



(19)中華民國智慧財產局

(12)發明說明書公開本

(11)公開編號：TW 201249154 A1

(43)公開日：中華民國 101 (2012) 年 12 月 01 日

(21)申請案號：101104398

(22)申請日：中華民國 101 (2012) 年 02 月 10 日

(51)Int. Cl. : **H04L29/06 (2006.01)**

(30)優先權：2011/02/11	美國	61/441,911
2011/02/18	美國	61/444,323
2011/03/14	美國	61/452,422
2011/05/13	美國	61/485,631
2011/06/14	美國	61/496,812
2011/06/24	美國	61/501,046
2011/06/24	美國	61/500,798
2011/07/15	美國	61/508,564

(71)申請人：內數位專利控股公司 (美國) INTERDIGITAL PATENT HOLDINGS, INC. (US)
美國

(72)發明人：王崇廣 WANG, CHONGGANG (CN)；路瑟爾二世 保羅 RUSSELL, JR., PAUL L. (US)；陸廣 LU, GUANG (CA)；席德 戴爾 SEED, DALE N. (US)；董利君 DONG, LIJUN (CN)；史塔新尼克 麥克 STARSINIC, MICHAEL F. (US)

(74)代理人：蔡清福

申請實體審查：無 申請專利範圍項數：37 項 圖式數：48 共 247 頁

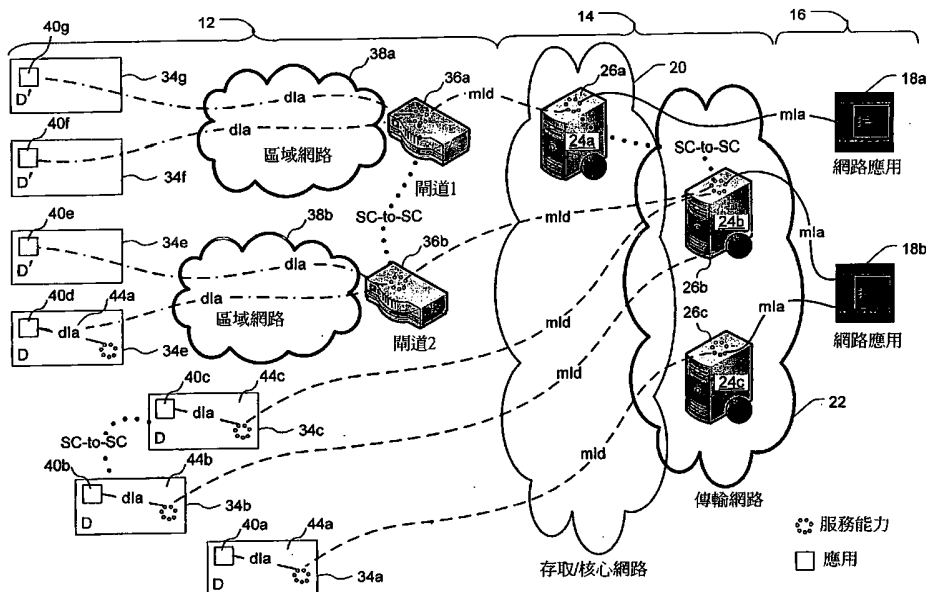
(54)名稱

管理機器對機器(M2M)實體系統、方法及裝置

SYSTEMS, METHODS AND APPARATUS FOR MANAGING MACHINE-TO-MACHINE (M2M) ENTITIES

(57)摘要

公開了用於管理機器對機器(M2M)實體的系統、方法和設備。本文所包括的是可包括於管理 M2M 環境中用於管理 M2M 實體的一個或多個管理層的方法。該方法還可包括使用複數管理層來管理 M2M 區域網路，其中 M2M 區域網路可包括一個或多個 M2M 端裝置。M2M 端裝置可包括例如 M2M 開道和/或 M2M 裝置。管理層可包括應用管理層、服務管理層、網路管理層和裝置管理層的任一個。管理層可提供 M2M 實體的配置管理、故障管理和性能管理的任一個。



12：裝置域

14：網路域

16：網路應用域

18a：M2M 網路應用

18b：M2M 網路應用

20：存取和/或核心網路

22：傳輸網路

24b：M2M 伺服器

24c：M2M 伺服器

26c：網路服務能力層
("N-SCL")

34a：M2M 裝置

34b：M2M 裝置

34c：M2M 裝置

34e：M2M 裝置

34f：M2M 裝置

34g：M2M 裝置

36a：M2M 閘道

36b：M2M 閘道

38a：M2M 區域網路

38b：M2M 區域網路

40a：M2M 應用
("DA")40b：M2M 應用
("DA")40c：M2M 應用
("DA")40d：M2M 應用
("DA")40f：M2M 應用
("DA")40g：M2M 應用
("DA")

44a：SCL("D-SCL")

44b：SCL("D-SCL")

44c：SCL("D-SCL")

dIa：參考點

mla：參考點

mld：參考點

SC：服務能力



(19)中華民國智慧財產局

(12)發明說明書公開本

(11)公開編號：TW 201249154 A1

(43)公開日：中華民國 101 (2012) 年 12 月 01 日

(21)申請案號：101104398

(22)申請日：中華民國 101 (2012) 年 02 月 10 日

(51)Int. Cl. : **H04L29/06 (2006.01)**

(30)優先權：2011/02/11	美國	61/441,911
2011/02/18	美國	61/444,323
2011/03/14	美國	61/452,422
2011/05/13	美國	61/485,631
2011/06/14	美國	61/496,812
2011/06/24	美國	61/501,046
2011/06/24	美國	61/500,798
2011/07/15	美國	61/508,564

(71)申請人：內數位專利控股公司 (美國) INTERDIGITAL PATENT HOLDINGS, INC. (US)
美國

(72)發明人：王崇廣 WANG, CHONGGANG (CN)；路瑟爾二世 保羅 RUSSELL, JR., PAUL L. (US)；陸廣 LU, GUANG (CA)；席德 戴爾 SEED, DALE N. (US)；董利君 DONG, LIJUN (CN)；史塔新尼克 麥克 STARSINIC, MICHAEL F. (US)

