I336836

公告本

發明專利說明書

（本說明書格式、順序及粗體字，請勿任意更動，※記號部分請勿填寫）

※ 申請案號：94114079

※ 申請日期：94年5月2日　　※IPC 分類：G06F 12/00 (2006.01)

一、發明名稱：（中文/英文）

擴展跨越記憶體描述符以描述另一分區記憶體之系統與方法

SYSTEM AND METHOD FOR EXTENDING THE CROSS-MEMORY DESCRIPTOR TO DESCRIBE ANOTHER PARTITION'S MEMORY

二、申請人：（共1人）

姓名或名稱：（中文/英文）

萬國商業機器公司

INTERNATIONAL BUSINESS MACHINES CORPORATION

代表人：（中文/英文）傑羅　羅森梭/ROSENTHAL, GERALD

住居所或營業所地址：（中文/英文）

美國紐約州 10504 亞芒克市新奧爾察德路

New Orchard Road, Armonk, NY 10504, U.S.A.

國　籍：（中文/英文）　美國/US

三、發明人：（共2人）

姓　名：（中文/英文）

1. 威斯爾　齊特瑞賈　艾斯洛/ ASLOT, VISHAL CHITTRANJAN

2. 布魯斯　G　蒙雷/ MEALEY, BRUCE G.

國　籍：（中文/英文）

1.－2.　均為美國/ US

四、聲明事項：

☐ 主張專利法第二十二條第二項☐第一款或☐第二款規定之事實，其事實發生日期為：　年　月　日。

☑ 申請前已向下列國家（地區）申請專利：

【格式請依：受理國家（地區）、申請日、申請案號　順序註記】

☑ 有主張專利法第二十七條第一項國際優先權：

美國；西元 2004 年 5 月 27 日；10/855,726

☐ 無主張專利法第二十七條第一項國際優先權：

☐ 主張專利法第二十九條第一項國內優先權：

【格式請依：申請日、申請案號　順序註記】

☐ 主張專利法第三十條生物材料：

☐ 須寄存生物材料者：

國內生物材料　【格式請依：寄存機構、日期、號碼　順序註記】

國外生物材料　【格式請依：寄存國家、機構、日期、號碼　順序註記】

☐ 不須寄存生物材料者：

所屬技術領域中具有通常知識者易於獲得時，不須寄存。

九、發明說明：

【發明所屬之技術領域】

本發明大致上針對一種改善的資料處理系統。特別是，本發明係針對一系統以及用來擴展該跨越記憶體描述符(cross-memory descriptor)的方法，該方法可被用來描述一個被邏輯分區之計算環境中另一分區之記憶體。

【先前技術】

高階交互執行(Advanced Interactive Executive, AIX)作業系統支持萬國商業機器公司(International Business Machine Corporation, IBM Corporation) 中 pSeries 機器的邏輯分區(logical partitioning, LPAR)。LPARs 允許一單處理器集團(single processor complex)上多樣的系統影像之運轉。每個這樣的影像有 CPUs(專用的或共用的)、核心存儲器、擴充存儲器、以及通道等之完全的編制。由於邏輯分區，各個分區之間有一清楚的資源隔離，因此一分區不會造成任何其他分區的系統環境不穩定。IBM 公司的機器中，能使各分區資源清楚的分隔之工具是系統超管理程式(hypervisor)。

自從各個分區有自己的一組隔離的資源，各個分區有對其他分區隱瞞的自己"本地的(local)"記憶體。直到虛擬輸入/輸出(virtual input/output, VIO)的出現，沒有合理的理由讓一分區去直接地存取其他分區的記憶體。

在分區之間，VIO(虛擬輸入/輸出)點對點直接記憶

體存取(Direct Memory Access, DMA)運作以及記憶體複製運作之觀念已被提出。DMA 運作容許讓一記憶體到另一記憶體直接地轉移資料，而不用使用中央處理單元(central processing unit, CPU)去執行該資料轉移。若需使用 CPU 來轉移資料，DMA 運作容許更快速的資料轉移。記憶體複製運作容許記憶體的部分從一記憶體被直接地複製到另一記憶體。

因此，VIO 允許共用實體的資源，例如分區之間的一輸入/輸出轉接器(I/O adapter)等等。由於一資源只可屬於一分區(稱為一"伺服端"分區)，其他分區為了使用該資源必須經過該伺服端分區(其他分區被稱為"客戶端"分區)。

至於該共用的資源為一 I/O 轉接器時，為了使用一伺服端分區的 I/O 轉接器，該伺服端分區需要從 I/O 轉接器到客戶端的記憶體建立一 DMA 運作。然而，為了建立該 DMA 運作，該伺服端分區需要去了解有關客戶端分區記憶體結構的資訊。此外，該伺服端分區需要能夠傳送有關該客戶端分區記憶體結構的資訊往下至自己的 I/O 堆疊，亦即所規定的軟體層階層，供一資料在一可用的形式中執行一 I/O 運作(例如邏輯磁碟管理[Logical Volume Manager, LVM]、磁頭驅動程式[head driver]、轉接器驅動程式、匯流排驅動程式、核心等等)，而被處理。因此，擁有一系統和方法以描述一伺服端分區環境中的一客戶端分區記憶體是有助益的，以此方法則該描述可輕易地往下傳送到該伺服端分區的 I/O 堆疊。

【發明內容】

本發明提供一系統和方法以擴展跨越記憶體描述符在一伺服端分區環境的用處，像是可用於描述另一分區的記憶體，例如一客戶端分區記憶體(在此後則參照為一"遠端"記憶體)。由於此系統和方法，當在一被邏輯地分區的計算系統中之一作業系統被初始化，例如該邏輯分區的開機期間，作業系統核心服務被召喚以執行開機運作，像是載入裝置的驅動程式以及類似者。作為這些作業系統核心服務的部分，一核心服務被召喚，附加其他邏輯分區的遠端記憶體到此作業系統，以致遠端記憶體複製以及 DMA 運作可與該其他邏輯分區的遠端記憶體被執行。

此附加的核心服務檢查該計算系統的一裝置樹，該裝置樹可由開放韌體或類似者所產生，並且該附加的核心服務被保持在伺服端分區的記憶體中，以致可被在該伺服端分區中的該作業系統存取。每個邏輯分區中有一裝置樹實例。裝置樹包含在該分區中供資源使用的節點。這些資源包含供各分區的本地記憶體、PCI/ISA I/O 槽、處理器、支援的韌體呼叫等等。

典型的實施例中，當 VIO 在計算環境中被支援，伺服端的裝置樹中，該/vdevice 節點或其子節點包含客戶端分區本地記憶體的描述。該描述包含客戶端本地記憶體的一邏輯識別符、該本地記憶體的一起始位址、以及該本地記憶體的一長度。從此資訊可知，供各邏輯分區的各本地記憶體之一擴展的跨越記憶體描述符可被產生。

這些擴展的跨越記憶體描述符則被目前分區的作業系統保持，以供執行遠端記憶體複製以及 DMA 運作之使用。

當存取一遠端記憶體是執行一運作之所需，則伺服端分區之作業系統使用該擴展的跨越記憶體描述符，其描述該客戶端分區的"遠端"記憶體，而此即為該遠端記憶體運作的主題。本發明的此擴展的跨越記憶體描述符是現存的跨越記憶體描述符之一延伸，現存的跨越記憶體描述符典型為一資料結構，其只用於描述之一分區的本地記憶體，其中該跨越記憶體描述符被使用於該分區中。

本發明中該跨越記憶體描述符的欄位被擴展，以致他們被用來儲存有關其他分區本地記憶體之資訊，即一"遠端"記憶體。特別是，該擴展的跨越記憶體描述符之欄位包含位址空間識別符、該遠端記憶體的大小、一識別符以識別是哪個遠端記憶體(在許多客戶端分區的情形下)、以及在該遠端記憶體內的一起始位址。當該跨越記憶體描述符被用來代表一"遠端"記憶體，則該位址空間識別符識別該跨越記憶體描述符為一遠端記憶體描述符。如此，I/O 堆疊內的任何介面，或檢查跨越記憶體描述符的記憶體管理服務則可視該位址空間識別符，而識別此為一遠端記憶體描述。因此，該介面可採取適當的動作以建立一 DMA 運作去存取該"遠端"記憶體。

本發明的這些以及其他的特徵和優點將被描述於接下來的較佳實施例之詳細說明中，或對熟此技藝者視為

明顯的。

【實施方式】

本發明提供一機制，用來擴展一第一分區環境中一跨越記憶體描述符，以致其描述另一分區的一本地記憶體，為了有助於需要直接記憶體存取的運作或遠端記憶體複製的運作。以此方式，本發明在一計算裝置中被較佳地實施，而其中之資源被用一邏輯分區機制所分區。

關於圖示，特別是關於圖 1，在一資料處理系統的方塊圖中，揭露本發明的實施方式。資料處理系統 100 可為一對稱多處理器(symmetric multiprocessor, SMP)系統，包含複數個處理器 101、102、103、以及 104 連接到系統匯流排 106。舉例來說，資料處理系統 100 可為一 IBM eServer，即為紐約州亞芒克市萬國商業機器公司的一產品，在一網路中以一伺服端方式實施。供選擇地，一單處理器系統可被使用。也連接到系統匯流排 106 的是記憶體控制器/高速快取記憶體(cache)108，其提供一介面給複數個本地記憶體 160-163。I/O 匯流排橋 110 連接到系統匯流排 106 並且提供一介面給 I/O 匯流排 112。記憶體控制器/高速快取記憶體 108 以及 I/O 匯流排橋 110 可如上述揭露以整合。

