



(19)中華民國智慧財產局

(12)發明說明書公告本

(11)證書號數：TW I734900 B

(45)公告日：中華民國 110 (2021) 年 08 月 01 日

(21)申請案號：107109430

(22)申請日：中華民國 107 (2018) 年 03 月 20 日

(51)Int. Cl. : **G06F21/60 (2013.01)**

|                    |    |            |
|--------------------|----|------------|
| (30)優先權：2017/03/20 | 美國 | 62/474,039 |
| 2017/09/21         | 美國 | 62/561,625 |
| 2017/09/22         | 美國 | 62/562,219 |
| 2018/01/19         | 美國 | 15/876,028 |

(71)申請人：南韓商三星電子股份有限公司 (南韓) SAMSUNG ELECTRONICS CO., LTD. (KR)  
南韓

(72)發明人：奇亮爽 KI, YANG SEOK (KR)

(74)代理人：林孟閔；盧珮君；陳怡如

(56)參考文獻：

|    |                |    |                |
|----|----------------|----|----------------|
| TW | 201315212A     | TW | 201443647A     |
| US | 8504535B1      | US | 2015/0019937A1 |
| US | 2015/0363269A1 |    |                |

審查人員：陳奕昌

申請專利範圍項數：20 項 圖式數：7 共 38 頁

(54)名稱

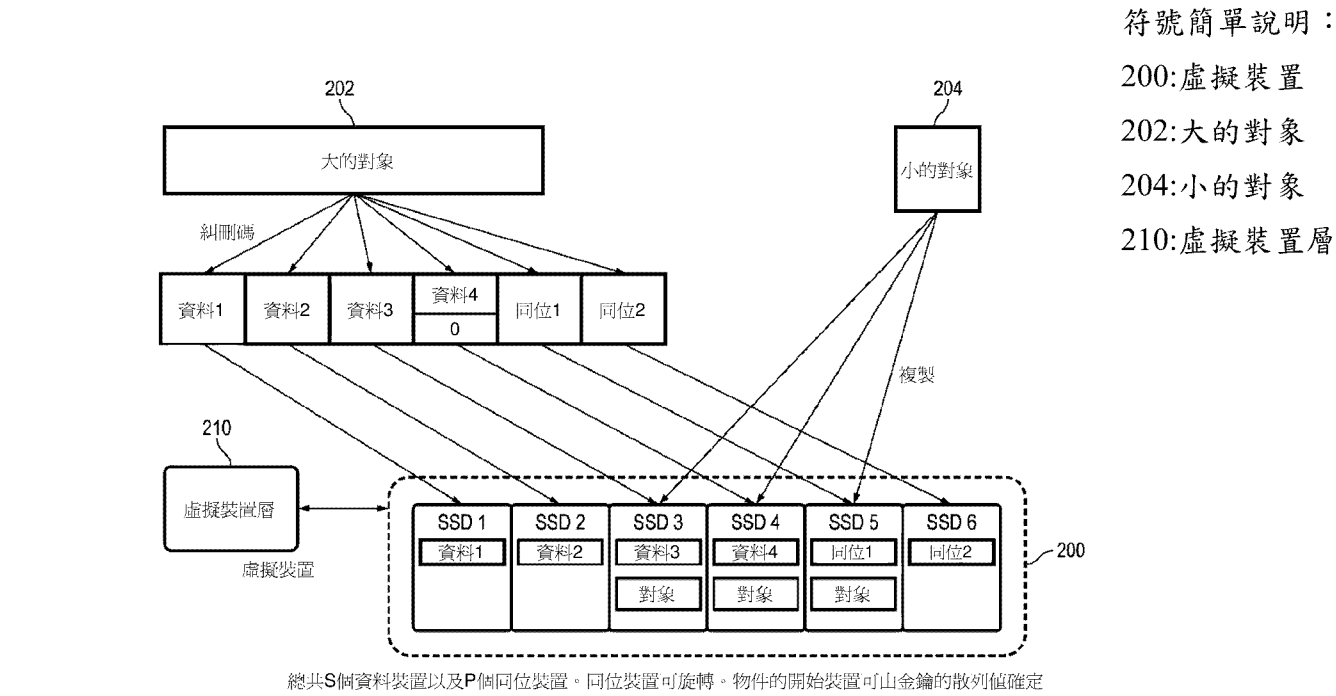
儲存裝置、利用虛擬裝置層儲存物件的方法以及由虛擬裝置層利用金鑰讀取物件的方法

(57)摘要

一種儲存裝置及金鑰值固態驅動器。儲存裝置包括：多個記憶體裝置，被配置成利用無狀態資料保護的虛擬金鑰值固態驅動器裝置；以及虛擬裝置層，被配置成管理所述虛擬裝置以通過以下方式儲存物件：根據所述物件各自的大小向所述物件中的一些物件應用糾刪編碼、以及向所述物件中的其他物件應用複製。

A storage device and a key value solid state drive. The storage device includes: a plurality of memory devices configured as a virtual device utilizing stateless data protection; and a virtual device layer configured to manage the virtual device to store objects by applying erasure coding to some of the objects and replication to other ones of the objects depending on respective sizes of the objects.

指定代表圖：



【圖2】



I734900

## 【發明摘要】

【中文發明名稱】儲存裝置、利用虛擬裝置層儲存物件的方法以及由虛擬裝置層利用金鑰讀取物件的方法

【英文發明名稱】 STORAGE DEVICE, METHOD OF UTILIZING VIRTUAL DEVICE LAYER TO STORE OBJECTS AND METHOD OF READING OBJECT WITH A KEY BY VIRTUAL DEVICE LAYER

【中文】一種儲存裝置及金鑰值固態驅動器。儲存裝置包括：多個記憶體裝置，被配置成利用無狀態資料保護的虛擬金鑰值固態驅動器裝置；以及虛擬裝置層，被配置成管理所述虛擬裝置以通過以下方式儲存物件：根據所述物件各自的大小向所述物件中的一些物件應用糾刪編碼、以及向所述物件中的其他物件應用複製。

【英文】 A storage device and a key value solid state drive. The storage device includes: a plurality of memory devices configured as a virtual device utilizing stateless data protection; and a virtual device layer configured to manage the virtual device to store objects by applying erasure coding to some of the objects and replication to other ones of the objects depending on respective sizes of the objects.

【指定代表圖】圖2。

【代表圖之符號簡單說明】

200：虛擬裝置

202：大的對象

204：小的對象

210：虛擬裝置層

【特徵化學式】

無

## 【發明說明書】

【中文發明名稱】儲存裝置、利用虛擬裝置層儲存物件的方法以及由虛擬裝置層利用金鑰讀取物件的方法

【英文發明名稱】 STORAGE DEVICE, METHOD OF UTILIZING VIRTUAL DEVICE LAYER TO STORE OBJECTS AND METHOD OF READING OBJECT WITH A KEY BY VIRTUAL DEVICE LAYER

[相關申請的交叉參考]

【0001】 本申請主張在 2017 年 3 月 20 號提出申請且標題為金鑰值固態驅動器的美國臨時專利申請第 62/474,039 號、在 2017 年 9 月 21 號提出申請且標題為用於金鑰值儲存的混合無狀態資料保護方法及系統的美國臨時專利申請第 62/561,625 號、以及在 2017 年 9 月 22 號提出申請且標題為用於金鑰值儲存的混合無狀態資料保護方法及系統的美國臨時專利申請第 62/562,219 號的優先權及權利，所述三個美國臨時專利申請的全部內容均併入本案供參考。

【技術領域】

【0002】 本發明概念涉及金鑰值儲存系統。

【先前技術】

【0003】 傳統固態驅動器（solid state drive，SSD）通常僅使用區

塊介面( block interface )並通過獨立盤的冗餘陣列( redundant array of independent disks , RAID )、糾刪編碼、或複製來提供資料可靠性。隨著物件格式變得在大小方面可變且變得非結構化，期望在物件與區塊級介面之間進行有效的資料轉換。此外，可取的是在保持空間效率及快速存取時間特性的同時確保資料可靠性。

### 【發明內容】

【0004】 根據本發明的示例性實施例涉及與區塊裝置( block device )不同的金鑰值儲存系統(例如，金鑰值固態驅動器)。

【0005】 本發明的一些示例性實施例涉及如何針對金鑰值固態驅動器實現資料可靠性。將基於空間開銷( space overhead )的複製與糾刪編碼的混合應用至一組金鑰值固態驅動器，所述金鑰值固態驅動器可實現物件的無狀態可變長度糾刪碼。

【0006】 本發明的一些示例性實施例具有一個或多個以下特性：1) 針對每一可變物件而不針對每一固定區塊提供可靠性；2) 可將複製與糾刪編碼混合以針對單一磁片組實現物件的目標可靠性；3) 空間效率是主要指標且性能是次要指標以為物件確定正確的技術；4) 與獨立盤的冗餘陣列( RAID )類似，機制是無狀態的；5) 對複製或糾刪編碼來說，不需要儲存額外的資訊；以及 6) 無論物件大小如何，不需要讀-修改-寫( read-modify-write )用於更新。

【0007】 本發明的一些示例性實施例提供一種實現一組金鑰值固態驅動器的可靠性的方法。此外，示例性實施例可避免針對由於根據示例性實施

例在對區塊內的一部分資料進行更新的情形中的區塊裝置發生的讀-修改-寫，可靠性針對每一物件（例如，可變物件）而不是針對每一區塊（例如，固定區塊）來提供。

【0008】 根據本發明的示例性實施例，一種儲存裝置包括：多個記憶體裝置，被配置成利用無狀態資料保護的虛擬裝置；以及虛擬裝置層，被配置成管理所述虛擬裝置以通過以下方式儲存物件：根據所述物件各自的大小向所述物件中的一些物件應用第一資料保護、以及向所述物件中的其他物件應用第二資料保護。

【0009】 所述記憶體裝置可被配置成一個或多個資料裝置以及一個或多個同位裝置。

【0010】 所述第一資料保護可包括糾刪編碼，且所述第二資料保護可包括複製。

【0011】 當所述物件中的對應一者被分類為大的物件時可利用所述糾刪編碼進行資料保護。

【0012】 當 $((P+1)*O > (S+P)*m$  且  $O \geq S*m$ )時，所述物件中的所述對應一者可被分類為所述大的對象，其中  $O$  是指對象大小； $P$  是指同位裝置的數目； $S$  是指資料裝置的數目；且  $m$  是指容許的最小大小值。