(74)代理人：蔡清福

申請實體審查：無 申請專利範圍項數：37 項 圖式數：48 共 247 頁

(54)名稱

管理機器對機器(M2M)實體系統、方法及裝置

SYSTEMS, METHODS AND APPARATUS FOR MANAGING MACHINE-TO-MACHINE (M2M) ENTITIES

(57)摘要

公開了用於管理機器對機器(M2M)實體的系統、方法和設備。本文所包括的是可包括於管理 M2M 環境中用於管理 M2M 實體的一個或多個管理層的方法。該方法還可包括使用複數管理層來管理 M2M 區域網路，其中 M2M 區域網路可包括一個或多個 M2M 端裝置。M2M 端裝置可包括例如 M2M 開道和/或 M2M 裝置。管理層可包括應用管理層、服務管理層、網路管理層和裝置管理層的任一個。管理層可提供 M2M 實體的配置管理、故障管理和性能管理的任一個。

六、發明說明：

【發明所屬之技術領域】

相關申請的交叉引用

本申請案涉及並要求 (i) 2011 年 2 月 11 日申請的美國臨時專利申請案第 61/441,911 號、名稱為 “Enhanced Gateway-Based Machine-To-Machine (M2M) Device Management” (代理人編號 IDC-10928US01); (ii) 2011 年 6 月 14 日申請的美國臨時專利申請案第 No. 61/496,812 號、名稱為 “Data Model For Managing M2M Area Networks and M2M Devices behind the M2M Gateway” (代理人編號 IDC-11072US01); (iii) 2011 年 6 月 24 日申請的美國臨時專利申請案第 No. 61/500,798 號、名稱為 “Data Model For Managing M2M Area Networks and M2M Devices behind the M2M Gateway” (代理人編號 IDC-11077US01); (iv) 2011 年 2 月 18 日申請的美國臨時專利申請案第 No. 61/444,323 號、名稱為 “Machine-To-Machine (M2M) Remote Entity Management” (代理人編號 IDC-10930US01); (v) 2011 年 3 月 14 日申請的美國臨時專利申請案第 No. 61/452,422 號、名稱為 “Machine-To-Machine (M2M) Remote Entity Management” (代理人編號 IDC-10954US01); (vi) 2011 年 5 月 13 日申請的美國臨時專利申請案第 No. 61/485,631 號、名稱為 “Remote Entity Management for Machine-to-Machine (M2M) Communications” (代理人編號 IDC-11047US01); (vii) 2011 年 6 月 24 日申請的美國臨時專利申請案第 No. 61/501,046 號、名稱為 “Remote Entity Management for Machine-To-Machine (M2M) Communications” (代理人編號 IDC-11078US01); 和 (viii)

2011 年 7 月 15 日申請的美國臨時專利申請案第 No. 61/508,564 號、名稱為 “Remote Entity Management for Machine-To-Machine (M2M) Communications” (代理人編號 IDC-11098US01) 的利益和/或優先權。上述每一個美國臨時專利申請案以引用的方式併入本文。

技術領域

本申請案涉及通信，並且特別地，涉及機器對機器 (“M2M”) 通信。

【先前技術】

機器對機器 (“M2M”) 通信涉及由被稱為機器的裝置和/或在它們之間實現的一類通信，這些裝置適用於經由這樣的 M2M 通信來發送、接收或交換用於執行各種應用 (“M2M 應用”) 的資訊，例如智慧測量、家庭自動化、電子健康和車隊 (fleet) 管理。總的來說，各種應用的執行，並且反過來說，伴隨這樣的執行的 M2M 通信由機器執行，而不必須用於觸發、發起和/或引起 M2M 通信開始的人類干涉。可理解地，M2M 應用的成功實現和興起可能取決於產業範圍內接受確保 (例如定義需求以便確保) 可由各種實體製造和操作的各種機器之間的相互操作性的標準。

【發明內容】

【實施方式】

示例系統架構

第 1A 圖-第 1C 圖是示出其中一個或多個實施例可被實現或執行的系統 10 的示例的框圖。這樣的實施例可包括例如指向機器到機器 (“M2M”) 通信和/或操作的實施例，包括管理 M2M 遠端實體，例如 M2M 應用、M2M 服務能力

(“SC”)、M2M 區域網路 (area network)、M2M 閘道和 M2M 裝置。

系統 10 可根據和/或使用一個或多個架構來配置和/或實現；這些架構中任一個可基於和/或根據各種標準來配置。這些標準可包括那些指向 M2M 通信和/或操作的，例如由歐洲電信標準協會 (“ETSI”) 發佈的、標題為 “Machine-to-Machine (M2M) Communications; Functional Architecture” 並被稱為 “ETSI TS 102 690” 的草案技術規範 (“TS”)。其他示例可包括由第 3 代合作夥伴計畫 (“3GPP”) 和/或第 3 代合作夥伴計畫 2 (“3GPP2”) 發佈的那些標準，包括那些涉及機器類型通信 (“MTC”) 的標準，例如標題為 “Technical Specification Group Services and System Aspects; Service requirements for Machine-Type Communications (MTC)” 的 3GPP TS 22.368。ETSI TS 102 690 和 3GPP TS 22.368 均以引用方式併入本文。系統 10 的架構也可根據其他標準。

系統 10 可包括裝置域 12、網路域 14 和網路應用域 16。網路應用域 16 可包括 M2M 網路應用 18a、18b。這些 M2M 網路應用 18a、18b 可在各別的主機裝置 (未示出) 上儲存、執行和/或駐留。替代地，M2M 網路應用 18a、18b 在相同的主機裝置 (也未示出) 上儲存、執行和/或駐留。主機裝置可包括一個或多個伺服器，例如包括主機應用伺服器，並且可被部署在一個或多個通用或專用電腦、個人電腦、大型主機、迷你電腦、伺服器類型電腦和/或在任何適當的作業系統上運行並能夠執行軟體的任意以處理器為基礎的平台。主機裝置可包括若干元件，這些元件可形成在單一裝置中並集中在單一節點上；服務、用戶端、對等 (peer) 或其他。替代地，主機裝置的元件可從兩個或更多個分離的裝置

組成，並且正如這樣可分別在複數節點間；服務、用戶端、對等或其他分布。

網路域 14 可包括存取和/或核心（“存取/核心”）網路 20 以及傳輸網路 22。存取/核心網路 22 可以是例如根據用於(i) 數位用戶線路技術（總稱為“xDSL”）、(ii) 混合光纖同軸（“HFC”）網路、(iii) 可編程邏輯控制器（“PLC”）、(iv) 衛星通信和網路、(v) 全球行動通信系統（“GSM”）/增強型資料 GSM 環境（“EDGE”）無線電存取網路（“GERAN”）、(vi) 通用行動通信系統（“UMTS”）陸上無線電存取網路（“UTRAN”）、(vii) 演進型 UTRAN（“eUTRAN”）、(viii) 無線區域網路（“WLAN”）、微波存取全球互通（“WiMAX”）等的一個或多個協定而配置用於通信的網路。可代表存取和/或核心網路 20 的示例存取和/或核心網路的細節參考第 48A 圖-第 48E 圖而描述於下。可代表傳輸網路 22 的示例傳輸網路的細節也參考第 48A 圖-第 48E 圖而描述於下。