資料處理系統 100 是一邏輯分區(LPAR)的資料處理系統。由此，資料處理系統 100 可具有多異質作業系統(multiple heterogeneous operating systems)(或一單作業系統之多個的副本)同步進行。這些多作業系統的每個可具

有任何數量的軟體程式在其中執行。資料處理系統 100 被邏輯分區以至於該等不同的 PCI I/O 轉接器 120-121、128-129、以及 136、圖形轉接器 148、以及硬碟轉接器 149 被指派到不同的邏輯分區。此實施例中，當硬碟轉接器 149 提供一連接去控制硬碟 150 時，圖形轉接器 148 提供一連接給一顯示裝置(未圖示)。

因此，舉例來說，假設資料處理系統 100 被分隔為三個邏輯分區，P1、P2、以及 P3。PCI I/O 轉接器 120-121、128-129、136 的每個、圖形轉接器 148、硬碟轉接器 149、主處理器 101-104 的每個、以及本地記憶體 160-163 的記憶體被指派到這三區的各區。這些例子中，記憶體 160-163 可用雙直列記憶體模組(dual in-line memory modules, DIMMs)的形式。DIMMs 通常不會以單一 DIMM 為基礎指派到各分區。反而是，一分區將得到被該平台所見之全部記憶體的一部分。例如，處理器 101、本地記憶體 160-163 的一部分記憶體、以及 I/O 轉接器 120、128、以及 129 可被指派到邏輯分區 P1；處理器 102-103，本地記憶體 160-163 的一部分記憶體、以及 PCI I/O 轉接器 121 和 136 可被指派到邏輯分區 P2；以及處理器 104、本地記憶體 160-163 的一部分記憶體、圖形轉接器 148、以及硬碟轉接器 149 可被指派到邏輯分區 P3。

執行於資料處理系統 100 內的各個作業系統被指派到不同的邏輯分區。由此，執行於資料處理系統 100 內的各個作業系統只可存取在其邏輯分區內的那些 I/O 單元。因此，舉例來說，高階交互執行(Advanced Interactive

Executive, AIX)作業系統的一實例可在邏輯分區 P1 內被執行，AIX 作業系統的一第二實例(圖像) 可在邏輯分區 P2 內被執行，以及一 Linux 或 OS/400 作業系統可在邏輯分區 P3 內被執行。

週邊組件互連(peripheral component interconnect, PCI)主橋 114 連接到 I/O 匯流排 112，並提供一介面給 PCI 本地匯流排 115。一數量的 PCI I/O 轉接器 120-121 可經由 PCI-to-PCI 橋 116、PCI 匯流排 118、PCI 匯流排 119、I/O 槽 170、以及 I/O 槽 171 連接到 PCI 匯流排 115。PCI-to-PCI 橋 116 提供一介面給 PCI 匯流排 118 和 PCI 匯流排 119。PCI I/O 轉接器 120 和 121 被分別地置入 I/O 槽 170 和 171。典型的 PCI 匯流排施行將會支持四到八個之間的 I/O 轉接器(即供給增加到電腦上可以增加其性能的組件的連接器之擴充槽)。各個 PCI I/O 轉接器 120-121 提供一介面於資料處理系統 100 和 I/O 裝置之間，該介面例如為其他網路電腦，是為客戶端到資料處理系統 100。

一附加的 PCI 主橋提供一介面給一附加的 PCI 匯流排 123。PCI 匯流排 123 連接到複數個 PCI I/O 轉接器 128-129。PCI I/O 轉接器 128-129 可經由 PCI-to-PCI 橋 124、PCI 匯流排 126、PCI 匯流排 127、I/O 槽 172、以及 I/O 槽 173 連接到 PCI 匯流排 123。PCI-to-PCI 橋 124 提供一介面給 PCI 匯流排 126 和 PCI 匯流排 127。PCI I/O 轉接器 128 和 129 被分別地置入 I/O 槽 172 和 173。以此方式，附加的 I/O 裝置，像是例如數據機或網路轉接器

可經由各個 PCI I/O 轉接器 128-129 被支持。以此方式，資料處理系統 100 允許與多網路電腦的連接。

被插進 I/O 槽 174 的一記憶體映射的圖形轉接器 148 可經由 PCI 匯流排 144、PCI-to-PCI 橋 142、PCI 匯流排 141、以及 PCI 主橋 140 連接到 I/O 匯流排 112。硬碟轉接器 149 可被置入連接到 PCI 匯流排 145 的 I/O 槽 175。照順序地，此 PCI 匯流排 145 連接到 PCI-to-PCI 橋 142，其經由 PCI 匯流排 141 連接到 PCI 主橋 140。

一 PCI 主橋 130 提供一介面給一 PCI 匯流排 131 去連接到 I/O 匯流排 112。PCI I/O 轉接器 136 連接到 I/O 槽 176，其經由 PCI 匯流排 133 連接到 PCI-to-PCI 橋 132。PCI-to-PCI 橋 132 連接到 PCI 匯流排 131。此匯流排 131 也將 PCI 主橋 130 連接到該服務處理器信箱介面和 ISA 匯流排存取口之邏輯 194(service processor mailbox interface and ISA bus access pass-through logic) 以及 PCI-to-PCI 橋 132。服務處理器信箱介面和 ISA 匯流排存取口之邏輯 194 發送 PCI 預定的存取到 PCI/ISA 橋 193。NVRAM 儲存 192 連接到 ISA 匯流排 196。服務處理器 135 經由其本地 PCI 匯流排 195 連接到服務處理器信箱介面和 ISA 匯流排存取口之邏輯 194。服務處理器 135 也經由複數個 JTAG/I^2C 匯流排 134 連接到處理器 101-104。JTAG/I^2C 匯流排 134 是 JTAG/scan 匯流排(參考 IEEE 1149.1)以及 Philips I^2C 匯流排的結合。然而，供選擇地，JTAG/I^2C 匯流排 134 可被 Philips I^2C 匯流排取代或是被 JTAG/scan 匯流排取代。主處理器 101、102、

103、以及 104 的所有 SP-ATNN 訊號都相連一起到該服務處理器的一中斷的輸入訊號。服務處理器 135 有自己的本地記憶體 191，並有通道存取該硬體 OP-控制板(hardware OP-panel)190。

當資料處理系統 100 最初被啟動時，服務處理器 135 使用 JTAG/I^2C 匯流排 134 去查問該系統(主)處理器 101-104、記憶體控制器/高速快取記憶體(cache)108、以及 I/O 橋 110。這步驟的完成時，服務處理器 135 則對資料處理系統 100 有一清單目錄以及拓撲的理解。服務處理器 135 也在查問該主處理器 101-104、記憶體控制器/高速快取記憶體(cache)108、以及 I/O 橋所發現的所有元件上，執行內建自我測試(Built-In-Self-Tests, BISTs)、基本保證測試(Basic Assurance Tests, BATs)、以及記憶體測試。在 BISTs、BATs、以及記憶體測試期間所偵測到的任何失敗之錯誤訊息，被服務處理器 135 所集結並且報告。

若拿掉在 BISTs、BATs、以及記憶體測試中所發現有缺點的元件後，系統資源的一有意義的/有效的結構仍然合適時，則資料處理系統 100 被允許去進行下載可執行的程式到本地(主)記憶體 160-163。接下來，服務處理器 135 釋放主處理器 101-104 以執行該程式的下載到本地記憶體 160-163。當主處理器 101-104 從資料處理系統 100 中個別的作業系統執行程式時，服務處理器 135 則進入監控和報告錯誤的模式。由服務處理器 135 監控的項目類型包含例如：冷卻風扇(cooling fan)的速度和運作、

熱感應器(thermal sensor)、電力提供調節器(power supply regulators)、以及由處理器 101-104、本地記憶體 160-163、以及 I/O 橋 110 所報告之可修復與不可修復的錯誤。

服務處理器 135 負責保留和報告關於資料處理系統 100 中所有監控項目之錯誤資訊。服務處理器 135 也基於錯誤的類型和被定義的門檻(thresholds)採取行動。舉例來說，服務處理器 135 可注意一處理器的高速快取記憶體上過度可修復的錯誤，並決定此為一預兆的確實失敗。基於此決斷力，服務處理器 135 可在當時的運作期間以及未來的起始程式載入(Initial Program Loads, IPLs)中標明供規劃的資源。IPLs 有時也意指為一"啟動(boot)"或"自我啟動(bootstrap)"。

資料處理系統 100 可使用不同的商業可用的電腦系統來實施。例如，資料處理系統 100 可使用萬國商業機器公司所發行的 IBM eServer iSeries Model 840 系統來實施。此一系統可支持使用一 OS/400 作業系統之邏輯分區，此作業系統也是由萬國商業機器公司所發行。

一般熟此技藝者將可察知圖 1 所描述之硬體可加以變化。舉例來說，其他週邊的裝置，像是光碟機以及類似者可用來附加或取代該描述的硬體。此描述的例子並不意指關於本發明建構的限制。

現在參照圖 2，描述本發明可被實施的一例示的邏輯分區平台之方塊圖。在邏輯分區平台 200 中的硬體可

被實施為，例如，圖 1 中的資料處理系統。邏輯分區平台 200 包含分區的硬體 230、作業系統 202、204、206、208、以及分區管理韌體 (partition management firmware)210。作業系統 202、204、206、以及 208 可為在邏輯分區平台 200 上同步進行的一單作業系統之多個副本或多異質作業系統(multiple heterogeneous operating systems)。這些作業系統可使用 OS/400 來實施，這些作業系統被設計為分區管理韌體之介面；像是一系統超管理程式(hypervisor)。OS/400 只是被用來當做這些例示的實施例中的一個例子。當然，其他類型的作業系統，像是 AIX 和 linux，則可視特定的實施以使用。作業系統 202、204、206、以及 208 位於分區 203、205、207、以及 209 中。系統超管理程式(hypervisor)軟體是可用來實施分區管理韌體 210 的一軟體例子，由萬國商業機器公司所發行。韌體是"軟體"儲存在一記憶體晶片，而該晶片不需電力去保持其內容，像是例如唯讀記憶體(read-only memory, ROM)、可程式唯讀記憶體(programmable ROM, PROM)、可抹除可程式唯讀記憶體(erasable programmable ROM, EPROM)、電子可抹除可程式唯讀記憶體(electrically erasable programmable ROM, EEPROM)、非揮發性記憶體(nonvolatile random access memory, nonvolatile RAM)。