【0013】 當所述物件中的對應一者被分類為小的物件時可利用所述複製進行資料保護。

【0014】 當 $((P+1)*O \leq (S+P)*m)$ 時，所述物件中的所述對應一者可被分類為所述小的對象，其中  $O$  是指對象大小； $P$  是指同位裝置的數目； $S$  是指資料裝置的數目；且  $m$  是指容許的最小大小

值。

【0015】 當所述物件中的對應一者既不被分類為大的對象也不被分類為小的物件時，可基於性能指標及資料使用特性利用所述糾刪編碼或所述複製進行資料保護。

【0016】 當 $((P+1)*O > (S+P)*m) > S*m > O$ 時，所述物件中的所述對應一者可被分類為中等對象，其中  $O$  是指對象大小； $P$  是指同位裝置的數目； $S$  是指資料裝置的數目；且  $m$  是指容許的最小大小值。

【0017】 當儲存一個或多個大的物件時，所述同位裝置可以是固定的。

【0018】 當儲存一個或多個大的物件時，所述同位裝置可旋轉。

【0019】 所述記憶體裝置可包括固態驅動器。

【0020】 根據本發明的另一示例性實施例，提供一種利用虛擬裝置層在包括多個記憶體裝置的虛擬裝置中儲存物件的方法。所述方法包括：由所述虛擬裝置層判斷所述物件中的對應一者是大還是小；如果所述物件中的所述對應一者被分類為大的對象：確定用於糾刪編碼的塊（**chunk**）大小以及所述物件中的所述對應一者的資料塊（**data chunk**）的填充量；使用糾刪編碼來計算  $P$  個同位塊；確定用以儲存所述資料塊及同位塊的所述記憶體裝置；以及將所述資料塊及同位塊寫入所述記憶體裝置；且如果所述物件中的所述對應一者被分類為小的對象：確定用於資料及複製物的所述記憶體裝置；以及將所述資料及所述複製物寫入所述記憶體裝置。

【0021】 當所述對象中的所述對應一者既不是大的物件也不是小



的物件時，所述物件中的所述對應一者可被分類為中等物件，且可基於性能指標及資料使用特性應用所述複製或所述糾刪編碼。

【0022】 對應於所述物件中的至少兩者的所述同位塊可儲存在所述記憶體裝置的固定子集上。

【0023】 對應於所述物件中的不同者的所述同位塊可不儲存在所述記憶體裝置的固定子集上。

【0024】 對應於所述物件中的至少兩者的所述資料及所述複製物可儲存在所述記憶體裝置中的不同者上。

【0025】 所述資料塊中的至少一者可被填充以零。

【0026】 根據本發明的另一示例性實施例，提供一種由虛擬裝置層利用金鑰從包括多個記憶體裝置的虛擬裝置讀取物件的方法。所述方法包括：由所述虛擬裝置層向所有所述記憶體裝置發送讀取請求；以及由所述虛擬裝置層接收來自所述記憶體裝置的回應，其中如果所述物件是大的物件，那麼由所述虛擬裝置層接收資料塊及同位塊以利用糾刪編碼重建所述物件，且如果所述物件是小的物件，那麼所述資料塊是所述物件或是所述物件的複製物。

【0027】 所述金鑰可包括用於從所述多個裝置中確定開始裝置（start device）或主裝置（primary device）的散列（金鑰）。

### 【圖式簡單說明】

【0028】 以下，將參照附圖更詳細地闡述示例性實施例，在所有附圖中相同的參考編號指代相同的元件。然而，本發明可實現為各種不同形式，而不應被視為僅限於本文中所說明的實施例。確切來說，提供這些實施例作為實例是為了使公開內容將透徹及完

整，並將向所屬領域中的技術人員充分傳達本發明的方面及特徵。因此對所屬領域中的普通技術人員完全理解本發明的方面及特徵來說非必要的工藝、元件及技術可不再進行闡述。除非另有說明，否則在所有附圖及書面說明通篇中相同的參考編號表示相同的元件，且因此將不再對其予以重複贅述。在圖式中，為清晰起見可誇大各元件、層及區的相對大小。

【0029】 儘管已說明並闡述了本發明的某些實施例，但所屬領域中的普通技術人員將理解，在不背離由以上權利要求書及其等效範圍界定的本發明的精神及範圍的條件下，可對所述實施例作出某些修改及變化。舉例來說，如所屬領域中的技術人員可理解，在不背離本發明的精神及範圍的條件下，各種圖式中的示例性實施例的特徵可進行組合。

#### 【0030】

圖 1 是根據本發明示例性實施例的金鑰值（key value，KV）固態驅動器（SSD）的示意圖。

圖 2 是根據示例性實施例說明包括一組裝置的虛擬裝置以及在所述虛擬裝置中對物件的儲存的概念圖。

圖 3 是根據本發明示例性實施例將物件寫入虛擬裝置的流程圖。

圖 4 是根據示例性實施例說明以共用同位方式（shared parity manner）將大的物件儲存在圖 2 所示的虛擬裝置中的概念圖。

圖 5 是根據示例性實施例說明以專用同位方式（dedicated

parity manner) 將大的物件儲存在圖 2 所示的虛擬裝置中的概念圖。

圖 6 是根據示例性實施例說明將小的物件儲存在圖 2 所示的虛擬裝置中的概念圖。

圖 7 是根據本發明示例性實施例從虛擬裝置讀取物件的流程圖。

### 【實施方式】

【0031】 應理解，儘管本文中可能使用“第一 (first)”、“第二 (second)”、“第三 (third)”等用語來闡述各種元件、元件、區、層及/或區段，然而這些元件、元件、區、層及/或區段不應受這些用語限制。這些用語僅用於區分元件、元件、區、層或區段與另一元件、元件、區、層或區段。因此，在不背離本發明的精神及範圍的條件下，以下闡述的第一元件、元件、區、層、或區段亦可被稱為第二元件、元件、區、層、或區段。

【0032】 應理解，當稱一個元件或層位於另一元件或層“上 (on)”，“連接到 (connected to)”或“耦合到 (coupled to)”另一元件或層時，所述元件或層可直接位於所述另一元件或層上，直接連接到或耦合到所述另一元件或層，抑或可存在一個或多個中間元件或層。另外，還應理解，當稱一個元件或層“位於”兩個元件或層“之間 (between)”時，所述元件或層可為所述兩個元件或層之間的唯一元件或層，抑或也可存在一個或多個中間元件。

【0033】 本文中所使用的術語僅是為了闡述特定實施例而並非旨

在限制本發明。除非上下文清楚地另外指明，否則本文中所使用的單數形式“一（a 及 an）”旨在也包括複數形式。應進一步理解，當在本說明書中使用用語“包括（comprises 及 comprising）”及“包含（includes 及 including）”時，是指明所陳述特徵、整數、步驟、操作、元件及/或元件的存在，但不排除一個或多個其他特徵、整數、步驟、操作、元件、元件及/或其群組的存在或添加。本文中所使用的用語“及/或（and/or）”包括相關所列項其中一個或多個項的任意及所有組合。當例如“...中的至少一個（at least one of）”等表達位於一系列元件之前時，是修飾整個系列元件而不是修飾所述一系列元件中的個別元件。

【0034】 本文所使用的用語“大體上（substantially）”、“大約（about）”及類似用語用作近似用語而非程度用語，並且旨在考慮所屬領域中的普通技術人員將認識到的測量值或計算值的固有偏差。此外，在闡述本發明的實施例時使用的“可（may）”是指“本發明的一個或多個實施例”。本文中所使用的用語“使用（use）”、“正使用（using）”、及“被使用（used）”可被視為分別與用語“利用（utilize）”、“正利用（utilizing）”、及“被利用（utilized）”同義。此外，用語“示例性”旨在指實例或例證。

【0035】 根據本文中所述的本發明的實施例，電子裝置或電氣裝置及/或任意其他相關的裝置或元件（例如，主機、固態驅動器、記憶體裝置、及虛擬裝置層）可利用任意適當的硬體、韌件（例如，專用積體電路）、軟體、或軟體、韌件及硬體的適當組合來實現。舉例來說，這些裝置的各種元件（例如，金鑰值固態驅動器、

主機、固態驅動器、記憶體裝置、及虛擬裝置層）可形成在一個積體電路（integrated circuit，IC）晶片上或形成在單獨的積體電路晶片上。此外，這些裝置的各種元件可在柔性印刷電路膜、載帶封裝（tape carrier package，TCP）、印刷電路板（printed circuit board，PCB）上實現，或可形成在一個基板上。此外，這些裝置的各種元件可為在一個或多個計算裝置中在一個或多個處理器上運行的進程（process）或執行緒（thread），用於執行電腦程式指令並與其他系統元件交互作用以執行本文中所述的各種功能。所述電腦程式指令儲存在記憶體中，所述記憶體可在使用標準記憶體裝置（例如，隨機存取記憶體（random access memory，RAM）或快閃記憶體（例如，NAND 快閃記憶體）裝置）的計算裝置中實現。所述電腦程式指令也可被儲存在其他非暫時性電腦可讀介質（例如，CD-ROM、或快閃驅動器等）中。此外，所屬領域中的技術人員應認識到，在不背離本發明的示例性實施例的精神及範圍的條件下，各種計算裝置的功能可組合或集成到單個計算裝置中，或特定計算裝置的功能可分佈在一個或多個其他計算裝置上。

【0036】 除非另外定義，否則本文中所用的所有用語（包括技術及科學用語）的意義均與本發明所屬領域中的普通技術人員所通常理解的意義相同。應進一步理解，用語（例如在常用字典中所定義的用語）應被解釋為具有與其在相關技術的上下文及/或本說明書中的含義一致的含義，且除非在本文中明確定義，否則不應將其解釋為具有理想化或過於正式的意義。

【0037】 圖 1 是根據本發明示例性實施例的金鑰值（key value，

KV) 固態驅動器 (SSD) 10 的示意圖。根據示例性實施例的儲存系統 (或儲存裝置) 包括一個或多個金鑰值固態驅動器, 例如圖 1 中所示的一個金鑰值固態驅動器, 但本發明並非僅限於此。