存取/核心網路 20 還根據網際網路協定（“IP”）系列提供連線。存取/核心網路 20 也可根據其他通信協定提供連線。此外，存取/核心網路 20 可提供服務和網路控制功能、與其他網路的互連和漫遊服務。以示例的方式，存取/核心網路 20 可以是根據由 3GPP 發佈的協定、用於高級網路的 ETSI 電信和涵蓋網際網路之服務及協定（“TISPAN”）以及由 3GPP2 發佈的協定而配置用於通信的網路。

存取/核心網路 20 可包括 M2M 伺服器 24a，並且傳輸網路 22 可包括 M2M 伺服器 24b、24c。M2M 伺服器 24a-24c 的每一個可由各自的服務供應商所有、維護和/或營運。例如，M2M 伺服器 24a 可由無線（例如胞元）電信服務供應商所有、維護和/或營運，而 M2M 伺服器 24b、24c 可由其他服務供應商所有、維護和/或營運。在一些實例中，第一、

第二和第三 M2M 伺服器 24a-24b 的每一個的所有權、維護和/或營運可在兩個或更多個供應商之間分開。

M2M 伺服器 24a-24c 可包括或被配置以包括各自的網路服務能力層 (“N-SCL”) 26a-26c 和各自的網路通信協定棧 28a-28c (第 1B 圖)。N-SCL 26a-26c 可包括各自的 M2M SC (“N-SC”) 30a-30c 和伴隨的資源結構 32a-32c (第 1B 圖)。

裝置域 12 可包括 M2M 裝置 34a-34g、M2M 閘道 36a、36b 和 M2M 區域網路 38a、38b。M2M 裝置 34a-34g 可包括各自的 M2M 應用 (“DA”) 40a-40h 和各自的網路通信協定棧 42a-42g。M2M 裝置 34a-34g 中的一些，即 M2M 裝置 34a-34d (此後稱為 “D 34a-34d” 或 “D 類型裝置 34a-34d”) 可包括或被配置以包括各自的 SCL (“D-SCL”) 44a-44d。D-SCL 44a-44d 可包括或被配置以包括各自的 SC (“D-SC”) 46a-46d 和伴隨的資源結構 48a-48d (第 1B 圖)。

M2M 裝置 34e-34g (此後稱為 “D’ 34e-34g” 或 “D’ 類型裝置 34e-34g”) 沒有 D-SCL。D 34a-34d 在其他方面也可不同於 D’ 34e-34g。例如，D’ 34e-34f 可受到資源限制，例如處理能力和儲存限制，而 D 34a-34d 可不受到這樣的資源限制。替代地和/或附加地，D’ 34e-34g 可包括或被配置以包括不同於 D-SCL 44a-44d 的功能。

M2M 閘道 36a、36b 可包括或被配置以包括各自的 M2M 應用 (“GA”) 50a、50b、各自的網路通信協定棧 52a、52b 和各自的 SCL (“G-SCL”) 54a、54b (第 1 圖)。G-SCL 54a、54b 可包括或被配置以包括各自的 SC (“G-SC”) 56a、56b 和伴隨的資源結構 58a-58e (第 1B 圖)。

M2M 區域網路 38a、38b 可通信地耦合於 M2M 閘道 36a、36b 和其他 M2M 閘道 (未示出)，如果有的話。M2M 區域網路 38a、38b 可包括除了 D 34d 和 D’ 34e-34g 以外的

M2M 裝置（未示出）。這些額外的 M2M 裝置可以是 D 類型或 D'類型的裝置。M2M 區域網路 38a、38b 的每一個可使用例如網格和/或點對點架構來配置。

通信地將 D 34d 和 D' 34e 耦合在一起和/或在 D 34d、D' 34e 和/或 M2M 區域網路 38a 的鄰近 M2M 裝置之間的通信鏈路可以是有線和/或無線的。通信地將 D 34f 和 D' 34g 耦合在一起和/或在 D 34d、D' 34e 和/或 M2M 區域網路 38b 的鄰近 M2M 裝置之間的通信鏈路也可以是有線和/或無線的。

通信地將 D 34d、D' 34e 和 M2M 區域網路 38a 的其他 M2M 裝置耦合到 M2M 閘道 36a 的通信鏈路和通信地將 D' 34f、34g 和 M2M 區域網路 38b 的其他 M2M 裝置耦合到 M2M 閘道 36b 的通信鏈路一起可以是有線和/或無線的。這些通信鏈路的每一個可根據私有介面、標準介面和/或開放介面來定義。替代地，通信鏈路可被定義為參考點，例如 dIa 參考點。可代表這樣的 dIa 參考點的示例 dIa 參考點的細節可在 ETSI TS 102 690 中找到。

M2M 閘道 36a、M2M 閘道 36b 和 D 34b-34c 和 D 34a 可經由有線和/或無線通信鏈路而分別與 M2M 伺服器 24a、24b 和 24c 通信地耦合。這些通信鏈路可根據私有介面、標準介面和/或開放介面來定義。替代地，通信鏈路可被定義為參考點，例如 mIa 參考點。可代表這樣的 mIa 參考點的示例 mIa 參考點的細節可在 ETSI TS 102 690 中找到。

通信地耦合 M2M 網路應用 18a 和 M2M 伺服器 24a 的通信鏈路以及通信地耦合 M2M 網路應用 18b 和 M2M 伺服器 24b、24c 的通信鏈路可以是有線和/或無線的。這些通信鏈路的每一個可根據私有介面、標準介面和/或開放介面來定義。替代地，通信鏈路可被定義為參考點，例如 mIa 參考點。可代表這樣的 mIa 參考點的示例 mIa 參考點的細節可在

ETSI TS 102 690 中找到。

D 34a-34d 各自的 DA 40a-40d 與 D-SCL 44a-44d 之間的通信可使用 mla 參考點來執行。dla 和 mla 參考點可在 M2M 網路應用 18b 和 DA 40a-40e 之間以及在 M2M 網路應用 18a 和 DA 40f-40g 之間提供統一的介面。