另外，這些分區也包含分區韌體 211、213、215、以及 217。分區韌體 211、213、215、以及 217 可使用起始自我啟動碼(initial boot strap code)、IEEE-1275 標準開放韌體、以及執行時抽象軟體(runtime abstraction software,

RTAS)來實施，由萬國商業機器公司所發行。當分區203、205、207、以及209被舉例說明時，自我啟動碼(boot strap code)的一副本被平台韌體210下載到分區203、205、207、以及209。之後，控制被轉移到自我啟動碼，然後自我啟動碼下載該開放韌體和RTAS。該等被結合或被分派到各分區的處理器之後，被派遣到分區的記憶體去執行分區韌體。

分區硬體230包含複數個處理器232-238、複數個系統記憶體單元240-246、複數個輸入/輸出(I/O)轉接器248-262、以及一儲存單元270。處理器232-238、記憶體單元240-246、NVRAM儲存298、以及I/O轉接器248-262當中的每一個可被指派到邏輯分區平台200中多數個分區之一，當中每一個分區都對應至作業系統202、204、206、及208的其中之一。

分區管理韌體210執行一些功能和服務，以供分區203、205、207、以及209去產生並加強邏輯分區平台200的分區動作。分區管理韌體210是一韌體實施的虛擬機器，其相同於其下的硬體。由此，分區管理韌體210允許由虛擬化邏輯分區平台200的所有硬體資源，而使獨立OS圖像202、204、206、以及208同步的執行。

服務處理器290可用來提供不同的服務，像是分區中平台錯誤的處理。這些服務也表現如同一服務代理人，去回報錯誤訊息給供應商，像是萬國商業機器公司。不同分區的作業可經由一硬體管理操作台來控制，像是

硬體管理操作台 280。硬體管理操作台 280 是一個別的資料處理系統，有別於一系統管理者可執行不同的功能，其包含對不同分區之資源的重新分配。

在這樣一個邏輯分區環境中，不論何時當資料在核心與一個非目前程序所在的位址空間之間被移動時，一跨越記憶體核心服務被利用去執行此資料之移動。在一位址空間的一區域內之一資料區由呼叫 xmattach 核心服務而被附加，該 xmattach 核心服務為跨越記憶體運作，而附加到一使用者緩衝器。當此 xmattach 核心服務被召喚時，此 xmattach 核心服務產生一跨越記憶體描述符。之後，其他跨越記憶體核心服務可被利用來從核心移動或複製此資料，到一非目前程序所在的位址空間。舉例來說，xmemin 核心服務經由從特定位址空間複製資料到核心全域的記憶體，以執行一跨越記憶體的行動。xmemout 核心服務經由從核心全域的記憶體複製資料到特定位址空間，以執行一跨越記憶體的行動。Xmemdma 核心服務準備一頁的記憶體供 DMA I/O，或在 DMA I/O 完成後處理一頁。

已知的計算系統中，該跨越記憶體描述符是用來描述本地記憶體的一資料結構。該本地記憶體跨越記憶體描述符由作業系統的虛擬記憶體管理(Virtual Memory Management, VMM)元件維持的資訊所產生。

依據本發明的一擴展的跨越記憶體描述符，由基於一裝置樹的核心服務所產生，此裝置樹由位於啟動定時

器(boot timer)的作業系統所產生和剖析。此裝置樹相似於該開放韌體裝置樹，其為一有階層的資料結構，描述該系統硬體以及使用者結構的選擇。此開放韌體裝置樹也包含硬體驅動程式和支持的小程式(support routines)供這些驅動程式使用。以下是一開放韌體裝置樹的一例子：

```
root        /
ff8885f8 /rtas
ff866bf4 /rom@ff000000
ff8627c0 /flash@fff00000
ff8513d0 cpus
ff88a1c8     /PowerPC, 604ev@0
ff88a788         /12-cache
ff84d8e0 /pci
ff89552c    /ethernet@4
ff8952e0    /display
ff88cf44    /mac-io@2
ff893e68        /misc@0
ff894688            /iic
ff893d7c        /via@16000
ff8939fc        /escc@13000
ff893c2c            /ch-b@13000
ff893af4            /ch-a@13020
ff88fd08        /scsi@10000
ff892e1c            /tape
```

ff8924d0 /disk
ff88f944 /escc-legacy@12000
ff88fba8 /ch-b@12000
ff88fa60 /ch-a@12002
ff88d78c /adb@11000
ff88f364 /mouse@3
ff88e6dc /keyboard@2
ff88d638 /open-pic@40000
ff88aad8 /ide@1,1
ff88c504 /disk
ff85a534 /isa@1
ff864208 /sound@i534
ff8640e8 /midi@i330
ff863ff8 /game@i200
ff863368 /gpio@i800
ff863008 /nvram@me0000
ff862aa4 /rtc@i70
ff85f644 /8042@i60
ff8618b0 /mouse@aux
ff860260 /keyboard@
ff85d804 /floppy@i3f0
ff85d3b4 /parallel@i3bc
ff85c704 /serial@i2f8
ff85b9f4 /serial@i3f8
ff85b490 /timer@i40

ff85b01c /interrupt-controller@i20

ff85ae08 /dma-controller@i00

ff84a650 /mmu

ff83f2e4 /memory@0

對應本發明，一相似的裝置樹代表被用來獲得供產生該擴展的跨越記憶體描述符之資訊。此裝置樹是物件導向，如此則此裝置樹的各個項目有可被檢查之相關的性質。在此裝置樹中一節點的一例子，其中關於遠端記憶體的資訊可被呈現給該作業系統，如以下說明：

/devicenode

...

remote-memory-info 10000000 00200000 00040000

20000000 003D0000 00020000

...

...

其中"remote-memory-info"是該節點"/devicenode"的一性質。客戶端-伺服端虛擬 I/O 模型(client-server virtual I/O model)的一典型的實施例中，該節點"/devicenode"會代表該伺服端裝置的節點。如此後將討論的，此節點可被剖析，並且從剖析此節點所獲得之資訊可被用來產生與本發明一致的一擴展的跨越記憶體描述符。由此，對一個第一客戶來說，該"10000000"項目是此記憶體的邏輯識別証，該"00200000"項目是此記憶體的起始位址，

以及該"00040000"項目是此記憶體位元的大小。同樣地，對一個第二客戶來說，"20000000"是此記憶體的邏輯識別証，"03D00000"是此記憶體中之起始位址，以及該"00020000"是此記憶體位元的大小。這三個一組(邏輯id、起始位址、大小)為該伺服端提供可被重複給各個客戶。

從這裝置樹資訊，此系統硬體和使用者組態選擇的詳細描述可被獲得。特別是，如以上所示，此裝置樹的記憶體節點包含指明該記憶體的一邏輯識別証之性質、此記憶體的起始位址、以及此記憶體之長度。此資訊可被該核心服務使用，為該計算系統的本地記憶體資源產生一跨越記憶體描述符。

跨越記憶體描述符被使用來當作一種方式，去從兩個不相關的背景存取記憶體，以至於不需要知道該問題中的記憶體(memory-in-question)的擁有者。舉例來說，當一中斷操作者需要傳遞資料到一程序，則其使用一跨越記憶體描述符，去獲得關於程序的記憶體空間之資訊，以複製該資料到此程序的記憶體空間。這是必要的，因為該中斷操作者不必然在一程序的前後背景中進行，而該程序持有其資料，且因此該操作者因此不知道該資料屬於哪個程序。該中斷操作者所知道的是記憶體的一描述，即該跨越記憶體描述符，而在此該資料應被複製。

另一個使用一本地記憶體跨越記憶體描述符的例子是磁碟 I/O。舉例來說，一使用者層次(user-level)應用程

式可請求一 IOCTL 去讀取關於一磁碟的資訊。此資訊會被儲存在該應用程式供給緩衝器(application supplied buffer)，其位於該使用者空間(user-space)中。當該磁碟驅動程式被呼叫時，該磁碟驅動程式請求 xmattach 去連結該應用程式的緩衝器。這些全部被該磁碟驅動程式的上半部所完成。

在未來的某一時點，當該 I/O 要求完成且資料變為可用的，該磁碟驅動程式的位於下半部的"iodone"小程式，則使用跨越記憶體描述符，去呼叫 xmout 去複製資料到該使用者空間。此跨越記憶體描述符被該磁碟驅動程式儲存，以至於其可被上、下半部存取。

該跨越記憶體描述符在作業系統初始化被該核心服務所產生，並且在計算裝置的 I/O 堆疊中，依需要被從介面到介面(或記憶體管理服務)間傳遞。這些介面和 I/O 堆疊的記憶體管理服務檢查該跨越記憶體描述符，以獲得執行跨越記憶體運作的必要資訊。

作業系統面臨一相似的問題，是關於另一分區的遠端記憶體，因為此遠端記憶體複製/DMA 運作不具有另一分區中擁有者裝置的任何知識。由此，本發明利用一擴展的跨越記憶體描述符形式，當執行一遠端記憶體複製/DMA 運作時的遠端記憶體去提供關於另一分區之資訊。

