1	19 March 2013
		US 8850124 B1	30 September 2014

国际检索报告

国际申请号
PCT/CN2016/076296

A. 主题的分类

G06F 12/0871(2016.01)i

按照国际专利分类(IPC)或者同时按照国家分类和IPC两种分类

B. 检索领域

检索的最低限度文献(标明分类系统和分类号)

G06F; H04L

包含在检索领域中的除最低限度文献以外的检索文献

在国际检索时查阅的电子数据库(数据库的名称，和使用的检索词（如使用））

CPRSABS;CNTXT;VEN;USTXT;CNKI:缓存，内存，存储，链表，应用，超出，大于，小于，超过，删除，时间，清除，buffer，memory，storage，cache，link?? w list，chain?? w list，link?? w table，chain?? w table，size，small+，higher，application，less，big+，great+，time，delet+，clear

C. 相关文件

类 型*	引用文件，必要时，指明相关段落	相关的权利要求
X	CN 104850507 A (腾讯科技深圳有限公司) 2015年 8月 19日 (2015 - 08 - 19) 说明书第0034-0066段	1-16
E	CN 105786723 A (深圳创维-RGB电子有限公司) 2016年 7月 20日 (2016 - 07 - 20) 权利要求1-10	1-16
A	CN 103984639 A (宁波三星电气股份有限公司) 2014年 8月 13日 (2014 - 08 - 13) 全文	1-16
A	US 7536529 B1 (AMERICAN MEGATRENDS INC) 2009年 5月 19日 (2009 - 05 - 19) 全文	1-16

☐ 其余文件在C栏的续页中列出。

☑ 见同族专利附件。

* 引用文件的具体类型：

“A” 认为不特别相关的表示了现有技术一般状态的文件

“E” 在国际申请日的当天或之后公布的在先申请或专利

“L” 可能对优先权要求构成怀疑的文件，或为确定另一篇引用文件的公布日而引用的或者因其他特殊理由而引用的文件（如具体说明的）

“O” 涉及口头公开、使用、展览或其他方式公开的文件

“P” 公布日先于国际申请日但迟于所要求的优先权日的文件

“T” 在申请日或优先权日之后公布，与申请不相抵触，但为了理解发明之理论或原理的在后文件

“X” 特别相关的文件，单独考虑该文件，认定要求保护的发明不是新颖的或不具有创造性

“Y” 特别相关的文件，当该文件与另一篇或者多篇该类文件结合并且这种结合对于本领域技术人员为显而易见时，要求保护的发明不具有创造性

“&” 同族专利的文件

国际检索实际完成的日期	国际检索报告邮寄日期
2016年 11月 25日	2016年 12月 14日
ISA/CN的名称和邮寄地址 中华人民共和国国家知识产权局(ISA/CN) 中国北京市海淀区蓟门桥西土城路6号 100088 传真号 (86-10)62019451	受权官员 范文婧 电话号码 (86-10)62411255

国际检索报告
关于同族专利的信息

国际申请号
PCT/CN2016/076296

检索报告引用的专利文件			公布日 (年/月/日)	同族专利			公布日 (年/月/日)
CN	104850507	A	2015年 8月 19日	无			
CN	105786723	A	2016年 7月 20日	无			
CN	103984639	A	2014年 8月 13日	无			
US	7536529	B1	2009年 5月 19日	US	8549230	B1	2013年 10月 1日
				US	8402209	B1	2013年 3月 19日
				US	8850124	B1	2014年 9月 30日