【0038】 根據本發明的示例性實施例, 金鑰值固態驅動器 10 中的金鑰值應用程式介面 15 與不需要傳統區塊映射的使用者金鑰值裝置驅動器 20 一起運作。

【0039】 根據本發明的示例性實施例, 包括金鑰值固態驅動器 10 的儲存系統通過以下方式利用混合無狀態資料保護方法: 根據物件各自的大小向一些物件應用第一資料保護 (例如, 糾刪編碼), 並向其他物件應用第二資料保護 (例如, 複製), 以實現所需的可靠性 (例如, 目標可靠性)。如此一來, 可在不犧牲可靠性的情況下提供空間高效型 (space-efficient) 解決方案。在根據一些實施例金鑰值固態驅動器 10 自身可執行混合無狀態資料保護方法時, 當所述混合無狀態資料保護方法是由例如儲存系統執行時, 對驅動器 (固態驅動器) 的管理 (例如, 虛擬裝置層的操作) 可變得更容易。

【0040】 根據本發明的示例性實施例, 可對對象進行分類以獲得空間效率, 可基於大小對所述對象進行分類, 且針對每一大小等級可使用不同的備份 (backup) 方法。

【0041】 如果對一個物件進行糾刪編碼的空間開銷小於對所述物件進行複製的空間開銷, 那麼所述物件可被視為大的物件。在此種情形中, 由於糾刪編碼具有較小的空間佔用區域 (space footprint), 因此可期望使用糾刪編碼。換句話說, 當物件滿足以下不等式 ( $((P+1)*O > (S+P)*m$  且  $O \geq S*m$ ) 時, 可將物件視為

大的物件，其中  $O$  是物件值大小。在本文中且在以下不等式中， $O$ =物件大小（即，物件的大小）； $P$ =同位裝置計數（即，在虛擬裝置中同位裝置的數目）； $S$ =資料裝置計數（即，虛擬裝置中資料裝置的數目）；且  $m$ =容許的最小大小值（即，個別裝置的所有最小值大小中的最大值）。舉例來說，根據示例性實施例的“容許的最小大小值”是指在不違反系統中任意裝置的最小值大小要求的情況下可被儲存到系統中任意裝置的值大小。每一裝置具有裝置支援的最小物件大小。由於根據示例性實施例物件被分成用於所有裝置的相等大小，因此所述大小應大於裝置支援的任意最小大小。如果嘗試儲存大小小於  $m$  的物件，那麼至少一個裝置無法儲存所述物件。

**【0042】** 換句話說，當滿足以下條件中的兩者時將物件視為大的物件：1) 物件的大小  $O$  乘以同位裝置的數目加一（ $P+1$ ）大於容許的最小大小值  $m$  乘以資料裝置  $S$  的數目與同位裝置  $P$  的數目的總和  $S+P$ ；以及 2) 物件的大小  $O$  大於或等於資料裝置  $S$  的數目乘以容許的最小大小值  $m$ 。

**【0043】** 根據示例性實施例，可對大的物件進行糾刪編碼。也就是說，可將物件分成  $S$  個塊（即，資料塊或  $S$  個部分），且使用所述  $S$  個塊來計算同位塊（即，同位部分）。如本文中其他地方所述， $S$  個塊及  $P$  個塊中的每一者可儲存在對應的裝置中。

**【0044】** 當對一個物件進行複製的空間開銷小於對所述物件進行糾刪編碼的空間開銷時，所述物件可被視為小的物件。在此種情形中，由於複製提供更好的讀取性能且可比相對複雜的糾刪編碼更好地處理更新，因此可期望使用複製。從應用中繼資料

( application metadata ) 常常為小的觀察結果來看，此也是合理的。換句話說，如果物件滿足以下不等式 (  $((P+1)*O \leq (S+P)*m)$  )，那麼所述物件可被視為小的物件且可被複製。

【0045】 換句話說，當物件大小  $O$  乘以同位裝置的數目加一 (  $P+1$  ) 小於資料裝置  $S$  的數目與同位裝置  $P$  的數目的總和  $S+P$  乘以容許的最小大小值  $m$  時，可將物件視為小的物件。

【0046】 可存在一些其中物件可被分類為小的物件或大的物件的灰色區域。舉例來說，當物件滿足以下不等式 (  $((P+1)*O > (S+P)*m > S*m > O)$  ) 時，所述物件可被視為中等物件，且可基於性能指標 ( 例如，空間對 ( vs. ) 存取時間 ) 及/或資料使用特性 ( 例如，更新頻率 ) 來使用複製或糾刪碼。

【0047】 換句話說，當物件大小  $O$  乘以同位裝置的數目加一 (  $P+1$  ) 大於資料裝置 (  $S$  ) 的數目與同位裝置 (  $P$  ) 的數目的總和乘以容許的最小大小值  $m$ 、資料裝置 (  $S$  ) 的數目與同位裝置 (  $P$  ) 的數目的總和乘以容許的最小大小值  $m$  大於資料裝置的數目乘以容許的最小大小值、且資料裝置的數目乘以容許的最小大小值大於物件大小  $O$  時，可將物件視為中等物件。

【0048】 舉例來說，如果性能更為重要且物件被頻繁更新，那麼複製可為更好的選擇。在此種情形中，可將中等物件分類為小的物件。舉例來說，在以下不等式 (  $((P+1)*O \leq (S+P)*m)$  或  $((P+1)*O > (S+P)*m)$  且  $S*m > O$ ，即如果  $((P+1)*O \leq (S+P)*m$  或  $O < S*m)$  ) 的情形中，根據示例性實施例可將物件分類為小的物件。

【0049】 另舉例來說，如果空間效率更為重要，那麼可使用糾刪編碼。在此種情形中，可將中等物件分類為大的物件。舉例來說，



在滿足以下不等式 ( $((P+1)*O > (S+P)*m$  且  $O \geq S*m$ ) 或  $((P+1)*O > (S+P)*m) > S*m > O = ((P+1)*O > (S+P)*m)$ ，即如果 ( $((P+1)*O > (S+P)*m)$ ) 的情形中，根據示例性實施例可將物件分類為大的物件。

【0050】圖 2 是說明包括一組裝置(固態驅動器 1、固態驅動器 2、固態驅動器 3、固態驅動器 4、固態驅動器 5、固態驅動器 6)的虛擬裝置 200 以及在虛擬裝置 200 中對物件(大的物件 202 及小的物件 204)的儲存的概念圖。在所述裝置中，固態驅動器 1、固態驅動器 2、固態驅動器 3 及固態驅動器 4 被配置成資料裝置，且固態驅動器 5 及固態驅動器 6 是同位裝置。儘管出於說明目的在圖 2 中僅示出了四個資料裝置固態驅動器 1、固態驅動器 2、固態驅動器 3 及固態驅動器 4 以及兩個同位裝置固態驅動器 5 及固態驅動器 6，但虛擬裝置 200 中資料裝置及同位裝置的數目並非僅限於此。此外，不同的固態驅動器(SSD)可被配置成資料裝置及同位裝置。

【0051】舉例來說，虛擬裝置 200 可包括總共  $S$  個資料裝置以及  $P$  個同位裝置，所述裝置中的同位裝置可為固定的或可旋轉(因此，我們可參照圖 2 看出使用示例性  $S$  值 4 及示例性  $P$  值 2)。例如，當同位裝置可旋轉時，並非不同的大的物件的所有同位塊(或同位部分)都可儲存在相同的同位裝置中，且一些資料裝置可用作於一個或多個大的物件的同位裝置。換句話說，當同位裝置是固定的時，對應於物件的“ $P$ ”個同位塊儲存在記憶體裝置的同一集合“ $P$ ”上，而當同位裝置可旋轉時，對應於物件的“ $P$ ”個同位塊沒有必要儲存在記憶體裝置的同一集合上。此外，所述裝置

可以平面方式或層級形式進行組織。在多個裝置上擴展或複製的物件的開始裝置可由金鑰的散列值（**hash value**）確定。

【0052】 此外，根據需求及/或使用者的設計選擇可將資料裝置重新配置成同位裝置，或可將同位裝置重新配置成資料裝置。舉例來說，虛擬裝置 200 的裝置的數目可基於可靠性目標進行配置。對於糾刪編碼來說，為了容忍  $P$  次故障，裝置的總數目可為資料裝置的數目（ $S$ ）與同位裝置的數目（ $P$ ）的總和。對於複製來說，可容忍  $P$  次故障的裝置的總數目可為  $P+1$ 。裝置的容量可為彼此相同或類似。

【0053】 根據本發明的示例性實施例，虛擬裝置 200 中（或對應於虛擬裝置 200）的裝置的集合構成作為可靠性管理的單元的群組。所述群組的裝置（固態驅動器 1、固態驅動器 2、固態驅動器 3、固態驅動器 4、固態驅動器 5 及固態驅動器 6）可存在於單個伺服器或機架（**rack**）內、或存在於多個伺服器或機架上，且所述裝置可被結構化為具有層級架構或平面架構。

【0054】 包括所述一組裝置的虛擬裝置 200 可由被稱為虛擬裝置層 210 的層管理，使得所述一組裝置可被呈現為單個虛擬裝置。虛擬裝置層 210 可為無狀態的。虛擬裝置層 210 可在執行時間緩存並保持裝置的最小中繼資料資訊，例如物件的數目及/或可用容量等。應注意，根據示例性實施例的虛擬裝置層 210 不需要保持金鑰資訊（例如，沒有針對金鑰的映射）。虛擬裝置 200 的容量可由所有裝置容量中的最小裝置容量（例如，圖 2 中固態驅動器 1、固態驅動器 2、固態驅動器 3、固態驅動器 4、固態驅動器 5 及固態驅動器 6 的容量中的最少者）乘以群組中裝置的數目來確定。

【0055】 虛擬裝置層 210 可知曉每一裝置可處理的最小值大小及最大值大小。虛擬裝置層 210 可確定虛擬裝置 200 的最小值大小及最大值大小。舉例來說，根據示例性實施例，個別裝置的所有最小值大小中的最大值 ( $m_i$ ) 可被定義為虛擬裝置 200 的最小值大小 ( $m$ )，而個別裝置的所有最大值大小中的最小值 ( $M_i$ ) 可被定義為虛擬裝置 200 的最大值大小 ( $M$ )。在其他實施例中，虛擬裝置的最大值大小 ( $M$ ) 可由個別裝置的所有最大值大小中的最小值 ( $M_i$ ) 乘以資料裝置的數目 ( $S$ ) 定義。