本地跨越記憶體描述符包括四個主要的欄位：一位

址空間識別符、程式段(segments)、一起始程式段識別符、以及程式段中一起始位址。該位址空間識別符提供一獨特的記憶體之識別符，其中資料被複製到該記憶體。程式段的數量提供被位址空間識別符識別的位址空間之程式段的一合計的數目。該起始程式段識別符識別在程式段總數內的一程式段，其中該資料被複製是在此被寫入。程式段中的該起始位址識別一位址，該位址在被起始程式段識別符識別的程式段中，其中該資料被複製是在此被寫入。

以下是一跨越記憶體資料結構的一例子，其中這些主要的欄位有被說明：

```
enum asid={LOCAL=0, REMOTE=1};
struct cross_memory {

    enum asid address_space_id;   /* LOCAL, REMOTE,
                                     etc. */
    int number_of_segments;   /* number of segments if
                                 address_space_id=LOCAL*/
    char vaddr;               /* starting address within
                                 the segment if
                                 address_space_id=LOCAL */
    uint flags;               /* any special attributes of the
                                 segment or address range
                                 */
};
```

```
#define remote_size (number_of_segments)    /*size of
                                    remote memory if
                                address_space_id=REMOTE */
#define remote_logical_id (segment_id)      /* remote
                                    memory's logical
                                    identifier */
#define remote_start_addr (vaddr)     /* starting address
                                    within
                                the remote memory */
```

由此，從上述欄位可清楚了解，該跨越記憶體描述符提供該記憶體的一描述，並具有明確性指出其可被用於執行跨越記憶體運作，而不需去知道一特別位址空間的擁有者，並且以此方式，為了存取該位址空間而傳遞該資料給該擁有者。本發明利用此已知資料結構去促進一不同的運作，其中關於資源所有權的資訊則不是完成此運作之所須。也就是說，本發明使用跨越記憶體描述符一擴展的形式，去促使遠端記憶體複製/DMA 運作從一分區到另一分區(其中該"遠端"記憶體是在不同於目前執行程序的分區中的一記憶體)。由於這樣的運作，一分區中的程序可存取另一分區的記憶體，如一虛擬輸入/輸出(virtual input/output, VIO)程序，而不用傳遞該資料到具有該遠端記憶體所有權的一特定的程序。結果，由於使用本發明的擴展的跨越記憶體描述符，而使遠端記憶體複製和 DMA 跨越分區變為可運作的。

由於本發明，該跨越記憶體描述符欄位被用來儲存

關於一遠端記憶體的資訊。該跨越記憶體描述符原先的四個欄位被超載去描述該遠端記憶體，而由此產生一擴展的跨越記憶體描述符，具有以下四個欄位：位址空間識別符、遠端記憶體的大小、識別該遠端記憶體的一識別符、以及該遠端記憶體內一起始位址。該擴展的跨越記憶體描述符之位址空間識別符相似於原先跨越記憶體的位址空間識別符，除外的是一預定值可被儲存在此欄位，以指明該擴展的跨越記憶體描述符對應一客戶端分區的一遠端記憶體。

該遠端記憶體的大小可由位元組或其他記憶體大小的單位被指定，並指定執行存取之遠端記憶體合計的大小。該遠端記憶體識別符識別一特定的客戶端分區之一特定的遠端記憶體。在此很重要的是，有複數個客戶端分區，且去指定出哪個遠端記憶體被存取是必要的。該遠端記憶體內的起始位址提供該遠端記憶體內，被遠端記憶體識別符指定之存取開始的位置。從這四個欄位，該遠端記憶體的一個完整的圖象則可被獲得，以便執行遠端記憶體複製/DMA 運作。

該擴展的跨越記憶體描述符可以與原先的跨越記憶體描述符同樣的方式，經由該 I/O 堆疊被從介面傳遞到介面。那些需要檢查擴展跨越記憶體描述符的介面和記憶體管理服務被擴大，以包含處理擴展的跨越記憶體描述符的程式。也就是說，那些介面和記憶體管理服務被擴大，以包含程式碼，識別當該位址空間識別符包含一預定值，而該預定值識別該擴展的跨越記憶體描述符為

描述一遠端記憶體供作存取要求。該介面和記憶體管理服務再來可執行適當的行動，基於遠端記憶體大小、遠端記憶體的身分、以及在遠端記憶體內由擴展的跨越記憶體描述符所提供的起始位址。舉例來說，d_map_page 和 d_map_list 介面使用遠端的跨越記憶體描述符，從一伺服端分區所擁有的一實體裝置到一遠端記憶體，去正確地建立 DMA。所以該遠端的跨越記憶體描述符被該實體裝置的驅動程式傳遞到 d_map_page/list 介面以設定 DMA。由伺服端分區的定義，該實體裝置屬於該伺服端分區。結果，遠端記憶體複製/DMA 運作可跨越分區被執行。

由超載該存在的跨越記憶體描述符以成為一擴展的跨越記憶體描述符，其具有與原先跨越記憶體描述符本質上相同的形式，但不同的資料儲存於跨越記憶體描述符的欄位內，不同的與此資料相關之內涵，本發明允許需要跨越記憶體描述符的許多介面和記憶體管理服務，但不檢查該跨越記憶體描述符之內容，以他們具有的去運作而沒有修改。也就是說，該擴展的跨越記憶體描述符允許該作業系統去維持與介面和記憶體管理服務二進位兼容性(binary compatibility)，其需要跨越記憶體描述符，但事實上並無真正檢查。舉例來說，大部分裝置驅動程式需要跨越記憶體描述符，但不檢查。他們僅僅傳遞該跨越記憶體描述符到該核心或該 PCI 匯流排驅動程式。因為本發明對該跨越記憶體描述符所作的改變，這些裝置驅動程式不需被重新編譯，並且他們可繼續如同往常之運作。

圖 3 是一例示圖，提示依照本發明一實施例說明一機制的例子，以產生一擴展的跨越記憶體描述符。如圖 3 所示，當邏輯分區計算系統中之一邏輯分區 310 的一作業系統 320 被初始化，如該邏輯分區 310 啟動期間，作業系統核心服務 325 被召喚，以執行啟動的運作，像是裝置驅動程式的載入以及類似者。如這些作業系統核心服務 325 的一部分，一核心服務被召喚，去連結其他邏輯分區 312-316 的遠端記憶體到此作業系統，以致於遠端記憶體複製和 DMA 運作可與其他邏輯分區的遠端記憶體被執行。本發明的一實施例中，連結遠端記憶體的該核心服務被稱之為 xmattach_remio，並且是已知 xmattach 跨越記憶體核心服務的一擴展的版本。該 xmattach_remio 連結遠端記憶體以供跨越記憶體運作，可被已知的 xmemout 和 xmemin 跨越記憶體核心服務以及 d_map_page 和 d_map_list 核心服務所執行。

該擴展的連結核心服務檢查該計算系統的一裝置樹 332-338，其可被維持在一系統超管理程式(hypervisor)330 內或類似者內。該系統超管理程式 330 是一設施，可在邏輯分區計算系統內提供和管理多數的虛擬機器。供各個邏輯分區的一個別的裝置樹 332-338 被維持在該系統超管理程式 330 中。此外，供一邏輯分區的該裝置樹 332-338 之一副本可被儲存在該邏輯分區的一本地記憶體中，以至於其可被該邏輯分區的作業系統核心所存取。該作業系統提供函數庫，供該核心的擴展(例如裝置驅動程式)以及使用者層碼可使用，以剖析該裝置樹 332-338。

該裝置樹 332-338 包含計算系統之各分區和該等裝置樹的資源的節點。這些資源包含供各分區的本地記憶體 340-360。該等裝置樹 332-338 的節點代表該分區的這些本地記憶體，包含這些本地記憶體的性質，其包含該本地記憶體的一邏輯識別符、該本地記憶體的一起始位址、以及該本地記憶體的一長度。從這個資訊可知，供各個邏輯分區(其為有關目前邏輯分區的"遠端"記憶體)的本地記憶體之大小、供各個邏輯分區的本地記憶體之一身分、以及供各個邏輯分區的本地記憶體內之一起始位址可被獲得。這資訊可被包裝進針對一擴展的跨越記憶體描述符 370-390 中供各邏輯分區的各本地記憶體。這些擴展的跨越記憶體描述符 370-390 之後被目前的邏輯分區 310 的作業系統 320 所保持，以供執行遠端記憶體複製和 DMA 運作到其他邏輯分區 312-316 之使用。

圖 4 是一例示圖，說明依據本發明一實施例之一機制的一例子，其供使用一擴展的跨越記憶體描述符，去執行需要遠端記憶體存取的一運作。如圖 4 所示，當一客戶端裝置 400 的一程序，例如一應用程式 410，產生一輸入/輸出(I/O)請求，例如經由一 read() 系統呼叫到該作業系統核心空間 420，該空間需要一遠端記憶體存取運作被執行，而該 I/O 請求被處理，通過該 I/O 堆疊 422 到一客戶端虛擬裝置驅動程式 424。該客戶端虛擬裝置驅動程式 424 遞送該請求到一伺服端 480 的對應，即伺服端虛擬驅動程式 430。

該伺服端虛擬驅動程式 430 已經產生一擴展的跨越

記憶體描述符 440 給記憶體 415 的一遠端分區，例如在啟動時間的一邏輯分區 2。舉例來說，該伺服器虛擬驅動程式 430 在啟動時，可呼叫本發明的該 xmattach_remio 核心服務，去產生一擴展的跨越記憶體描述符。該伺服端虛擬驅動程式 430 傳送與該擴展的跨越記憶體描述符 440 一起的該 I/O 請求，傳送到伺服端上該 I/O 堆疊 450。此 I/O 請求最後從該 I/O 堆疊 450 被傳送到實體裝置驅動程式 460，例如一 SCSI 轉接器驅動程式，實體裝置驅動程式 460 用其 I/O 轉接器 470 安排一 DMA 運作。當 DMA 被該 I/O 轉接器 470 完成，則該資料被寫入到該遠端記憶體 415。由此，如上述例子所說明，除了該資料本身，供該遠端直接記憶體發生所需的唯一其他必須要素是該遠端記憶體的描述，該描述經由本發明的擴展的跨越記憶體描述符所提供。