雖然在第 1 圖中示出了 2 個 M2M 閘道和 8 個 M2M 裝置，但裝置域 12 可包括更多或更少的 M2M 閘道和更多或更少的 M2M 裝置。實際上，裝置域 12 可能具有許多 M2M 裝置和許多 M2M 閘道。附加地和/或替代地，系統 10 可包括更多或更少的 M2M 伺服器、更多或更少的 M2M 網路應用和更多或更少的 M2M 區域網路。

M2M 區域網路 38a、38b 的每一個可以是例如根據諸如電氣與電子工程師協會（“IEEE”）802.15.x、Zigbee、藍牙、網際網路工程工作小組（“IETF”）低功率有損網路上的路由（“ROLL”）、自動化國際協會（“ISA”）用於工業自動化的無線系統：處理控制和相關應用（“ISA100.11a”）等個人區域網路協定而被配置來通信的網路。M2M 區域網路 38a、38b 可根據例如參考第 48A 圖-第 48E 圖描述如下的其他網路協定來配置。

現在參考第 1C 圖，N-SC 30a-30c 的每一個可包括或被配置以包括應用允許（enablement）能力（“AE”），通用通信能力（“GC”），可達、定址和庫存能力（“RAR”），通信選擇能力（“CS”），遠端實體管理能力（“REM”），安全能力（“SEC”），歷史和資料保持能力（“HDR”），事務管理能力（“TM”），交互代理能力（“IP”）和電信運營商暴露（exposure）能力（“TOE”），這些的每一個可與用於從一個能力向另一個能力傳遞資訊（例如從經過處理的訊息）的路由功能相連接。N-SC 30a-30c 的每一個還可包括用於調度內

部軟體、用於管理作業系統介面等的管理器。

G-SC 56a、56b 的每一個和 D-SC 46a-46d 的每一個還可包括與路由功能和管理器相連接的 AE、GC、RAR、CS、REM、SEC、HDR、TM 和 IP。為了方便起見，此後字首“N”、“G”或“D”可添加於 AE、GC、RAR、CS、REM、SEC、HDR、TM、路由功能和管理器以區分 N-SC、G-SC 和 D-SC。字首“x”可添加於 AE、GC、RAR、CS、REM、SEC、HDR、TM、路由功能和管理器以共同地提及 N-SC、G-SC 和 D-SC。

N-SC 30a-30c、G-SC 56a、56b 和 D-SC 46a-46d 的每一個也可包括 SC 到 SC 互動能力，以便使能裝置對裝置（D2D）、閘道對閘道（G2G）和服務對伺服器（S2S）直接通信。N-SC 30a-30c、G-SC 56a、56b 和 D-SC 46a-46d（共同地稱為“xSC”）的一些或所有的部分可根據指向 M2M 通信和/或操作（諸如 ETSI TS 102 690）的標準來定義。

總的來說，SC 可定義和/或實現可由各種 M2M 應用使用的功能。為了便於此，SC 24 可通過一組開放介面將這樣的功能呈現給各種 M2M 應用。此外，SC 可使用存取/核心和/或傳輸網路 20、22 的功能（“網路功能”）。然而，SC 可向各種 M2M 應用隱藏網路特性，並且反而可透過 SC 為這樣的應用來處理網路管理。SC 可以是 M2M 特定的。替代地，SC 可以是一般的，例如為 M2M 應用和不是 M2M 應用的應用提供支援。如下更詳細的描述，N-SCL 資源結構 32a-32c、G-SCL 資源結構 58a-58e 和/或 D-SCL 44a-44d 的每一個可包括一個或多個資源和/或屬性，其可基於一個或多個 xSC 的架構而在階層中排列。

示例 xREM 管理層和功能

第 2 圖是示出定義用於執行 xREM 的功能架構的邏輯管理層 200 的示例集合的框圖。總的來說，管理層 200 可定義

用於管理通信模組、SCL 和應用的功能。管理層 200 還可區分管理功能、定義相應的管理物件和資源結構、並在 xREM 分別為 M2M 裝置 (“DREM”)、M2M 閘道 (“GREM”) 和 M2M 區域網路 (“NREM”) 識別管理功能。管理層 200 管理功能的區分可基於 M2M 遠端實體的類型。管理層 200 可包括例如定義用於管理 (i) M2M 應用、(ii) M2M SC、(iii) M2M 區域網路和 M2M 閘道、和 (iv) M2M 裝置的功能的分離層。管理層 200 的每一個可包括用於執行駐留在這樣的管理層中的、M2M 遠端實體的配置管理 210、故障 (fault) 管理 212、性能管理 214 等的功能。在一個實施方式中，管理層 200 可包括應用管理層 202、服務管理層 204、網路管理層 206 和裝置管理層 208。管理層 200 也可包括其他層。

示例 M2M 應用管理層

M2M 應用管理層 202 可處理管理 M2M 應用，包括在與管理 M2M 應用相關聯的 xREM 處來定義管理功能、定義管理物件和資源結構和識別管理功能。M2M 應用管理層 202 可處理在 M2M 裝置和/或 M2M 閘道(總稱為“D/G”)中 M2M 裝置應用的生命週期管理，這可包括安裝、更新、刪除、啟動、去啟動 D/G 中應用軟體的任一個。M2M 應用管理層 202 還可處理 D/G 中應用的配置管理。這可包括例如配置和/或提供對這樣的應用的初始設定和/或更新。M2M 應用管理層 202 可處理 D/G 中應用的故障管理，包括例如收集和/或獲取 (retrieve) 與故障相關的資訊。M2M 應用管理層 202 可處理 D/G 中應用的性能管理，包括例如收集和/或獲取與性能相關的資訊。M2M 應用管理層的所有者可以是例如 M2M 應用供應商。

示例 M2M 服務管理層

M2M 服務管理層 204 可處理管理 M2M SC，包括在與

管理 M2M SC 相關聯的 xREM 處來定義管理功能、定義管理物件和資源結構以及識別管理功能。M2M 服務管理層 204 可處理 D/G 中 SCL 的軟體/韌體更新；D/G 中 SCL 的配置管理，包括配置或提供對 SCL 的初始設定和/或更新；D/G 中 SCL 的故障管理，包括例如收集和獲取與故障相關的資訊；和 D/G 中 SCL 的性能管理，其可包括收集和獲取與性能相關的資訊。M2M 服務管理層 204 的所有者可以是例如 M2M 服務供應商。