【0056】 根據本發明一些示例性實施例的虛擬裝置 200 可利用所屬領域中的技術人員已知的任意適當糾刪編碼演算法，且可使用可用的最大距離可分 (maximum distance separable, MDS) 演算法，例如裡德-所羅門 (Reed-Solomon, RS) 碼。如可在圖 2 中看到，將具有同位值二的糾刪碼 (erasure code, EC) (即，糾刪編碼演算法) 應用至大的物件 202，使得使用同位 (Parity) 1 及同位 2。

【0057】 根據示例性實施例，可將物件 (例如，大的物件 202) 分成  $S$  個塊並編碼 (具有相同大小並分佈在資料裝置及同位裝置 (即， $S+P$  個裝置) 上)。舉例來說，已經糾刪編碼的大的物件 202 可被分成資料 (Data) 1、資料 2、資料 3、資料 4、同位 1 以及同位 2。根據示例性實施例，物件佔據的實際儲存空間可被稱為頻帶 (band)。對於糾刪編碼來說頻帶可跨越  $S+P$  個裝置，而對於複製來說頻帶可跨越  $P+1$  個裝置。舉例來說，對小的物件 204 應用複製。頻帶可完全包含物件 (即，整個物件可被儲存在頻帶中)。在一些實施例中，頻帶可跨越其中儲存有物件的  $S$  個塊的  $S$  個裝置。

【0058】 當物件大小未對齊到裝置的分配或對齊單元時，可對頻帶中為對象分配的額外的空間進行填充（例如，可以 0 進行填充）。舉例來說，圖 2 示出大的物件 202 的資料 4 已被填充以 0，以佔據對儲存資料 4 的所有資料位元來說非必要的額外的空間。此外，根據本發明的實施例頻帶大小可為可變的。

【0059】 圖 3 是根據本發明示例性實施例將物件寫入虛擬裝置（例如，圖 2 及圖 4 到圖 6 所示的虛擬裝置 200）的流程圖。圖 4 是根據本發明示例性實施例說明以共用同位方式將大的物件 242 及 244 儲存在虛擬裝置 200 中的概念圖。圖 5 是根據本發明示例性實施例說明以專用同位方式將大的物件 242 及 244 儲存在虛擬裝置 200 中的概念圖。圖 6 是根據本發明示例性實施例說明將小的物件 262 及 264 儲存在虛擬裝置 200 中的概念圖。

【0060】 如可在圖 3 中看到，在方框 300 中，虛擬裝置層（例如，圖 2 及圖 4 到圖 6 所示的虛擬裝置層 210）接收（例如，自主機接收）指令或命令以利用金鑰將大小為  $O$  的物件寫入虛擬裝置（例如，圖 2 及圖 4 到圖 6 所示的虛擬裝置 200）。在其他實施例中，寫入指令或命令可由虛擬裝置層回應於由主機提供的寫入指令產生。

【0061】 在方框 302 中，虛擬裝置層使用上述不等式判斷物件是否為大的物件。舉例來說，當  $((P+1)*O > (S+P)*m$  且  $O \geq S*m$  時，可將物件視為大的物件，其中  $O$ =物件大小； $P$ =同位裝置數目； $S$ =資料裝置數目；且  $m$ =容許的最小大小值（即，個別裝置的所有最小值大小中的最大值）。

【0062】 當將物件分類為大的物件時，如在方框 312 中所示，虛

擬裝置層確定用於糾刪編碼的資料塊的大小以及一個或多個資料塊的填充（例如，以零進行填充）量。然後，將物件分成具有相同大小的  $S$  個塊，考慮通過填充進行對齊，且然後如在方框 314 中所示，利用所屬領域中的技術人員已知的適當的糾刪編碼演算法從  $S$  個塊產生（例如，計算） $P$  個碼塊（即， $P$  個同位塊）。

【0063】 然後在方框 316 中，虛擬裝置層基於分佈策略（distribution policy）確定用於儲存資料塊及同位塊的裝置（即， $S$  個裝置及  $P$  個裝置）。舉例來說，所述分佈策略可涉及通過金鑰的散列值來確定物件的開始裝置及/或將資料塊及/或同位塊儲存在固定的裝置及/或點上。在方框 318 中，將資料塊及同位塊寫入對應的裝置。舉例來說，將  $S+P$  個塊分佈且儲存在  $S+P$  個裝置（例如，圖 2 所示的固態驅動器 1、固態驅動器 2、固態驅動器 3、固態驅動器 4、固態驅動器 5 及固態驅動器 6）中。對於圖 4 所示的旋轉同位裝置來說，例如，在利用金鑰的散列確定的裝置處開始資料寫入，且依次寫入每一區塊（即，塊（以及同位區塊，即同位塊）），從第一裝置上的第一資料開始。對於如圖 5 所示的固定同位裝置，例如，將所有資料區塊及同位區塊（即，塊）儲存在預先指定的裝置中。此處，開始裝置也被預先指定。對於小的物件，也通過使金鑰散列而確定開始裝置及複製裝置。

【0064】 如可在圖 4 中看到，同位裝置可被共用（即，旋轉）。換句話說，單個裝置可根據被儲存的大的物件而被用作用於儲存資料塊的資料裝置或用於儲存同位塊的同位裝置兩者。舉例來說，物件 242 可被分成資料 1、資料 2、資料 3、資料 4（被填充以 0）、同位 1 以及同位 2，且物件 244 也可被分成資料 1、資料 2、資料

3、資料 4（被填充以 0）、同位 1 以及同位 2。可在圖 4 中看到，在大的物件 242 的資料 1、資料 2、資料 3 以及資料 4 分別被儲存在虛擬裝置 200 的固態驅動器 1、固態驅動器 2、固態驅動器 3 以及固態驅動器 4 中時，大的物件 244 的資料 1、資料 2、資料 3 以及資料 4 分別被儲存在虛擬裝置 200 的固態驅動器 6、固態驅動器 1、固態驅動器 2 以及固態驅動器 3 中。

【0065】此外，當物件 242 的同位 1 及同位 2 分別被儲存在虛擬裝置 200 的固態驅動器 5 以及固態驅動器 6 中時，物件 244 的同位 1 及同位 2 分別被儲存在虛擬裝置 200 的固態驅動器 4 以及固態驅動器 5 中。因此，當存在總共  $S$  個資料裝置及  $P$  個同位裝置時，所述同位裝置可旋轉，使得沒有專用同位裝置。

【0066】不同於圖 4 中所繪示的實例，圖 5 說明利用專用同位裝置（即，虛擬裝置 200 的固態驅動器 5 及固態驅動器 6）的實施方式。舉例來說，大的物件 242 以及大的物件 244 兩者的資料 1、資料 2、資料 3、資料 4（被填充以 0）、同位 1 以及同位 2 分別被儲存在虛擬裝置 200 的固態驅動器 1、固態驅動器 2、固態驅動器 3、固態驅動器 4、固態驅動器 5 及固態驅動器 6 中。

【0067】對於旋轉同位實施例且對於小的物件來說，物件的開始裝置可由金鑰的散列值確定。舉例來說，在圖 4 所示的共用同位裝置情形中，開始裝置可通過  $\text{Hash}(\text{key})\%(S+P)$  進行確定。然後，後續的資料塊及同位塊（即， $S+P$  個塊）被依序寫入  $(\text{Hash}(\text{key})+1)\%(S+P)$ 、 $(\text{Hash}(\text{key})+2)\%(S+P)$ 、...、 $(\text{Hash}(\text{key})+S+P-1)\%(S+P)$ 。如果存在專用同位裝置，那麼使用  $S$  個裝置代替（ $S+P$ ）。

【0068】 在將資料塊及同位塊寫入對應的裝置後，在方框 320 中結束（完成）大的物件寫入進程。

【0069】 當在方框 302 中未將物件確定為大的物件時，進程進入方框 304，在方框 304 中，判斷物件是否為小的物件（即， $((P+1)*O \leq (S+P)*m)$ 是否成立）。如果確定物件為小的物件，那麼虛擬裝置層繼續執行複製並在方框 308 中基於分佈策略確定利用哪些裝置來儲存資料及複製物。舉例來說，所述分佈策略可涉及通過金鑰的散列值來確定物件的開始裝置及/或將資料及/或複製物儲存在固定的裝置及/或點上。然後在方框 310 中，將資料及複製物寫入對應的裝置。

【0070】 根據示例性實施例，可為物件生成  $P+1$  個複製物（包括一個資料副本及  $P$  個同位副本），考慮通過填充進行對齊，且所述  $P+1$  個複製物可被分佈到  $P+1$  個裝置上。如圖 6 所示，例如，物件 1 262 被複製三次（包括資料及 2 個複製物），且副本分別被儲存在虛擬裝置 200 的固態驅動器 1、固態驅動器 2 以及固態驅動器 3 中。類似地，物件 2 264 被複製三次（包括資料及 2 個複製物），且副本分別被儲存在虛擬裝置 200 的固態驅動器 3、固態驅動器 4 以及固態驅動器 5 中。在圖 6 所示的實例中，虛擬裝置 200 包括總共  $S$  個資料裝置以及  $P$  個同位裝置。此外，由於物件 1 262 及物件 2 264 兩者均為小的物件，因此在圖 6 所示的實例中不使用糾刪編碼。

【0071】 可利用金鑰的散列值在  $S+P$  個裝置中選擇主裝置。可基於儲存組織、及/或性能等確定地選擇  $P$  個複製物。舉例來說，當資料可被儲存在主裝置中時，複製物可被儲存在

$(\text{Hash}(\text{key})+1)\%(S+P)$ 、 $(\text{Hash}(\text{key})+2)\%(S+P)$ 、...、 $(\text{Hash}(\text{key})+P)\%(S+P)$ 上，或儲存在不同的節點、機架上，而無論是否使用專用同位裝置。

【0072】現在返回圖 3，當在方框 304 中確定物件不是小的物件時（即，當物件既不是大的物件（參見方框 302）也不是小的對象（參見方框 304）時），將物件確定為中等物件（即， $(P+1)*O > (S+P)*m > S*m > O$ ），且進程進入方框 306 以判斷是否將把中等物件視為小的物件。如果將把所述物件視為小的物件，那麼進程進入方框 308 以起始小的物件儲存進程，且如果將把所述物件視為大的物件，那麼進程進入方框 312 以起始大的物件儲存進程。