圖 5 和圖 6 是流程圖，說明依據本發明實施例產生和使用一擴展的跨越記憶體描述符之例式的過程。將被了解的是，流程圖的各個區塊以及流程圖中區塊的組合可被電腦程式指令所實施。這些電腦程式指令可被提供給一處理器或其他可程式資料處理裝置去產生一機器，以至於在處理器或其他可程式資料處理裝置上執行的指令，產生實施流程圖區塊中特定功能的方法。這些電腦程式指令也可被儲存在一電腦可讀取記憶體或儲存媒體內，該儲存媒體可指示一處理器或其他可程式資料處理裝置去以一特定方式運作，以至於儲存於電腦可讀取記憶體或儲存媒體內的指令產生一製造的物品，包含指令方法，其實施流程圖區塊中特定的功能。

相應地，流程圖式的區塊支持用來執行特定功能的方法之組合、用來執行特定功能的步驟之組合、以及用來執行特定方法的程式指令方法。也將被了解的是，流程圖的各個區塊以及區塊的組合，可被特殊目的之基於硬體的(hardware-based)電腦系統所實施，該電腦系統執行該特定功能或步驟，或可被特殊目的硬體以及電腦指令之組合所實施。

圖 5 是一流程圖，概述一例示的程序以產生與本發明的一實施例一致的一擴展的跨越記憶體描述符。如圖 5 所示，該運作以一作業系統的初始化開始(步驟 510)。該作業系統核心服務之後為該電腦系統剖析該裝置樹，找尋代表其他邏輯分區遠端記憶體的節點(步驟 520)。這些遠端記憶體節點的性質之後則被擷取(步驟 530)，並且被用來為其他分區的各個遠端記憶體決定一識別符、大小、以及起始位址(步驟 540)。為了各個其他邏輯分區的各個遠端記憶體，該識別符、大小、以及起始位址之後與一遠端識別符被包裝進一擴展的跨越記憶體描述符(步驟 550)。這些擴展的跨越記憶體描述符之後被儲存，以供後來的執行遠端記憶體存取的使用(步驟 560)。

圖 6 是一流程圖，概述一例示的程序以使用與本發明的一實施例一致的一擴展的跨越記憶體描述符。如圖 6 所示，該運作以接收一遠端記憶體存取所需的請求開始(步驟 610)。供遠端記憶體的該擴展的跨越記憶體描述符被該請求作為目標，而該請求之後被擷取(步驟 620)。該擴展的跨越記憶體描述符然後與該請求一起被傳遞到

I/O 堆疊(步驟 630)，該 I/O 堆疊相對照地處理該擴展的跨越記憶體描述符和該請求。該遠端記憶體存取之後經由 I/O 轉接器使用該擴展的跨越記憶體描述符而被執行(步驟 640)。

由此，本發明提供一機制以擴展該已知的跨越記憶體描述符，以至於其可描述一不同於一程序正運行的分區之分區的遠端記憶體。因為本發明所提供的擴展與已知的跨越記憶體描述符使用相同形式，但具有不同的資料和與資料相關的內涵，於計算裝置中表示的許多介面和記憶體管理服務不需被修改以操作該擴展的跨越記憶體描述符。其他檢查該跨越記憶體描述符的介面和記憶體管理服務可被擴大，以包含處理擴展的跨越記憶體描述符的程式，以便提供執行遠端記憶體存取的機能。

雖然本發明的實施例已被描述在一典型的 DMA 運作背景中，但本發明不受限於此。舉例來說，本發明也可使用一擴展的跨越記憶體描述符以供遠端複製運作。該複製運作可被一系統超管理程式幫助，或其他硬體以及/或是軟體機制，其中該伺服端分區經由作業系統和系統超管理程式之間一妥善提供文件的介面，提供資訊給該機制，例如該系統超管理程式。被提供的該資訊從該擴展的跨越記憶體描述符而來(例如識別符識別一客戶端分區的遠端記憶體)。該系統超管理程式之後可執行該真正的複製運作，從伺服端到客戶端或是從客戶端到伺服端。

該運作可由伺服端開始被實施，並因此該伺服端可從客戶端分區拉回該資料(例如用 xmemin 系統呼叫)或可將資料推到客戶端分區(例如用 xmemout 系統呼叫)。該系統超管理程式可執行真正的複製運作，因為其比在上運作的作業系統具有一較高的監督特權。這允許該系統超管理程式去存取任何分區的任何記憶體。這本質的遠端複製運作促使更多精密資料的設計和實施，在伺服端和客戶端之間交換通訊協定。

很重要需注意的是，當本發明已被描述在一全功能資料處理系統的背景中，一般熟此技藝者將可查知，本發明的過程有能力被散佈在指令的一電腦可讀取媒體形式以及很多形式中，不管用來散佈的訊號媒介(signal bearing media)的特定的類型並且與本發明同樣的應用。電腦可讀取媒介的例子包含可記錄類型媒介(recordable-type media)，例如一軟碟、一硬碟驅動程式、一 RAM、CD-ROMs、DVD-ROMs、以及傳送類型媒介(transmission-type media)，例如數位和類比通訊連結、使用傳輸形式的有線或無線通訊連結，例如射頻和光波傳導。該電腦可讀取媒介可用編碼的形式，而在一特定資料處理系統中被解碼以供真正的使用。

為了說明和描述的目的，本發明的描述已被呈現，並且本發明並未耗盡或限制。許多修飾和改變對熟此技藝者來說將是明顯的。實施例的選擇和描述是為了對本發明的原則和實際應用作最好的解說，並且使其他熟此技藝者可了解當適於特定仔細考量的運用時，本發明可

有不同的實施例與其不同的修飾。

【圖式簡單說明】

本發明新穎的特徵於後述的申請專利範圍中被提出。然而，該發明本身、較佳實施例、更進一步的標的、以及其長處，將由參考例證實施例之實施方式，並結合其附有之圖示，以被最佳的理解，其中：

圖 1 是一資料處理系統的一方塊圖，本發明可被實施於其中；

圖 2 是一例示的邏輯分區平台的一方塊圖，本發明可被實施於其中；

圖 3 是一圖示，說明用以產生與本發明實施例一致的一擴展的跨越記憶體描述符之機制的一例子；

圖 4 是一圖示，說明依據本發明一實施例之機制的一例子，該實施例供使用一擴展的跨越記憶體描述符去執行需要遠端記憶體存取的一運作；

圖 5 是一流程圖，概述用以產生與本發明實施例一致的一擴展的跨越記憶體描述符之過程範例；以及

圖 6 是一流程圖，概述用以使用與本發明實施例一致的一擴展的跨越記憶體描述符之過程範例。

【主要元件符號說明】

100	資料處理系統
101~104	處理器
107	系統匯流排
109	記憶體控制器/高速快取記憶體
110	I/O 匯流排橋
113	I/O 匯流排
114、122、130、140	PCI 主橋
115	PCI 本地匯流排
116、124、132、142	PCI-to-PCI 橋
118-119、123、126-127、131、133、141、144-145	PCI 匯流排
120-121、128-129、136	PCI I/O 轉接器
135	服務處理器
148	圖形轉接器
149	硬碟轉接器
150	硬碟
160-163	本地記憶體
170-175	I/O 槽
191	記憶體
190	硬體 OP-控制板
192	NVRAM 儲存
193	PCI/ISA 橋
194	服務處理器信箱介面和 ISA 匯流排存取口之邏輯
195	PCI 匯流排

196	ISA 匯流排
200	邏輯分區平台
202、204、206、208	作業系統
203、205、207、209	分區
210	分區管理韌體
211、213、215、217	分區韌體
230	分區的硬體
232-238	處理器
240-246	系統記憶體單元
248-262	I/O 轉接器
270	儲存單元
280	硬體管理操作台
290	服務處理器
298	NVRAM 儲存
310	邏輯分區
312-316	其他邏輯分區
320	作業系統
325	作業系統核心服務
330	系統超管理程式
332-338	裝置樹
340-360	本地記憶體
370-390	擴展的跨越記憶體描述符
400	客戶端
410	應用
415	記憶體

420	作業系統核心空間
422	I/O 堆疊
424	客戶端虛擬裝置驅動程式
430	伺服端虛擬驅動程式
440	擴展的跨越記憶體描述符
450	I/O 堆疊
460	實體裝置驅動程式
470	I/O 轉接器
480	伺服端

五、中文發明摘要：

一種系統和方法，用以擴展一伺服端分區環境中跨越記憶體描述符的效用，以至於跨越記憶體描述符(cross-memory descriptor)可用來描述另一分區的記憶體，例如一客戶端分區記憶體(之後參照為一"遠端"記憶體)被提供。由該系統和方法，當一邏輯分區計算系統中之一作業系統被初始化，作業系統核心服務則被召喚，其檢查該計算系統的一裝置樹並且產生一擴展的跨越記憶體描述符，該跨越記憶體描述符描述另一邏輯分區的本地記憶體，其為本邏輯分區的一遠端記憶體。當一遠端記憶體的存取是執行一運作之所需，則該伺服端分區的作業系統使用儲存的擴展的跨越記憶體描述符，去執行該遠端記憶體的存取。

六、英文發明摘要：

A system and method for extending the use of the cross-memory descriptor in a server partition environment such that it may be used to describe another partition's memory, e.g., a client partition's memory (referred to hereafter as a "remote" memory), are provided. With the system and method, when an operating system in a logically partitioned computing system is initialized, operating system kernel services are invoked that examine a device tree of the computing system and generate an extended cross-memory descriptor that describes the local memory of another logical partition, which is a remote memory to the present logical partition. When an access to a remote memory is required to perform an operation, the operating system of the server partition uses the stored extended cross-memory descriptor to perform the remote memory access.