示例 M2M 網路管理層

M2M 網路管理層 206 可處理管理 M2M 區域網路，包括在與管理 M2M 區域網路相關聯的 xREM 處來定義管理功能、定義管理物件和資源結構以及識別管理功能。例如，此層可控制路由管理、拓撲管理和網路生命期管理。由於 M2M 區域網路在許多情況下由 M2M GW 來連接，因此 M2M GW 可在網路管理層 206 中扮演角色。

M2M 網路管理層 206 可處理 M2M 區域網路的配置管理，包括例如配置和/或提供 M2M 區域網路的初始操作配置，這可包括配置 IPv6 位址字首、工作頻率、WPAN ID 等。M2M 網路管理層 206 還可處理 (i) 更新 D/G 的配置，這可包括更新 6LoWPAN/ROLL/CoAP 中的參數和/或常數；(ii) M2M 區域網路的故障管理，包括異常檢測（例如，過期或錯誤路由、環回（loopback）路由）和/或警報生成和/或處理；(iii) M2M 區域網路的性能管理，這可包括（例如整個）M2M 區域網路的工作週期管理、拓撲/路由管理和 QoS 管理的任一個。網路管理層 206 的所有者可以是 M2M 區域網路供應商，其可以是 M2M 應用供應商、M2M 服務供應商或 M2M 用戶。

示例 M2M 裝置管理層

裝置管理層 208 可處理管理例如 D/G 的 M2M 端裝置，包括在與 M2M 端裝置相關聯的 xREM 處來定義管理功能、定義管理物件和資源結構和識別管理功能。裝置管理層 208 可處理用於資源受限的 M2M 裝置的管理功能，這可包括例如工作週期管理和功率管理。裝置管理層 208 可處理 (i) D/G 的配置管理，這可包括配置 D/G 的初始操作和/或更新 D/G 的配置；(ii) D/G 的故障管理，這可包括異常檢測和警報生成和處理的任一個；(iii) D/G 的性能管理，這可包括例如受限資源(感測器、激發器、電源/電池、記憶體、CPU、通信介面等)的管理、節電管理(例如整個節點的工作週期、收發器的工作週期)和感測器/激發器管理(例如在不同的應用間共用)的任一個。裝置管理層 208 的所有者可以是 M2M 應用供應商、M2M 服務供應商、M2M 區域網路供應商或 M2M 用戶。

參考第 3A 圖，展示了示出用於根據管理層集合及其功能向 SCL 提供資源結構的示例資源結構架構 300 的框圖。在其上可提供這樣的資源結構的 SCL 可以是主機 SCL，例如 N-SCL 26a-26c 的任一個。在主機 SCL 上提供的資源結構可以主機 SCL、其他主機 SCL 和/或遠端 SCL 之間的同步方式，在一個或多個其他主機 SCL 和/或一個或多個遠端 SCL (例如 G-SCL 56a、56b 和/或 D-SCL 44a-44d) 上隨後提供。替代地，可在其上提供這樣的資源結構的 SCL 可以是遠端 SCL，例如 G-SCL 56a、56b 和/或 D-SCL 44a-44d 的任一個。在遠端 SCL 上提供的資源結構可以遠端 SCL、其他遠端 SCL 和/或主機 SCL 間的同步方式，在一個或多個其他遠端 SCL 和/或一個或多個主機 SCL (例如 N-SCL 26a-26c) 上隨後提供(例如整體或部分地複製)。附加地和/或替代地，資源結構架構 300 還可被用來在多個主機 SCL 和/或多個遠端 SCL

上提供資源結構。

資源結構架構 300 可包括適當 SCL 的根資源 (“<sclBase>”) 302、從屬於<sclBase>302 的複數資源 (“子資源 (sub-resources)”) 和一個或多個屬性 304。屬性 304 可與直接從屬於<sclBase>302 的子資源的一些或全部相關聯 (例如相同)。替代地, 屬性 304 可與直接和/或間接從屬於<sclBase>302 的子資源相關聯。直接從屬於<sclBase>302 的子資源可包括 SCL (“scls”) 子資源 306、應用 (“applications”) 子資源 308、容器 (“containers”) 子資源 310、組 (“groups”) 子資源 312、存取權 (“accessRights”) 子資源 314、訂閱 (“subscriptions”) 子資源 316、發現 (“discovery”) 子資源 318、存取狀態 (“accessStatus”) 子資源 320 和管理物件 (“mgmtObjs”) 子資源 322。

scls 子資源 306 可以是各自的 SCL 資源的集合; 其每一個可代表被授權與主機 SCL 交互作用 (例如以 M2M 服務註冊過程的方式) 的相關聯 (例如遠端) SCL。scls 子資源 306 的每個 SCL 資源可回應於相關聯 SCL 向它的本地 SCL 的成功註冊而創建, 反之亦然。scls 子資源 306 可包括、維護和/或儲存關於各自的已註冊 SCL 的上下文資訊。scls 子資源 306 的每一個可包括一個或多個子資源和/或一個或多個屬性 (未示出)。

應用子資源 308 可以是各自的應用資源的集合; 其每一個可包括、維護和/或儲存關於應用的資訊。應用子資源 308 的每個應用資源可回應於相關聯應用向本地 SCL 的成功註冊而創建。

容器子資源 310 可以是各自的容器資源的集合; 其每一個可以用於在應用和/或 SCL 間交換資訊的通用資源。容器子資源 310 的每個容器資源可經由將相應的容器用作緩

存資訊的仲介而促進應用和/或 SCL 間資訊的交換。

組子資源 312 可以是各自的組資源的集合。組子資源 312 的每個組資源可被用來定義和/或存取其他子資源的組，包括直接地和/或間接地從屬於<sclBase>302 的子資源。

accessRights 子資源 314 可以是各自的存取權（“accessRight”）資源的集合；其每一個可包括、維護和/或儲存許可的表示。accessRights 子資源 314 的每個 accessRight 資源可與對主機 SCL 外部的實體來說是可存取的一個或多個其他子資源相關聯。accessRights 314 的每一個的許可表示可包括許可持有者的標識和授予該許可持有者的許可權的標識。授予該許可持有者的許可權的標識可以是例如與為相應子資源授予的一個或多個許可權相關聯的許可標誌。

訂閱子資源 316 可以是各自的訂閱資源的集合。訂閱 316 的每個訂閱資源可包括用於跟蹤對其父代資源（即 <sclBase>302）的（例如啟動）訂閱狀態的資訊。訂閱 316 的每一個可表示來自發佈者在<sclBase>302 上的修改通知的請求。