【0073】圖 7 說明根據本發明示例性實施例從虛擬裝置（例如，圖 2 及圖 4 到圖 6 所示的虛擬裝置 200）讀取物件的進程。虛擬裝置層（例如，圖 2 及圖 4 到圖 6 所示的虛擬裝置層 210）不知道將讀取的物件是小還是大，因為所述虛擬裝置層不保存物件中繼資料，例如金鑰及值大小。因此，虛擬裝置層通過使用物件的使用者金鑰向所有實體裝置（即， $S+P$  個裝置）發送讀取請求而起始讀取進程（700），其中如在方框 702 中所示向所有實體裝置發生子讀取請求。在方框 704 中，虛擬裝置層從裝置接收回應。當使用者（例如，主機）請求的物件是大的物件時，如果沒有錯誤（此在方框 706 中確定），那麼所有  $S+P$  個裝置都向具有使用者金鑰的請求返回相應的回應。

【0074】舉例來說，在沒有錯誤時，如果將讀取的物件是大的物件，那麼所有裝置（即， $S+P$  個裝置）將作出回應。然而，當  $N$  個裝置具有錯誤時，那麼僅有  $S+P-N$  個裝置可作出回應。只要虛



擬裝置層接收具有相同大小的任意  $S$  個塊（即，與資料塊  $S$  的總數目相等的資料塊  $S$  與同位塊  $P$  的任意組合），那麼便可重建使用者物件。換句話說，只要不超過裝置的同位數目（即，數目等於  $P$  的裝置）發生故障，那麼在大的物件的情形中資料便可被重建。

**【0075】** 如果所接收的塊的總數目小於  $S$  或塊的大小不相同，那麼存在錯誤。在所有裝置返回不存在（**NON\_EXIST**）錯誤的情形中，可能是對不存在的物件的讀取，或可能已發生了不可恢復的錯誤。

**【0076】** 最初虛擬裝置層不知道物件類型，因此虛擬裝置層將所述類型初始化為無（**NONE**）。當物件如在方框 708 中所確定是大的物件時，在方框 718 中確定類型。如果所述類型是無（**NONE**），那麼在方框 720 中將物件類型設定為大。如果在方框 718 中沒有將類型確定為無（**NONE**），那麼在方框 732 中，虛擬裝置層檢查所述類型是否為大。如果在方框 732 中所述類型並非為大，那麼如在方框 734 中所示確定錯誤。在方框 720 中將物件類型設定為大之後或如果在方框 732 中確定物件類型為大，那麼在方框 722 中虛擬裝置層判斷其是否具有所有資料塊。

**【0077】** 如果虛擬裝置層確定已接收到所有資料塊，那麼如在方框 730 中所示結束（即，完成）讀取進程。如果沒有接收到所有資料塊，那麼虛擬裝置層在方框 724 中判斷是否已從所有裝置接收到回應。如果所有裝置均以作出回應，那麼在方框 726 中虛擬裝置層判斷其是否具有資料的至少  $S$  個塊（將所有資料塊及同位塊計算在內）。如果已接收到少於  $S$  個塊，那麼如在方框 734 中所示，虛擬機器層確定存在錯誤。如果已準確地接收到至少  $S$  個塊

（將所有所接收的資料塊及同位塊計算在內），那麼虛擬裝置層在方框 728 中利用糾刪編碼演算法以  $S$  個塊重建物件，且在方框 730 中讀取進程結束。一個或多個裝置有可能不作出回應，例如，在一個或多個裝置意外離線的情形中。因此，在一些示例性實施例中，即使並非所有的裝置均作出回應，只要已接收到至少  $S$  個塊，虛擬裝置層便如在方框 728 中所示繼續進行重建物件。

**【0078】** 如果虛擬裝置層在方框 708 中確定物件不是大的物件，那麼進程進入方框 710 中以判斷類型是否為無（**NONE**）。如果所述類型是無（**NONE**），那麼在方框 712 中將物件類型設定為小。如果所述類型不是無（**NONE**），那麼在方框 716 中判斷所述類型是否為小。此處，如果所述類型不是小，那麼如在方框 734 中所示發現錯誤。在方框 712 中將物件類型設定為小之後或如果在方框 716 中虛擬裝置層確定所述類型是小，那麼在方框 714 中虛擬裝置層判斷所接收的塊是否有效。如果所接收的塊是有效的，那麼如在方框 730 中所示結束（即，完成）讀取進程。

**【0079】** 當使用者（例如，主機）請求的物件是小的物件時，如果沒有錯誤，那麼具有複製物（即，主副本及複製物中的一者）的  $P+1$  個裝置將返回，而其他的返回通知物件不存在的錯誤。只要在方框 714 中虛擬裝置層接收到任意有效塊，那麼其便具有物件。如果所有裝置均返回不存在（**NON\_EXIST**）錯誤，那麼便不存在此物件（或存在錯誤）。如果不是所有的裝置均返回但所有返回的裝置均報告不存在（**NON\_EXIST**），那麼如在方框 734 中所示已發生了不可恢復的錯誤。

**【0080】** 如果在方框 714 中虛擬裝置層確定所述塊是無效的，那

麼在方框 724 中判斷是否已從所有裝置接收到回應。如果沒有從所有裝置接收到回應，那麼在方框 704 中虛擬裝置層繼續進行從所有裝置取得回應，並繼續所述進程以在方框 706 中判斷是否存在錯誤等等，如圖 7 所示。

【0081】 根據本發明的示例性實施例，在讀取失敗的情形中，虛擬裝置層可要求每一裝置列舉所有的物件金鑰並具有所有金鑰的總序，用於在概念上進行重建。虛擬裝置層可按次序逐個檢查所述金鑰。

【0082】 在不使用固定同位裝置的情形中，如果所述物件是大的物件，那麼虛擬裝置層可使用 Hash(key)確定金鑰的開始裝置，並基於開始裝置資訊確定應生成哪一塊（資料塊或碼塊）。在使用同位裝置的情形中，哪一塊必須被重建是顯而易見的。類似於大的物件讀取情形，利用有效塊對用於新裝置的塊進行重建。

【0083】 如果所述物件是小的物件，那麼虛擬裝置層可使用 Hash(key)確定金鑰的主裝置，並基於主裝置資訊確定哪些裝置具有複製物。如果新裝置必須具有複製物，那麼將所述物件寫入所述新裝置。重複此過程直到已拜訪了裝置上的所有物件並重建了故障裝置。

【0084】 因此，根據本發明的一個或多個示例性實施例，基於空間開銷利用糾刪編碼與複製的無狀態混合。此外，中等大小物件可例如基於訪問圖案而在糾刪編碼與複製之間切換。另外，每個對象的塊大小是可變的。此外，可能沒有必要進行因與其他物件共用空間而產生的讀-修改-寫。

【0085】 應理解，本文中所述的實施例應僅以闡述性意義進行考

慮而並非用於限制目的。在每一實施例內對特徵或方面的闡述通常應被視為可用於其它實施例中的其他類似特徵或方面。儘管已參照圖式闡述了一個或多個實施例，但所屬領域中的普通技術人員應理解，在不背離由以上權利要求書及其等效範圍所界定的精神及範圍的條件下，可作出各種形式及細節上的變化。

## 【符號說明】

### 【0086】

10：固態驅動器（SSD）

15：金鑰值應用程式介面

20：使用者金鑰值裝置驅動器

200：虛擬裝置

202：大的對象

204：小的對象

210：虛擬裝置層

242、244：大的對象

262、264：小的對象

300、302、304、306、308、310、312、314、316、318、320：

方框

700、702、704、706、708、710、712、714、716、718、720、

722、724、726、728、730、732、734：方框

## 【發明申請專利範圍】

【第1項】 一種儲存裝置，包括：

多個記憶體裝置，所述記憶體裝置中的每一者具有容許的最小大小值，且被配置成利用無狀態資料保護的虛擬裝置，其中所述容許的最小大小值為所述多個記憶體裝置所支援的所有最小大小值中的最大值；以及

虛擬裝置層，被配置成管理所述虛擬裝置以通過以下方式儲存具有各自的大小的第一物件及第二物件：根據所述第一物件及所述第二物件各自的所述大小向所述第一物件及所述第二物件中的每一者應用第一資料保護及第二資料保護中的一者，

其中所述虛擬裝置層被配置成：

對大的物件應用所述第一資料保護；

對小的物件應用所述第二資料保護；

對中等物件應用所述第一資料保護或所述第二資料保護。

【第2項】 如申請專利範圍第1項所述的儲存裝置，其中所述記憶體裝置被配置成一個或多個資料裝置以及一個或多個同位裝置。

【第3項】 如申請專利範圍第2項所述的儲存裝置，其中所述第一資料保護包括糾刪編碼，且所述第二資料保護包括複製。

【第4項】 如申請專利範圍第3項所述的儲存裝置，其中當所述第一物件及所述第二物件中的對應一者被分類為大的物件時利用所述糾刪編碼進行資料保護。

【第5項】 如申請專利範圍第4項所述的儲存裝置，其中當 $((P+1)*O > (S+P)*m$ 且 $O \geq S*m$ )時，所述第一物件及所述第二物件中的所述對應一

者被分類為所述大的對象，其中O是指對象大小；P是指同位裝置的數目；S是指資料裝置的數目；且m是指所述多個記憶體裝置的各自所述容許的最小大小值中的所述容許的最小大小值的最大值。

【第6項】 如申請專利範圍第3項所述的儲存裝置，其中當所述第一物件及所述第二物件中的對應一者被分類為小的物件時利用所述複製進行資料保護。

【第7項】 如申請專利範圍第6項所述的儲存裝置，其中當 $((P+1)*O \leq (S+P)*m)$ 時，所述第一物件及所述第二物件中的所述對應一者被分類為所述小的對象，其中O是指對象大小；P是指同位裝置的數目；S是指資料裝置的數目；且m是指所述多個記憶體裝置的各自所述容許的最小大小值中的所述容許的最小大小值的最大值。

【第8項】 如申請專利範圍第3項所述的儲存裝置，其中當所述第一物件及所述第二物件中的對應一者既不被分類為大的對象也不被分類為小的物件的中等物件時，基於所述多個記憶體裝置的可用空間、儲存在所述虛擬裝置時所述第一物件或所述第二物件的存取時間、以及所述第一物件或所述第二物件的存取有多頻繁中的一或多個來利用所述糾刪編碼或所述複製進行資料保護。

【第9項】 如申請專利範圍第8項所述的儲存裝置，其中當 $((P+1)*O > (S+P)*m) > S*m > O$ 時，所述第一物件及所述第二物件中的所述對應一者被分類為中等物件，其中O是指對象大小；P是指同位裝置的數目；S是指資料裝置的數目；且m是指所述多個記憶體裝置的各自所述容許的最小大小值中的所述容許的最小大小值的最大值。