十、申請專利範圍：

1. 一種在一資料處理系統中供第一邏輯分區中的一第一程序去存取在一第二邏輯分區中之一遠端記憶體之方法，該方法包含：

取得針對該遠端記憶體之一擴展的跨越記憶體(cross-memory)描述符，其中該擴展的跨越記憶體描述符提供該遠端記憶體之一描述；以及

供該第一邏輯分區中之該第一程序，基於該擴展的跨越記憶體描述符，以存取該第二邏輯分區中之該遠端記憶體。

2. 如請求項 1 所述之方法，其中該擴展的跨越記憶體描述符包含一第一欄位，該第一欄位指派該擴展的跨越記憶體描述符作為描述一不同的邏輯分區中之一遠端記憶體。

3. 如請求項 2 所述之方法，其中該擴展的跨越記憶體描述符更包含一第二欄位，該第二欄位識別在第二邏輯分區中之該遠端記憶體，及包含一第三欄位，該該第三欄位指定該遠端記憶體之大小，以及包含一第四欄位，該第四欄位指定該遠端記憶體中之一起始位址。

4. 如請求項 1 所述之方法，其中，當在該第一邏輯分區中之一作業系統被初始化，即產生該擴展的跨越記憶體描述符。

5. 如請求項 1 所述之方法，其中該擴展的跨越記憶體描述符被一作業系統核心服務(operating system kernel service)所產生，該作業系統核心服務附加其他邏輯分區之遠端記憶體到該第

一邏輯分區的作業系統。

6. 如請求項 5 所述之方法，其中，基於該第二邏輯分區之一裝置樹(device tree)，該作業系統核心服務產生該擴展的跨越記憶體描述符，而當在該第一邏輯分區中之該作業系統被初始化，該裝置樹被該作業系統核心服務所剖析。

7. 如請求項 6 所述之方法，其中該裝置樹被一系統超管理程式(hypervisor)所維持。

8. 如請求項 1 所述之方法，其中該第一邏輯分區在一伺服端計算裝置中，而該第二邏輯分區在一客戶端計算裝置中。

9. 如請求項 1 所述之方法，其中存取該第二邏輯分區中之該遠端記憶體，其包含：用該擴展的跨越記憶體描述符執行一直接記憶體存取操作。

10. 如請求項 1 所述之方法，其中基於該擴展的跨越記憶體描述符，存取該第二邏輯分區中之該遠端記憶體，該方法包含：

傳送該擴展的跨越記憶體描述符往下至一輸入/輸出(I/O)堆疊並到一實體的裝置驅動程式；

基於該擴展的跨越記憶體描述符，產生一直接記憶體存取操作；以及

遞交該直接記憶體存取操作到一 I/O 轉接器，其中該 I/O 轉接器傳送該直接記憶體存取操作到該第二邏輯分區。

11. 一種在一電腦可讀的媒介中供第一邏輯分區中之一第一程序去存取第二邏輯分區中之一遠端記憶體之電腦程式產品，該電腦程式產品包含：

第一指令，針對該遠端記憶體，供取得一擴展的跨越記憶體描述符，其中該擴展的跨越記憶體描述符提供該遠端記憶體之一描述；以及

第二指令，供該第一邏輯分區中之該第一程序，基於該擴展的跨越記憶體描述符，以存取第二邏輯分區中之該遠端記憶體。

12. 如請求項 11 所述之電腦程式產品，其中該擴展的跨越記憶體描述符包含一第一欄位，該第一欄位指派該擴展的跨越記憶體描述符作為描述一不同的邏輯分區中之一遠端記憶體。

13. 如請求項 12 所述之電腦程式產品，其中該跨越記憶體描述符更包含一第二欄位，該第二欄位識別在第二邏輯分區中之該遠端記憶體，及包含一第三欄位，該第三欄位指定該遠端記憶體之大小，以及包含一第四欄位，該第四欄位指定該遠端記憶體中之一起始位址。

14. 如請求項 11 所述之電腦程式產品，其中，當在該第一邏輯分區中之一作業系統被初始化，即產生該擴展的跨越記憶體描述符。

15. 如請求項 11 所述之電腦程式產品，其中該擴展的跨越記憶體描述符被一作業系統核心服務所產生，該作業系統核心服

務附加其他邏輯分區之遠端記憶體到該第一邏輯分區的作業系統。

16. 如請求項 15 所述之電腦程式產品，其中，基於該第二邏輯分區之一裝置樹，該作業系統核心服務產生該擴展的跨越記憶體描述符，而當在該第一邏輯分區中之該作業系統被初始化，該裝置樹被該作業系統核心服務所剖析。

17. 如請求項 16 所述之電腦程式產品，其中該裝置樹被一系統超管理程式所維持。

18. 如請求項 11 所述之電腦程式產品，其中用來存取該第二邏輯分區中之該遠端記憶體的該第二指令，包含使用該擴展的跨越記憶體描述符來執行一直接記憶體存取操作的指令。