發現 318 可被用來允許子資源的發現。發現 318 可被用來獲取匹配一發現過濾標準的子資源的統一資源識別字（“URI”）的列表。accessStatus 320 可以是各自的存取狀態資源的集合。

mgmtObjs 子資源 322 可以是各自的管理物件（“mgmtObj”）資源的集合。mgmtObjs 322 的每個 mgmtObj 資源可包括、維護和/或儲存用於執行 REM 的管理資訊和/或參數。mgmtObjs 子資源 322 可包括應用管理物件子資源（“appMgmtObjects”）子資源 324、SCL 管理物件子資源（“sclMgmtObjects”）子資源 326、網路管理物件子資源

(“nwkMgmtObjects”) 子資源 328、裝置管理物件子資源 (“devMgmtObjects”) 子資源 330、OMA-DM 管理物件子資源 (“omaMgmtObjects”) 子資源 332 和 BBF-TR069 管理物件子資源 (“bbfMgmtObjects”) 子資源 334。mgmtObjs 子資源 322 還可包括其他和/或不同 mgmtObj 資源。

appMgmtObjs 子資源 324 可以是各自的應用管理物件 (“appMgmtObject”) 資源的集合。每個 appMgmtObject 資源可包括用於根據應用管理層及其功能 (例如應用管理層 202 (第 2 圖)) 來執行 REM 的資訊和/或參數。每個 appMgmtObject 資源可被部署為作為 <mgmtObject> 實例 324-1 而從屬於 appMgmtObjects 子資源 324。

sclMgmtObjects 子資源 326 可以是各自的 SCL 管理物件 (“sclMgmtObject”) 資源的集合。每個 sclMgmtObject 資源可包括用於根據服務管理層及其功能 (例如服務管理層 204 (第 2 圖)) 來執行 REM 的資訊和/或參數。每個 sclMgmtObject 資源可被部署為作為 <mgmtObject> 實例 326-1 而從屬於 sclMgmtObjects 子資源 326。

nwkMgmtObjects 子資源 328 可以是各自的網路管理物件 (“nwkMgmtObject”) 資源的集合。每個 nwkMgmtObject 資源可包括用於根據網路管理層及其功能 (例如網路管理層 206 (第 2 圖)) 而執行 REM 的資訊和/或參數。每個 nwkMgmtObject 資源可被部署為作為 <mgmtObject> 實例 328-1 而從屬於 nwkMgmtObjects 子資源 328。

devMgmtObjs 子資源 330 可以是各自的裝置管理物件 (“devMgmtObject”) 資源的集合。每個 devMgmtObject 資源可包括用於根據裝置管理層及其功能 (例如裝置管理層 208 (第 2 圖)) 而執行 REM 的資訊和/或參數。每個 devMgmtObjects 資源可被部署為作為 <mgmtObject> 實例

330-1 而從屬於 devMgmtObjects 子資源 330。

omaMgmtObjects 332 可以是各自的 OMA-DM 管理物件（“omaMgmtObject”）資源的集合。每個 omaMgmtObject 資源可包括用於根據 OMA-DM 和/或 OMA-DM 相容管理功能而執行 REM 的資訊和/或參數。每個 omaMgmtObject 資源可被部署為作為 <mgmtObject> 實例 332-1 而從屬於 omaMgmtObjects 子資源 332。

bbfMgmtObjects 子資源 334 可以是各自的 BBF-TR069 管理物件（“bbfMgmtObject”）資源的集合。每個 bbfMgmtObject 資源可包括用於根據 BBF-TR069 和/或 BBF-TR069 相容管理功能（例如 BBF-TR069 遠端程序呼叫（RPC）方法）而執行 REM 的資訊和/或參數。每個 bbfMgmtObject 資源可被部署為作為 <mgmtObject> 實例 334-1 而從屬於 bbfMgmtObjects 子資源 334。

雖然如第 3A 圖所示 mgmtObjs 子資源 322 僅存在於 <sclBase-of-Server>/scls/<scl>/mgmtObjs，但其他（例如多個）mgmtObjs 子資源可位於 <sclBase>302 的各種更遠的從屬分支/位置中。這樣，這樣的其他 mgmtObjs 子資源可明確地相應於具體的管理功能（例如相應於應用或 SCL）。

第 3B 圖是示出用於根據管理層集合及其功能向 SCL 提供資源結構的示例資源結構構架 340 的框圖。在其上可提供這樣的資源結構的 SCL 可以是本地 SCL，例如 G-SCL 56a、56b 和/或 D-SCL 44a-44d 的任一個。在本地 SCL 上提供的資源結構可以本地 SCL 和/或主機 SCL 間的同步方式，在一個或多個其他主機 SCL 和/或一個或多個遠端 SCL 上隨後提供（例如整體或部分地複製）。替代地，在其上可提供這樣的資源結構的 SCL 可以是主機 SCL，例如 N-SCL 26a-26c 的任一個。在主機 SCL 上提供的資源結構可以主機 SCL 和

遠端 SCL 間的同步方式，在一個或多個其他遠端 SCL（例如 G-SCL 56a、56b 和/或 D-SCL 44a-44d）上隨後提供（例如整體或部分地複製）。附加地和/或替代地，資源結構構架 340 還可被用於在多個本地 SCL 和/或多個主機 SCL 上提供資源結構。

資源結構構架 340 可包括<sclBase>342、<sclBase>342 的複數子資源、與直接從屬於<sclBase>342 的子資源的一些或全部相關聯的一個或多個屬性 344、和與間接從屬於<sclBase>342 的子資源相關聯的屬性 350 和 360。

除了以下的，直接從屬於<sclBase>342 的子資源類似於第 3A 圖的直接從屬於<sclBase>302 的子資源。直接從屬於<sclBase>342 的子資源可包括應用子資源 346 和 mgmtObjs 子資源 348。

mgmtObjs 子資源 348（在<sclBase>/mgmtObjs 處）可以是各自的管理物件（“mgmtObj”）資源的集合。mgmtObjs 子資源 348 可包括、維護和/或儲存用於根據（i）服務管理層及其功能；（ii）網路管理層及其功能；（iii）裝置管理層及其功能，（iv）OMA-DM 和/或 OMA-DM 相容管理功能和（v）BBF-TR069 和/或 BBF-TR069 相容管理功能的任一個而執行 REM 的管理資訊和/或參數。mgmtObjs 子資源 348 例如可包括 sclMgmtObjects 子資源 326、nwkMgmtObjects 子資源 328、devMgmtObjects 子資源 330、omaMgmtObjects 子資源 332 和 bbfMgmtObjects 子資源 334。mgmtObjs 子資源 348 還可包括其他和/或不同 mgmtObj 資源。