【第10項】 如申請專利範圍第2項所述的儲存裝置，其中所述同位裝置是所述記憶體裝置中的固定子集。

【第11項】 如申請專利範圍第2項所述的儲存裝置，其中所述同位裝置包括變化子集，其中所述多個記憶體裝置的每一者被配置成運作為資料裝置及同位裝置兩者。

【第12項】 如申請專利範圍第1項所述的儲存裝置，其中所述記憶體裝置包括固態驅動器。

【第13項】 一種利用虛擬裝置層在包括多個記憶體裝置的虛擬裝置中儲存物件的方法，每一記憶體裝置具有容許的最小大小值，其中所述容許的最小大小值為所述多個記憶體裝置所支援的所有最小大小值中的最大值，所述方法包括：

由所述虛擬裝置層接收需要儲存的物件；

由所述虛擬裝置層基於  $O$ 、 $P$ 、 $S$  及  $m$  判斷所述物件是大還是小，

其中  $O$  是指對象大小； $P$  是指同位裝置的數目； $S$  是指資料裝置的數目；且  $m$  是指所述多個記憶體裝置的各自所述容許的最小大小值中的所述容許的最小大小值的最大值，以將所述物件分類為大或小；

當所述物件被分類為大的時：

確定用於糾刪編碼的塊大小以及所述物件的資料塊的填充量；

將所述物件分割成多個資料塊，各個資料塊具有用於所述糾刪編碼的所述塊大小；

使用糾刪編碼來計算  $P$  個同位塊；

確定用以儲存所述資料塊及同位塊的所述記憶體裝置；以及

將所述資料塊及同位塊寫入所述記憶體裝置；且

當所述物件被分類為小的時：

確定用於資料及複製物的所述記憶體裝置；以及

將所述資料及所述複製物寫入所述記憶體裝置。

【第14項】 如申請專利範圍第13項所述的方法，其中當所述物件既不是大的也不是小的時，所述物件被分類為中等，且基於所述多個記憶體裝置的可用空間、儲存在所述虛擬裝置時所述物件的存取時間、以及所述物件的存取有多頻繁中的一或多個來應用複製或所述糾刪編碼。

【第15項】 如申請專利範圍第13項所述的方法，其中分別對應於第一物件及第二物件的所述同位塊儲存在所述記憶體裝置的固定子集上。

【第16項】 如申請專利範圍第13項所述的方法，其中對應於第一物件及第二物件的所述同位塊儲存在所述記憶體裝置的可變子集上。

【第17項】 如申請專利範圍第13項所述的方法，其中對應於第一物件及第二物件的所述資料及所述複製物儲存在所述記憶體裝置中的不同者上。

【第18項】 如申請專利範圍第13項所述的方法，其中所述資料塊中的至少一者被填充以零。

【第19項】 一種由虛擬裝置層利用金鑰從包括多個記憶體裝置的虛擬裝置讀取物件的方法，其中所述物件基於 $O$ 、 $P$ 、 $S$ 及 $m$ 為大的、中等或小的，每一記憶體裝置具有容許的最小大小值，其中 $O$ 是指對象大小， $P$ 是指同位裝置的數目， $S$ 是指資料裝置的數目，且 $m$ 是指所述多個記憶體裝置的各自所述容許的最小大小值中的所述容許的最小大小值的最大值，所述容許的