19. 如請求項 11 所述之電腦程式產品，其中基於該擴展的跨越記憶體描述符，存取該第二邏輯分區中之該遠端記憶體之該第二指令，該第二指令包含：

指令，供傳送該擴展的跨越記憶體描述符往下至一輸入/輸出(I/O)堆疊並到一實體的裝置驅動程式之指令；

指令，供基於該擴展的跨越記憶體描述符，產生一直接記憶體存取操作之指令；以及

指令，供遞交該直接記憶體存取操作到一 I/O 轉接器之指令，其中該 I/O 轉接器傳送該直接記憶體存取操作到該第二邏輯分區。

20. 一種供一第一邏輯分區中之一第一程序去存取一第二邏輯分區中之一遠端記憶體之系統，該系統包含：

供取得針對該遠端記憶體之一擴展的跨越記憶體描述符之裝置，其中該跨越記憶體描述符提供該遠端記憶體之一描述；以及

供針對該第一邏輯分區中之該第一程序，基於該擴展的跨越記憶體描述符，以存取該第二邏輯分區中之該遠端記憶體之裝置。

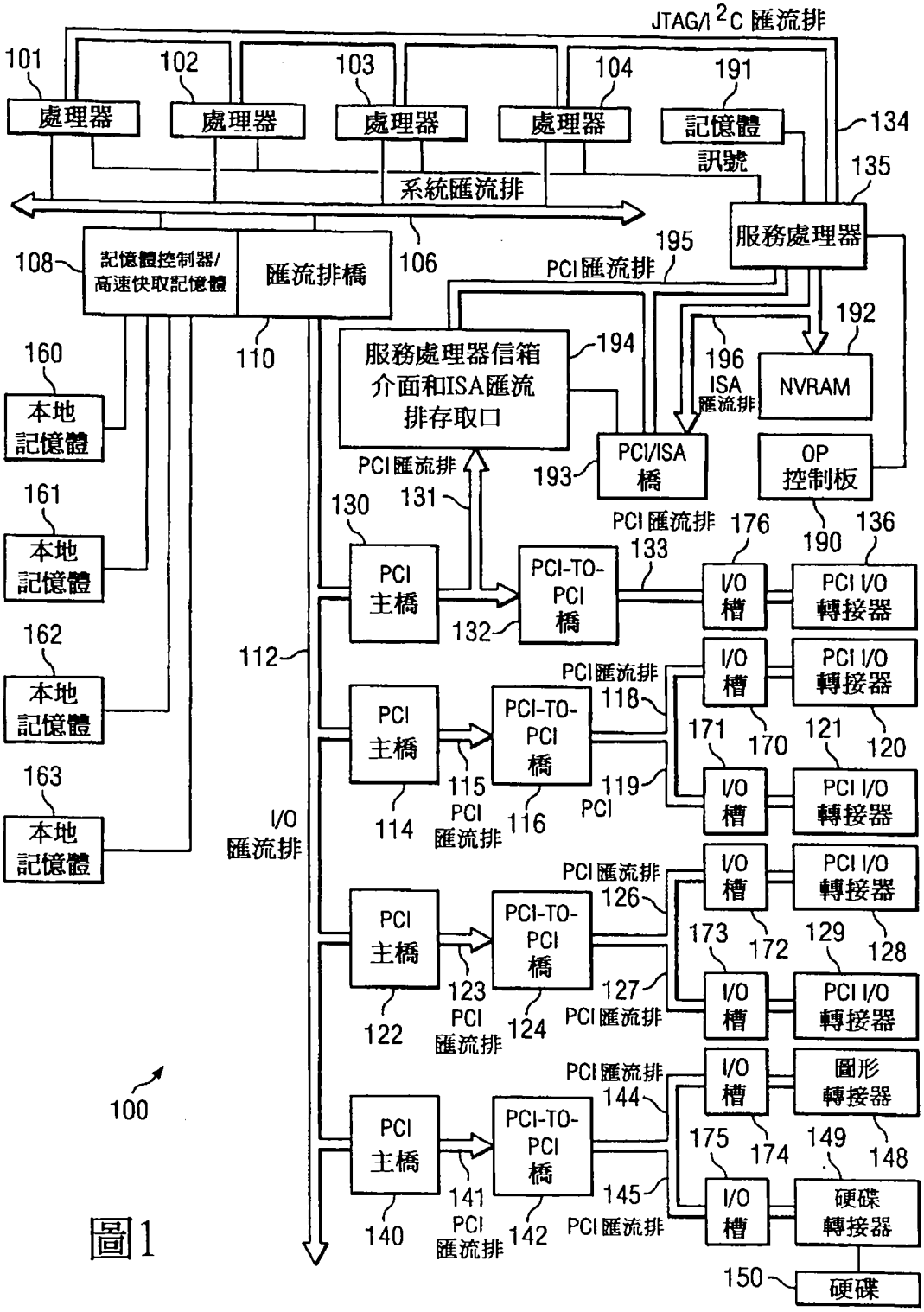

JTAG/I 2C 匯流排
101
處理器
102
處理器
103
處理器
104
處理器
191
記憶體
訊號
134
135
系統匯流排
106
108
記憶體控制器/
高速快取記憶體
匯流排橋
110
195
PCI匯流排
服務處理器
192
NVRAM
196
ISA
匯流排
160
本地
記憶體
服務處理器信箱
介面和ISA匯流
排存取口
194
PCI匯流排
193
PCI/ISA
橋
OP
控制板
161
本地
記憶體
130
131
PCI匯流排
176
190
136
PCI
主橋
PCI-TO-
PCI
橋
133
I/O
槽
PCI I/O
轉接器
132
162
本地
記憶體
112
PCI匯流排
118
171
170
121
120
PCI
主橋
PCI-TO-
PCI
橋
115
PCI
匯流排
119
PCI
163
本地
記憶體
I/O
匯流排
114
116
PCI匯流排
126
173
172
129
128
PCI
主橋
PCI-TO-
PCI
橋
123
PCI
匯流排
127
PCI匯流排
122
124
PCI匯流排
144
175
174
149
148
100
PCI
主橋
PCI-TO-
PCI
橋
圖形
轉接器
141
PCI
匯流排
145
PCI匯流排
140
142
硬碟
轉接器
150
硬碟

圖1

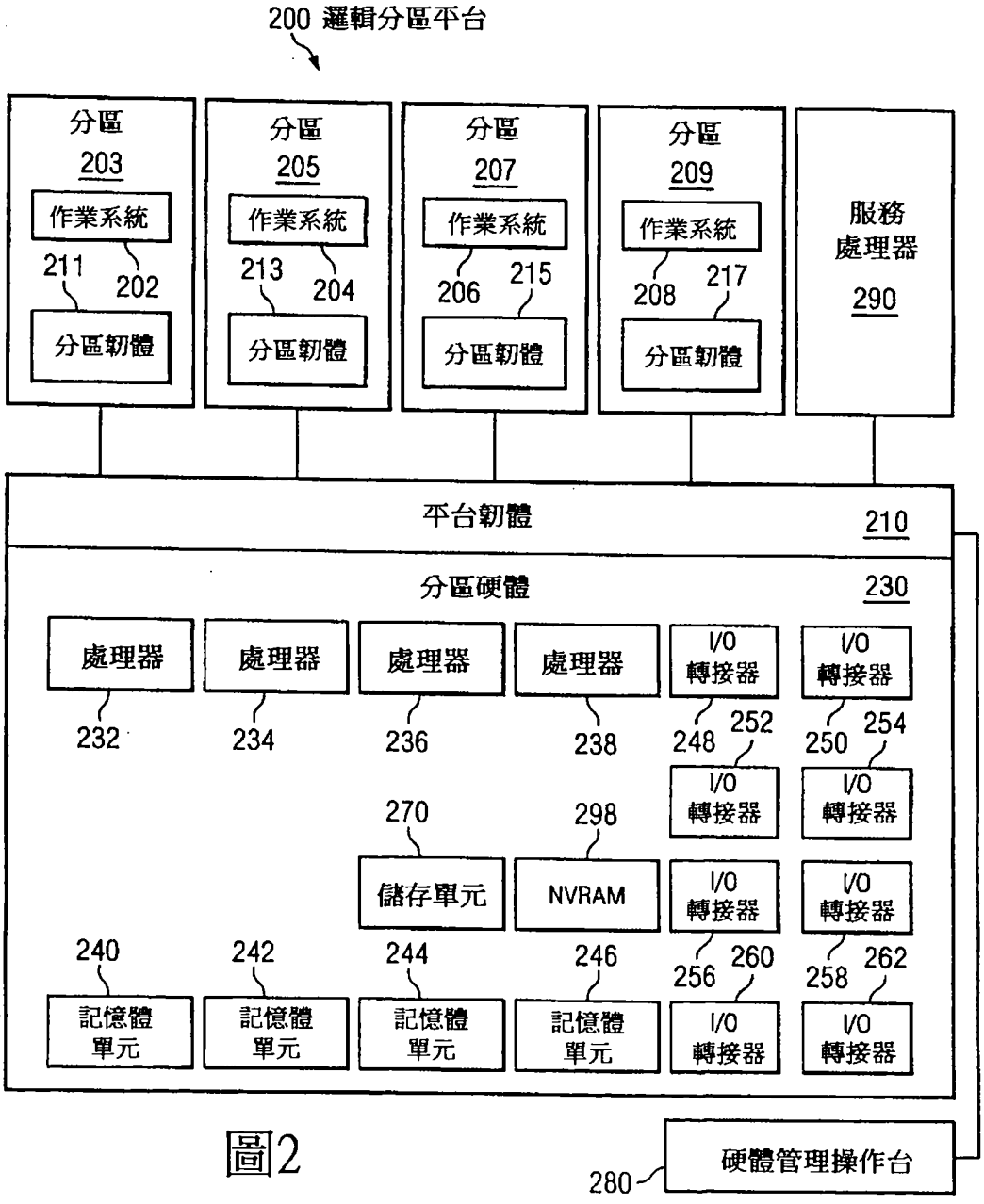

200 邏輯分區平台
分區
203
作業系統
211
202
分區韌體
分區
205
作業系統
213
204
分區韌體
分區
207
作業系統
206
215
分區韌體
分區
209
作業系統
208
217
分區韌體
服務
處理器
290
平台韌體
210
分區硬體
230
處理器
232
處理器
234
處理器
236
處理器
238
I/O
轉接器
248
252
I/O
轉接器
I/O
轉接器
250
254
I/O
轉接器
270
儲存單元
298
NVRAM
I/O
轉接器
I/O
轉接器
240
記憶體
單元
242
記憶體
單元
244
記憶體
單元
246
記憶體
單元
256
260
I/O
轉接器
258
262
I/O
轉接器
280
硬體管理操作台