應用子資源 346 可包括各自的應用（“<application>”）資源 352 的集合、accessStatus 子資源 354、subscriptions 子資源 356、mgmtObjs 子資源 358 和與應用子資源 346 的子資源相關聯的屬性 350。mgmtObjs 子資源 358（位於

<sclBase>/applications/mgmtObjs) 可包括用於執行作為整體註冊在<sclBase>342 下所有應用的 REM 的子資源的集合。

各自的應用資源 352 的每一個可包括 containers 子資源 362、groups 子資源 364、accessRights 子資源 366、accessStatus 子資源 368、subscriptions 子資源 370、mgmtObjs 子資源 372 和與相應的各自的應用資源 352 的子資源相關聯的屬性 360。mgmtObjs 子資源 372 (位於 <sclBase>/applications/<application>/mgmtObjs) 可包括用於執行與相應 <application> 子資源 352 相關聯的特定 <application> 的 REM 的子資源集合。

第 3C 圖是示出根據管理層集合及其功能向 SCL 提供資源結構的示例資源結構框架 376 的框圖。可在其上提供這樣的資源結構的 SCL 可以是主機 SCL，例如 N-SCL 26a-26c 的任一個。在主機 SCL 上提供的資源結構可以主機 SCL、其他主機 SCL 和/或遠端 SCL 間的同步的方式，在一個或多個其他主機 SCL 和/或一個或多個遠端 SCL 上(例如 G-SCL 56a、56b 和/或 D-SCL 44a-44d)隨後提供(例如整體或部分複製)。替代地，在其上可提供這樣的資源結構的 SCL 可以是遠端 SCL，例如 G-SCL 56a、56b 和/或 D-SCL 44a-44d 的任一個。在遠端 SCL 上提供的資源結構可以遠端 SCL、其他遠端 SCL 和/或主機 SCL 間的同步的方式，在一個或多個其他遠端 SCL 和/或一個或多個主機 SCL 上(例如 N-SCL 26a-26c)後續提供(例如整體或部分複製)。附加地和/或替代地，資源結構架構 340 也可被用於在多個主機 SCL 和/或多個遠端 SCL 上提供資源結構。

資源結構架構 376 可包括<sclBase> 378、<sclBase> 378 的多個子資源、與直接和/或間接從屬於<sclBase> 378 的子資源的一些或全部相關聯的一個或多個屬性。除了沒有

mgmtObjs 子資源直接從屬於<sclBase> 378 之外，直接從屬於<sclBase> 378 的子資源類似於第 3A 圖的直接從屬於<sclBase> 302 的子資源。反而，資源結構架構 376 包括多個 mgmtObjes 子資源，其可位於<sclBase> 376 的各種更遠的從屬分支/位置處。例如，<sclBase> 376 可包括 mgmtObjs 子資源 380（在<sclBase>scls/mgmtObjs 處）。mgmtObjs 子資源 380 可包括用於執行作為整體向 M2M 伺服器註冊的所有 SCL 的 REM 的子資源的集合。這些子資源可包括用於根據服務管理層及其功能（例如服務管理層 204（第 2 圖））而執行作為整體向 M2M 伺服器註冊的所有 SCL 的 REM 的資訊和/或參數。

<sclBase> 376 還可包括 mgmtObjs 子資源 382（在<sclBase>/scls/<scl>/mgmtObjs 處）。mgmtObjs 子資源 382 可包括用於執行向 M2M 伺服器註冊的<scl>的服務能力和其他管理功能（網路管理（mgmt）層和裝置管理層）的 REM 的子資源集合。在一個實施方式中，子資源可包括根據（i）網路管理層及其功能（例如網路管理層 206（第 2 圖））；和（ii）裝置管理層（例如裝置管理層 208（第 2 圖））而用於執行對 M2M 伺服器註冊的<scl>的服務能力和其他管理功能的 REM 的資訊和/或參數。

<sclBase> 376 還可包括 mgmtObjs 子資源 384（在<sclBase>/scls/<scl>/applications//mgmtObjs 處）。mgmtObjs 子資源 384 可包括用於執行作為整體向伺服器宣告的所有應用的 REM 的子資源的集合。<sclBase> 376 可進一步包括 mgmtObjs 子資源 386（在<sclBase>/scls/<scl>/applications/<applicationAnnc>/mgmtObjs 處）。此 mgmtObjs 子資源 386 可包括用於執行對 M2M 伺服器宣告的特定<applicationAnnc>的 REM 的子資源集合。

在一個實施方式中，mgmtObjs 子資源 382（在 <sclBase>/scls/<scl>/mgmtObjs 處）可被 DA 和/或 GA 使用，以管理向 M2M 伺服器註冊的另一個 D/G。M2M 伺服器（即 <scl>）可向 D/G 宣告它的<mgmtObj>。然後 DA/GA 可在 D/G 中存取這樣的經宣告的<mgmtObj>，並且反過來，能夠經由 M2M 伺服器處的訊息中繼來管理其他 D/G。

第 4A 圖是示出用於向 SCL 提供 mgmtObjs 的示例資源結構框架 400 的框圖。資源結構框架（“mgmtObjs 結構框架”）400 可包括作為根的 mgmtObjs 402、mgmtObjs 402 的複數子資源、與直接從屬於 mgmtObjs 402 的子資源的一些或全部相關聯的一個或多個屬性 404 以及與間接從屬於 mgmtObjs 402 的子資源相關聯的屬性 416、418、430 和 442。屬性 404 可包括存取權 ID、創建時間、最後修改時間、描述（例如 mgmtObjs 402 的文本格式描述）和允許的方法。允許的方法可指定用於處理 mgmtObjs 子資源 402 的允許的 RESTful 方法。

mgmtObjs 402 的複數子資源可包括 mgmtObj（“<mgmtObj>”）子資源 406、管理物件宣告（“<mgmtObjAnnc>”）子資源 408、accessRights 子資源 410、accessStatus 子資源 412 和 subscriptions 子資源 414。

<mgmtObj>子資源 406 可以是具體的管理物件和用於儲存此<mgmtObj>子資源 406 的相關管理資料/參數的佔位（placeholder）。<mgmtObjAnnc>子資源 408 可以用於經宣告的管理物件的佔位。<mgmtObjAnnc>子資源 408 可包括以下屬性：(i) 鏈路，(ii) 存取權 ID 和 (iii) 搜索字串。