最小大小值為所述多個記憶體裝置所支援的所有最小大小值中的最大值，  
所述方法包括：

由所述虛擬裝置層接收對應資料讀取請求的金鑰；

由所述虛擬裝置層向所有所述記憶體裝置發送讀取請求；以及

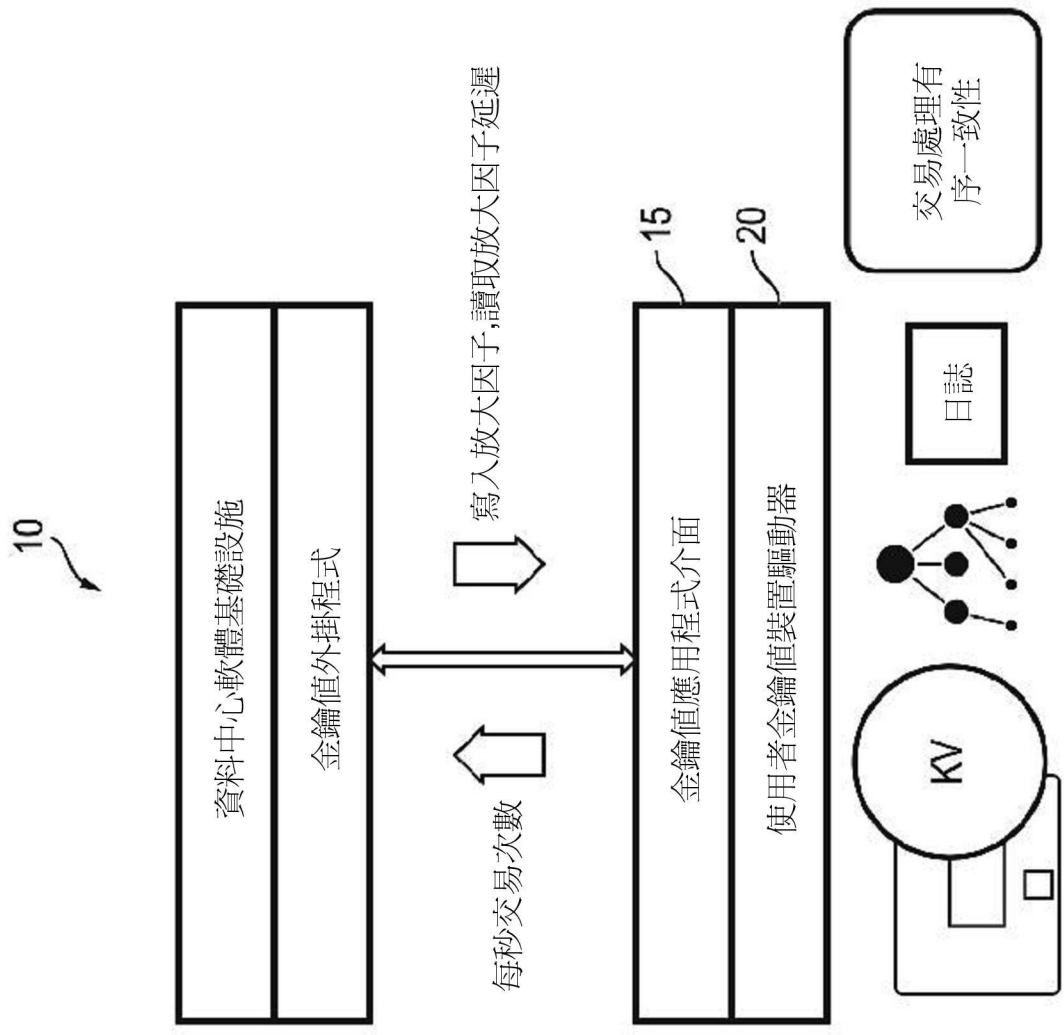
由所述虛擬裝置層接收來自所述記憶體裝置的回應，其中

當所述物件是大的或中等時，由所述虛擬裝置層接收資料塊及同位塊以利用糾刪編碼重建所述物件，且

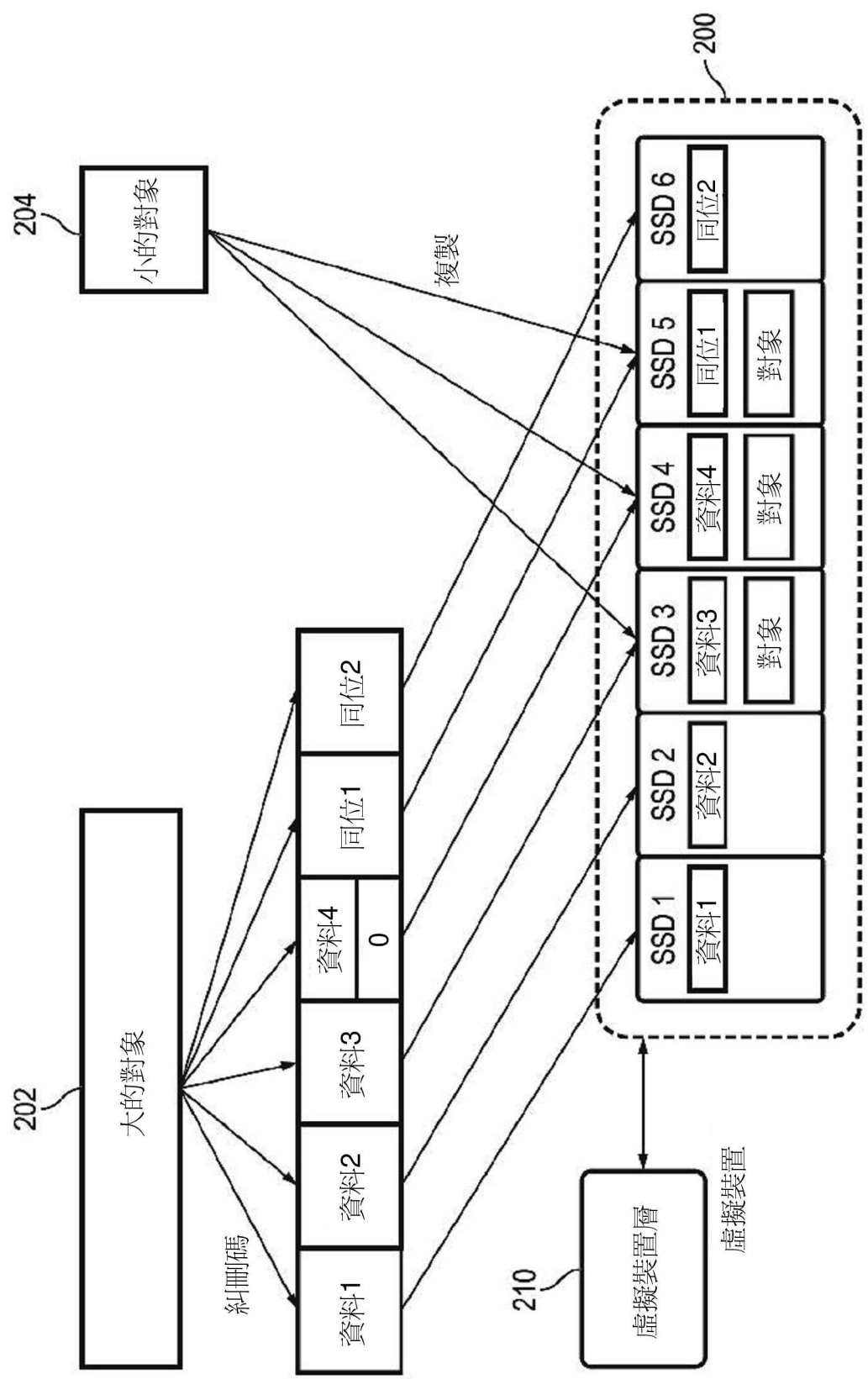
當所述物件是小的時，所接收的所述回應是所述物件或是所述物件的複製物。

【第20項】 如申請專利範圍第19項所述的方法，其中所述金鑰包括用於從所述多個記憶體裝置中確定開始裝置或主裝置的散列值。

【發明圖式】

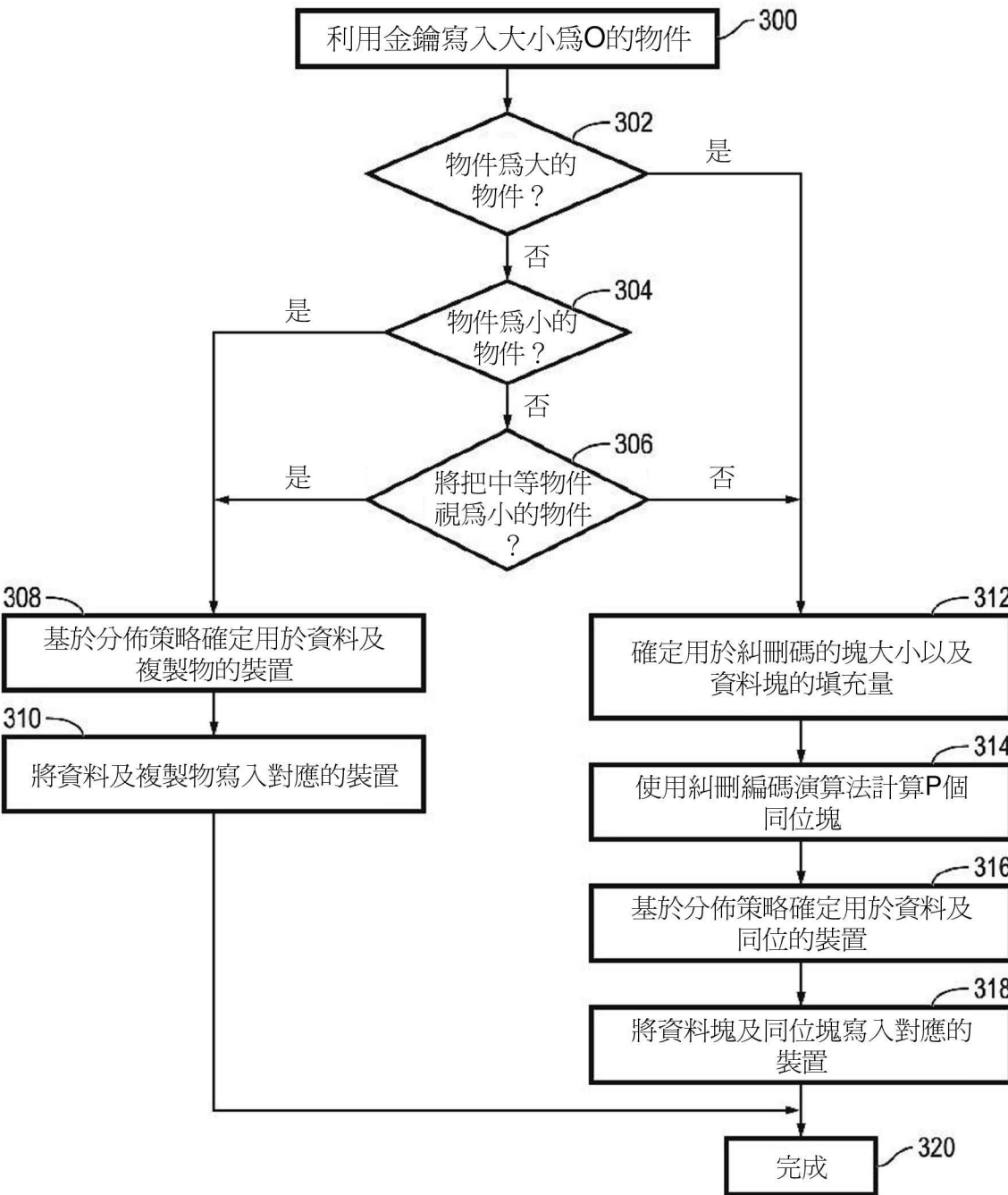