圖2

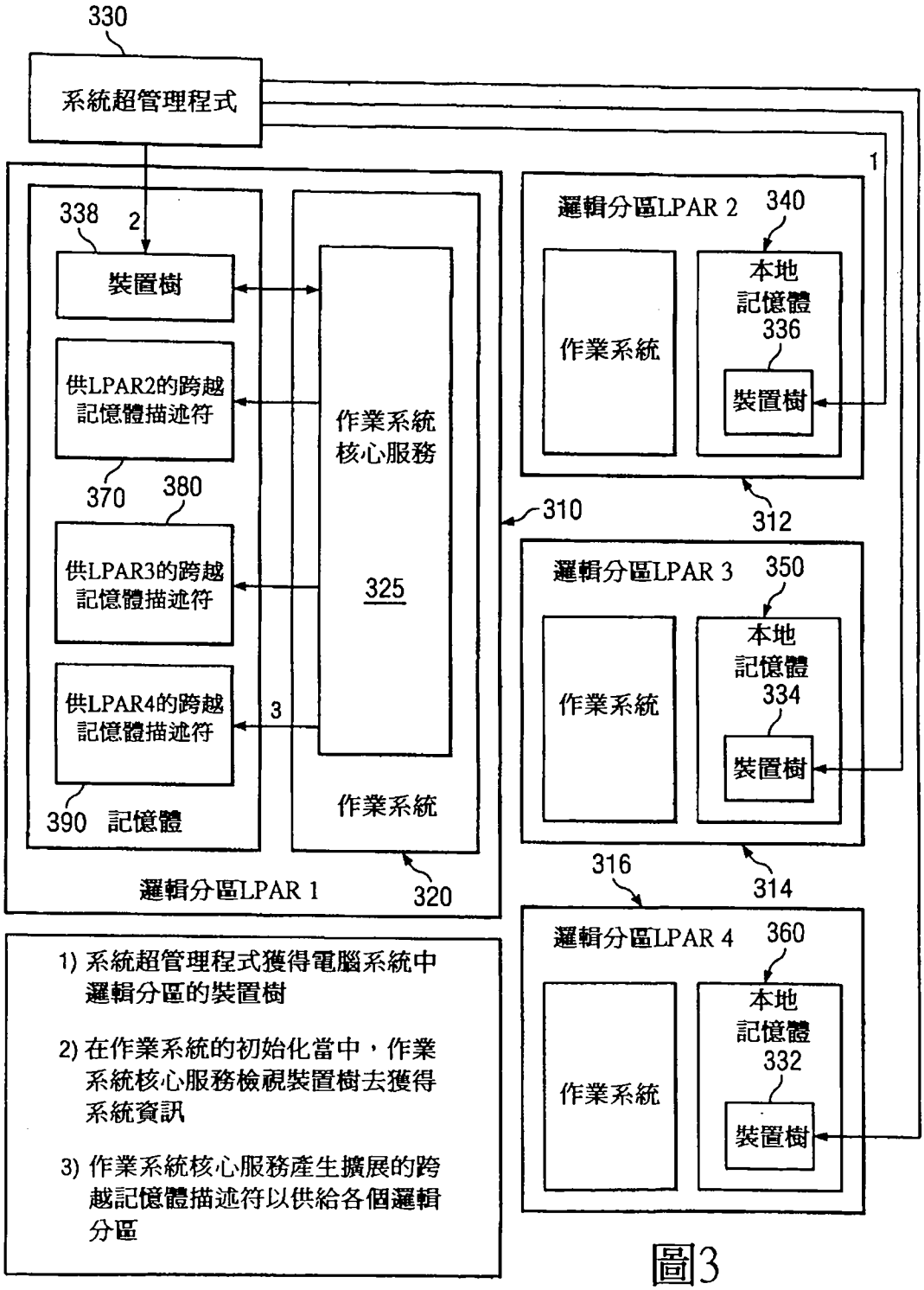
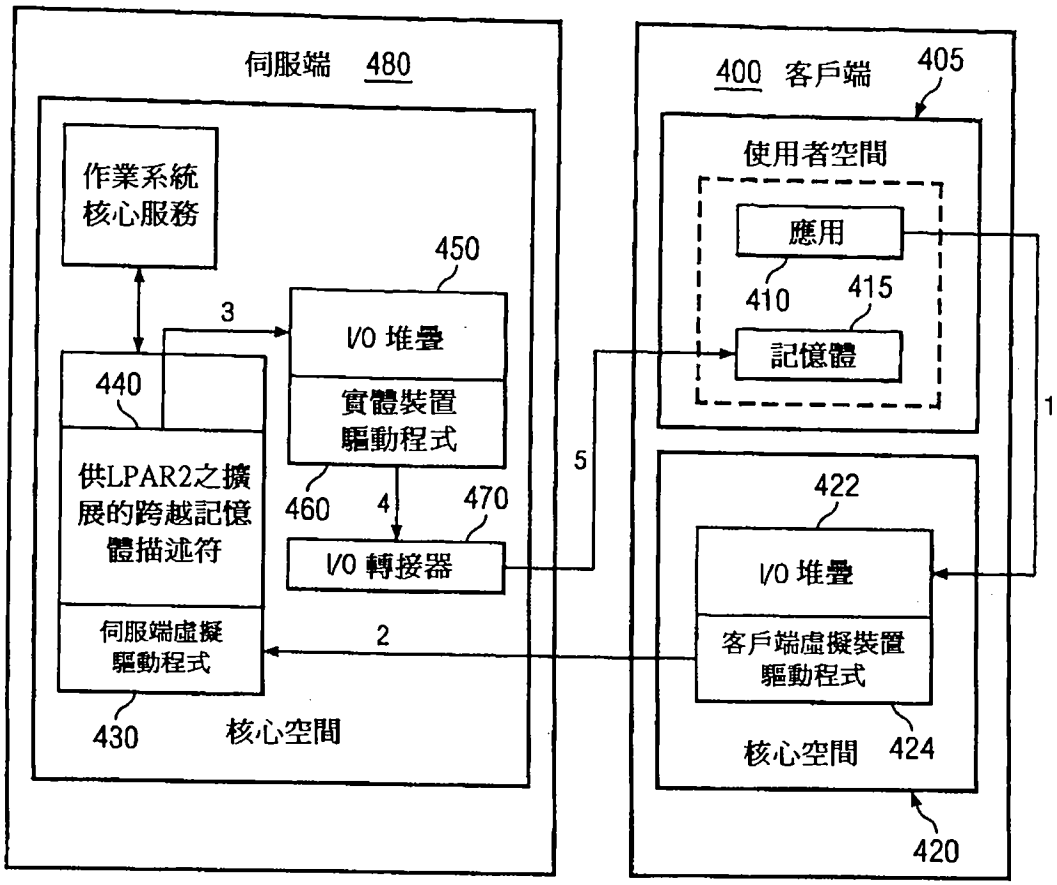

330
系統超管理程式
1
邏輯分區LPAR 2
340
作業系統
本地
記憶體
336
裝置樹
312
310
邏輯分區LPAR 3
350
作業系統
本地
記憶體
334
裝置樹
314
316
邏輯分區LPAR 4
360
作業系統
本地
記憶體
332
裝置樹
338
2
裝置樹
供LPAR2的跨越
記憶體描述符
370
380
供LPAR3的跨越
記憶體描述符
供LPAR4的跨越
記憶體描述符
3
390 記憶體
作業系統
核心服務
325
作業系統
邏輯分區LPAR 1
320
1) 系統超管理程式獲得電腦系統中邏輯分區的裝置樹
2) 在作業系統的初始化當中，作業系統核心服務檢視裝置樹去獲得系統資訊
3) 作業系統核心服務產生擴展的跨越記憶體描述符以供給各個邏輯分區

圖3



1) 要求I/O運作接收需要的遠端記憶體運作到LPAR2

2) 傳遞I/O要求到伺服端虛擬驅動程式並爲LPAR2識別擴展的跨越記憶體描述符

3) 傳遞供LPAR2之擴展的跨越記憶體描述符和I/O要求到I/O堆疊

4) 供LPAR2之擴展的跨越記憶體描述符和I/O要求被從介面傳遞到介面穿越I/O堆疊，並被送至I/O轉接器。使用跨越記憶體描述符設定遠端直接記憶體存取(DMA)

5) 在LPAR2中I/O轉接器直接記憶體存取(DMAs)遠端記憶體

圖4

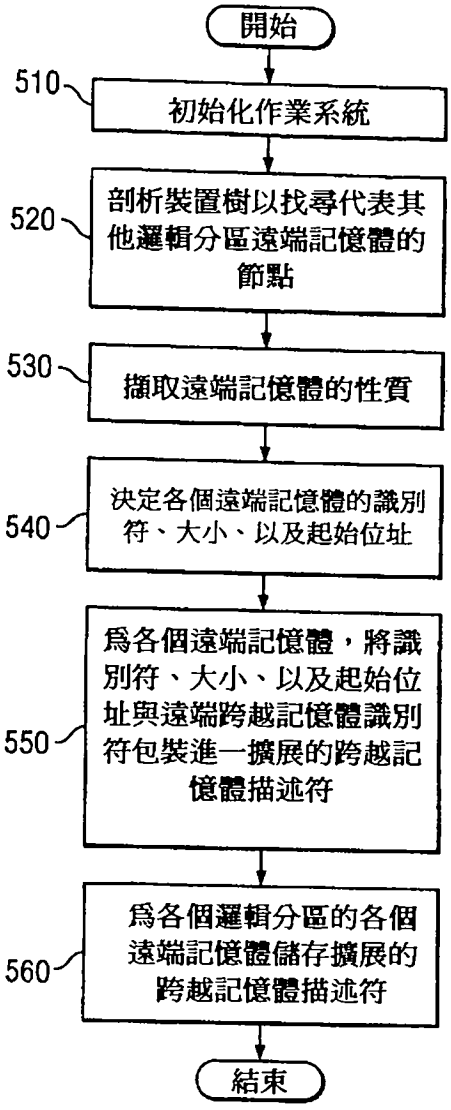

開始
510
初始化作業系統
520
剖析裝置樹以找尋代表其他邏輯分區遠端記憶體的節點
530
擷取遠端記憶體的性質
540
決定各個遠端記憶體的識別符、大小、以及起始位址
550
爲各個遠端記憶體，將識別符、大小、以及起始位址與遠端跨越記憶體識別符包裝進一擴展的跨越記憶體描述符
560
爲各個邏輯分區的各個遠端記憶體儲存擴展的跨越記憶體描述符
結束

圖5

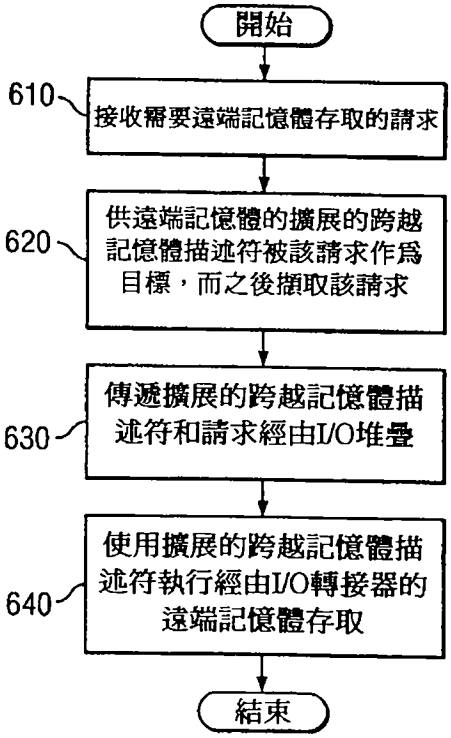

開始
610
接收需要遠端記憶體存取的請求
620
供遠端記憶體的擴展的跨越記憶體描述符被該請求作爲目標，而之後擷取該請求
630
傳遞擴展的跨越記憶體描述符和請求經由I/O堆疊
640
使用擴展的跨越記憶體描述符執行經由I/O轉接器的遠端記憶體存取
結束

圖6

七、指定代表圖：

(一)本案指定代表圖為：第 1 圖。

(二)本代表圖之元件符號簡單說明：

100	資料處理系統
101~104	處理器
106	系統匯流排
108	記憶體控制器/高速快取記憶體
110	I/O 匯流排橋
112	I/O 匯流排
114、122、130、140	PCI 主橋
115	PCI 本地匯流排
116、124、132、142	PCI-to-PCI 橋
118-119、123、126-127、131、133、141、144-145	PCI 匯流排
120-121、128-129、136	PCI I/O 轉接器
135	服務處理器
148	圖形轉接器
149	硬碟轉接器
150	硬碟
160-163	本地記憶體
170-175	I/O 槽
190	硬體 OP-控制板
191	記憶體
192	NVRAM 儲存
193	PCI/ISA 橋

194 服務處理器信箱介面和 ISA 匯流排存取口之邏輯

195 PCI 匯流排

196 ISA 匯流排

八、本案若有化學式時，請揭示最能顯示發明特徵的化學式：

無