【圖1】

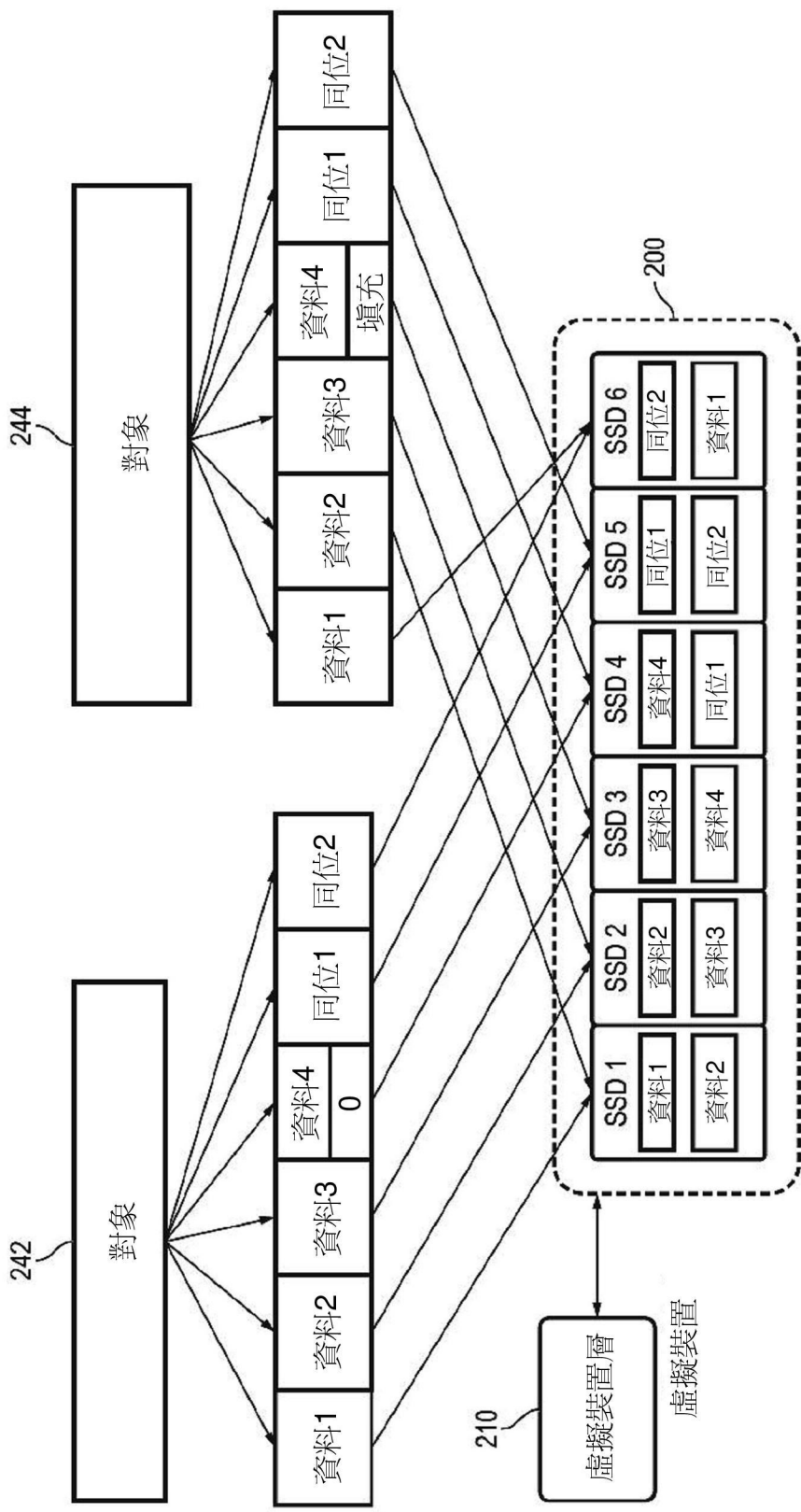


總共S個資料裝置以及P個同位裝置。同位裝置可旋轉。物件的開始裝置可由金鑰的散列值確定

【圖2】

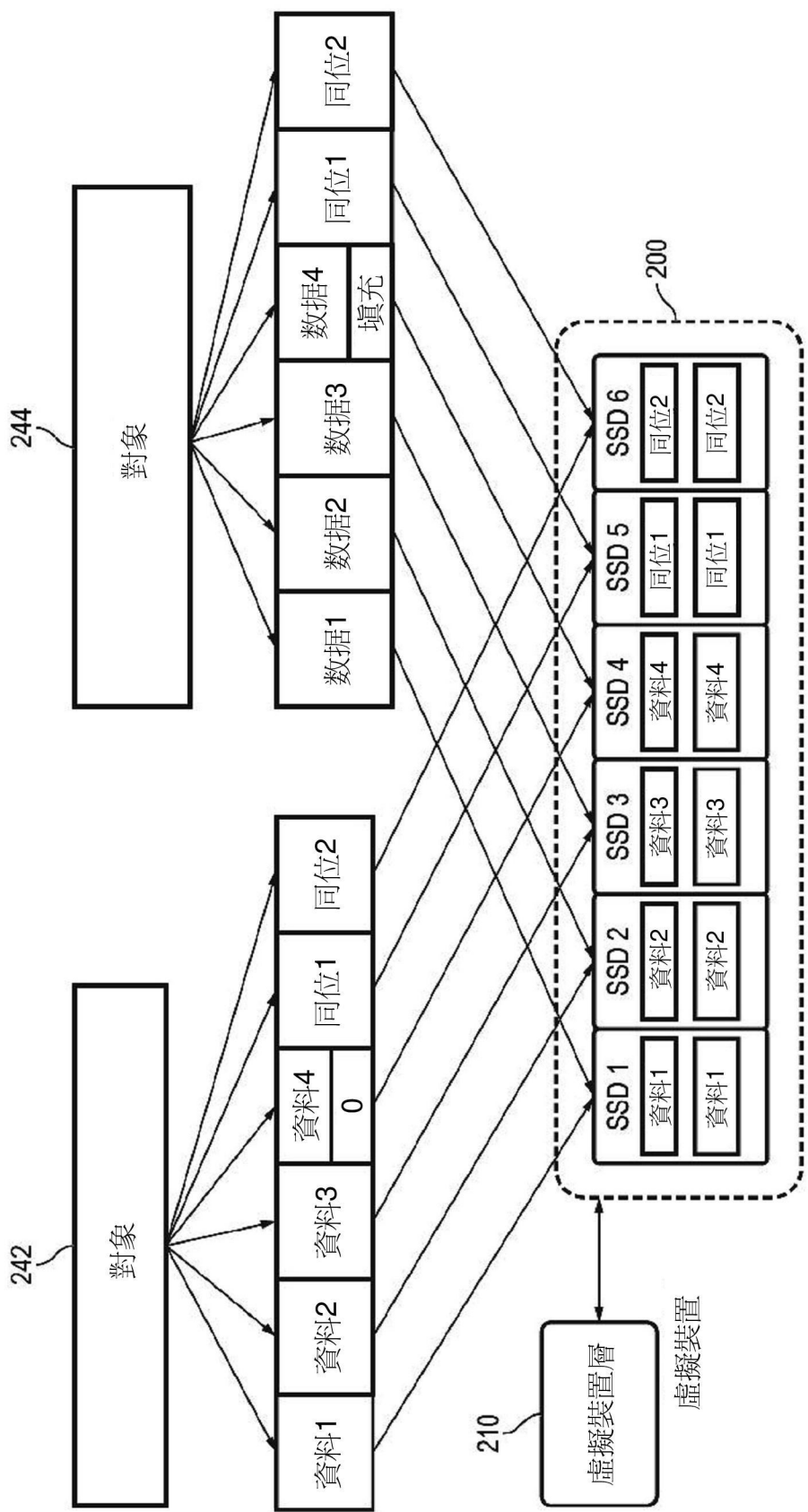


【圖3】



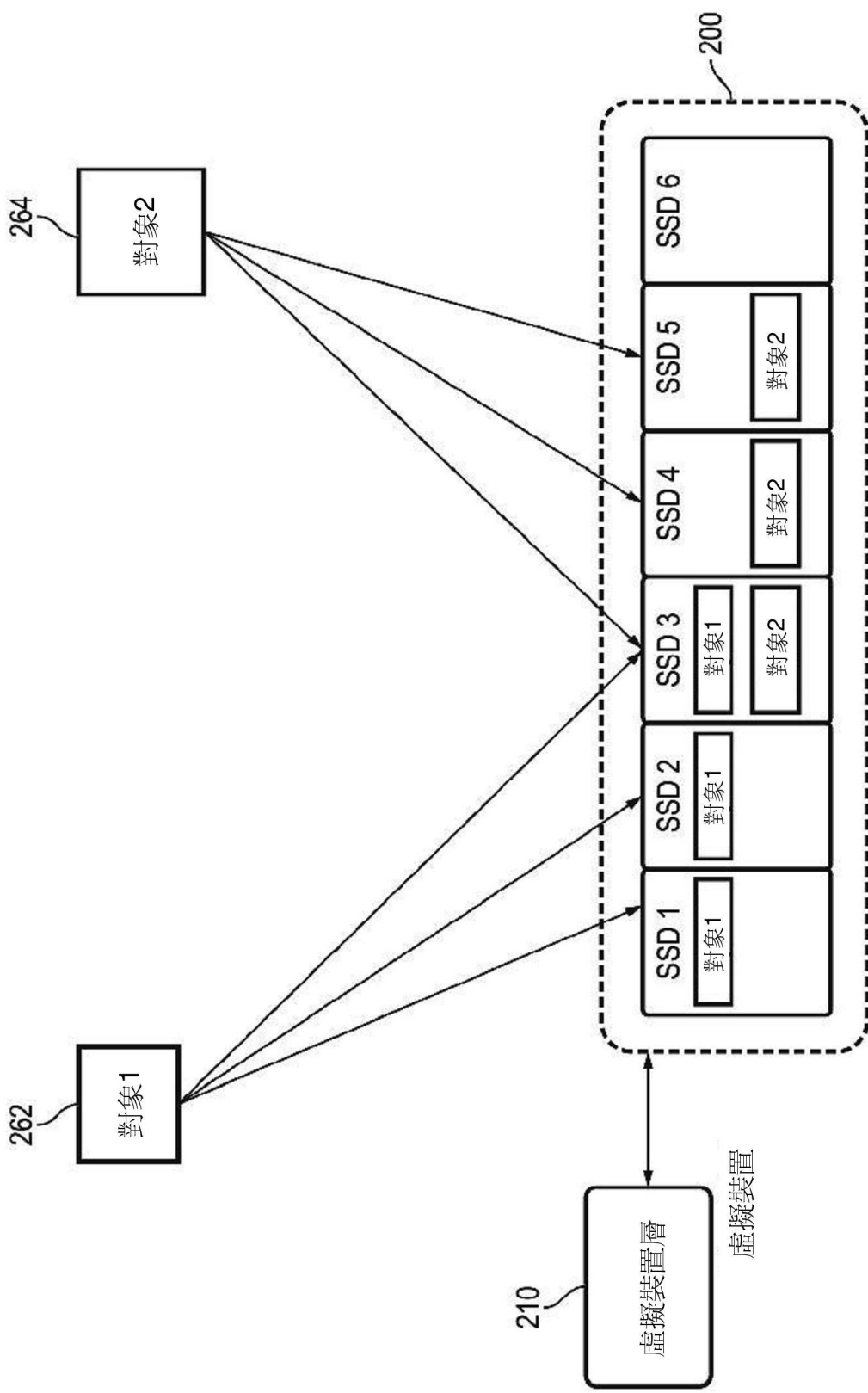
•總共S個資料裝置以及P個同位裝置。同位裝置可旋轉。物件的開始裝置可由金鑰的散列值確定

【圖4】



總共S個資料裝置以及P個同位裝置。同位裝置固定。物件的開始裝置可由金鑰的散列值確定

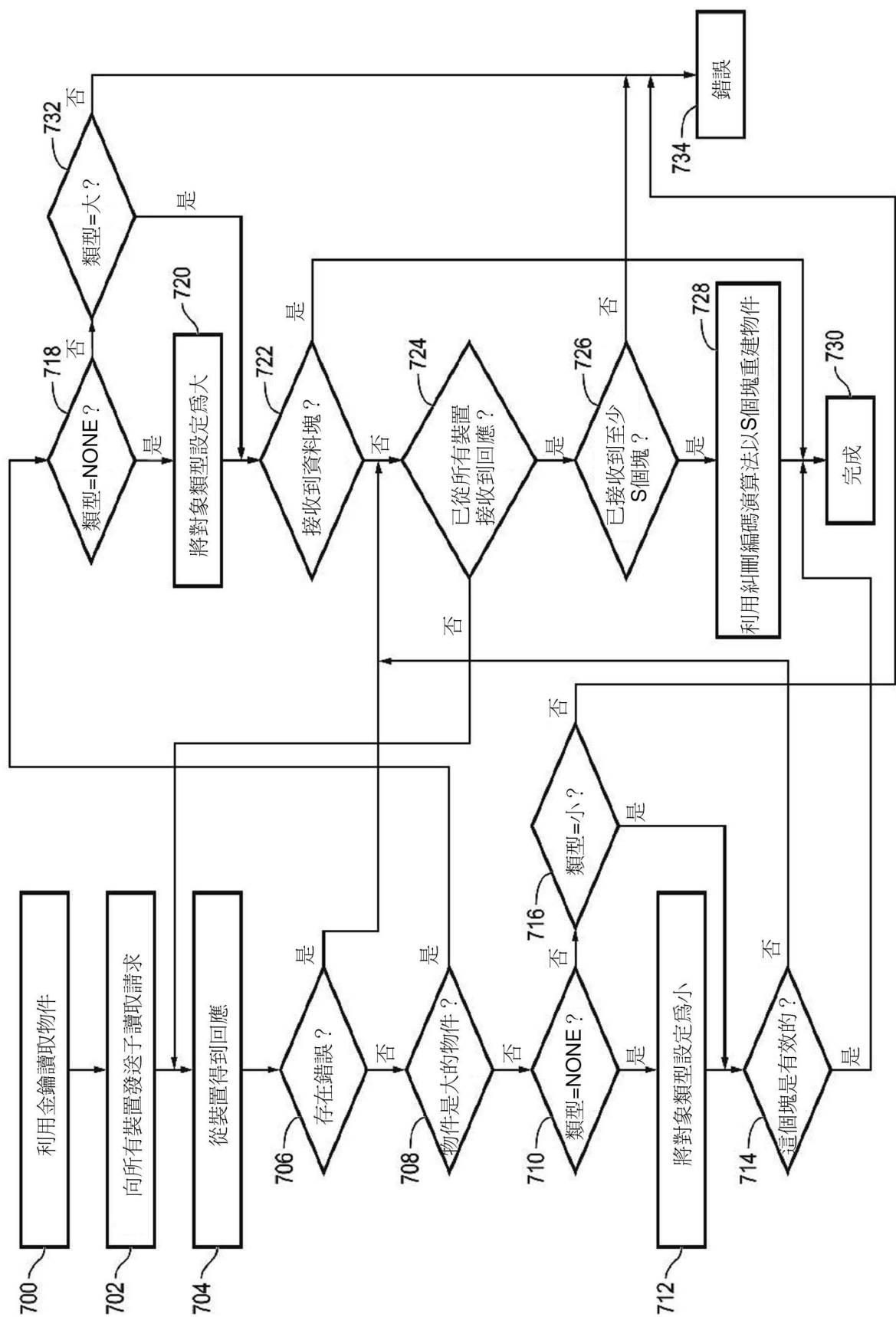
【圖5】



總共S個資料裝置以及P個同位裝置（目前沒有物件使用EC）。物件的開始裝置可由金鑰的散列值確定

【圖6】





【圖7】