<accessRights>子資源 410 可包括、維護和/或儲存用於用來執行 REM 的子資源的許可的表示。<accessRights>子資源 408 可繼承自它的父代，如果有的話。

`accessStatus` 子資源 412 可以是各自的存取狀態資源的集合。`subscriptions` 子資源 414 可以是訂閱資源的集合，它的每一個可包括用於跟蹤對其父代資源的（例如啟動的）訂閱的狀態的資訊。訂閱 414 的每一個可表示來自發佈者用於對父代資源的修改通知的請求。

`<mgmtObj>`子資源 406 可包括(i) `<parameters>`子資源 420，其可以用於管理目的的多個參數的集合的佔位；(ii) `<parameter>`子資源 422，其可以是單一的管理參數；(iii) 用於 xREM 目的的`<accessRights>`子資源 424；(iv) `<accessStatus>`子資源 426，和 (v) `<subscriptions>`子資源 428。`<accessRights>`子資源 424 可包括、維護和/或儲存用於用來執行 REM 的子資源的允許的表示。`<accessRights>`子資源 424 可繼承自它的父代，如果有的話。

`<mgmtObj>`子資源 406 可包括以下屬性：(i)存取權 ID、創建時間、最後修改時間、描述（例如`<mgmtObj>`子資源 406 的文本格式描述）、允許的方法和內容類型。允許的方法可指定用於處理`<mgmtObj>`子資源 406 的允許的 RESTful 方法。內容類型可指定`<mgmtObj>`子資源 406 的類型。內容類型屬性也可被稱為資料類型屬性。

`<parameters>`子資源 420 可包括`<parameters>`子資源 432。此`<parameters>`子資源 432 可以用於管理目的的多個參數的集合的佔位。經由使`<parameters>`子資源 432 從屬於`<parameters>`子資源 432，可支援階層式樹狀結構，並且現有的管理物件的輸入可比扁平結構更容易。

`<parameters>`子資源 420 還可包括(i) 用於維護和/或儲存單一管理參數的`<parameter>`子資源 434；(ii) 用於 xREM 目的的`<accessRights>`子資源 436；(iii) `<accessStatus>`子資源 438；和`<subscriptions>`子資源 440。`<accessRights>`子資

源 436 可包括、維護和/或儲存用於用來執行 REM 的子資源的允許的表示。<accessRights>子資源 436 可繼承自它的父代，如果有的話。

<parameters>子資源 420 可包括以下屬性：(i) 存取權 ID，(ii) 創建時間，(iii) 最後修改時間，(iv) 描述（例如 <parameters>子資源 420 的文本格式描述），(v) 允許的方法，和 (vi) 內容類型/資料類型。允許的方法可指定用於處理<mgmtObj>子資源 436 的允許的 RESTful 方法。內容類型/資料類型可指定<mgmtObj>子資源 436 的類型。

另外，如第 4A 圖的示例所示，mgmtObjs 402 經由使另一個<parameters>子資源 432 從屬於<parameters>子資源 420 而具有增加階層式結構。在此子結構中，<parameters>子資源 420 可包括若干個<parameters>子資源 432 和各自的<parameter>資源 434。這樣的階層式結構簡化了向 mgmtObjs 402 引入其他管理樹，和/或簡化了執行樹結構映射。應注意，這樣的階層式結構沒有<mgmtObj>子資源 406 和其子代（例如使用現有的容器或組資源）可能無法被實現，除非一個容器（或組）資源適用於允許以另一個容器（或組）來創建和使用內容實例（或成員），作為其子資源。

<parameter>資源 434 可包括 (i) 可包括、維護和/或儲存<parameter>子資源 434 預設值的<defaultValue>子資源 444；(ii) 可包括、維護和/或儲存<parameter>子資源 434 的現行值的<currentValue>子資源 446；(iii) 用於 xREM 目的的<accessRights>子資源 448；(iv) <accessStatus>子資源 450，和 (v) <subscriptions>子資源 444。<accessRights>子資源 448 可包括、維護和/或儲存用於用來執行 REM 的子資源的允許的表示。<accessRights>子資源 448 可繼承自它的父代，如果有的話。

<parameter>資源 434 可包括以下屬性：(i) 存取權 ID，(ii) 創建時間，(iii) 最後修改時間，(iv) 描述（例如 <parameter>資源 434 的文本格式描述）和允許的方法。允許的方法可指定用於處理 <parameter>資源 434 的允許的 RESTful 方法。

在替代的實施方式中，用於向 SCL 提供 mgmtObjs 的兩級資源結構框架 458 在第 4B 圖中顯示。資源結構框架 458 可包括 mgmtObjs 460。除了 mgmtObjs 460 不包括 <parameters>子資源 420 之外，mgmtObjs 460 類似於第 4A 圖的 mgmtObjs 402。

示例 M2M xREM 管理模型

在一個實施例中，xREM 可在用戶端/伺服器（C/S）模式下實現。附加地和/或替代地，xREM 可使用代理功能來管理在 M2M GW 後面的 M2M 裝置，例如 D 34d 和 D' 34e-34（第 1 圖）。

在一個實施方式中，xREM 伺服器可以是管理控制器。xREM 伺服器可例如操作為或具有類似於 SNMP 管理器、根據 OMA DM 的 DM 伺服器和根據 BBF-TR069 的 ACS 的任一個的功能。xREM 伺服器可控制和管理與 xREM 用戶端和 xREM 代理的交互作用。

在一個實施方式中，xREM 用戶端可以是由 xREM 伺服器控制的軟體實體。xRE 用戶端可以例如作為或具有類似於 SNMP 代理、根據 OMA DM 的 DM 用戶端和根據 BBF TR-069 的 CPE 的任一個的功能。xREM 伺服器和 xREM 用戶端可一起工作以建立管理對話和執行管理功能。

在一個實施方式中，xREM 代理可扮演 xREM 用戶端和 xREM 伺服器的角色。xREM 代理可具有用於管理 D'類型的裝置（例如根據 OMA DM 和/或 BBF TR-069 操作的 M2M

裝置和/或無 SCL 的 M2M 裝置) 的非 xREM 管理伺服器功能。xREM 代理可包括在 xREM 用戶端和 xREM 伺服器(或非 xREM 管理伺服器)之間的轉換和/或適應功能。示例功能可包括在 xREM 用戶端和 xREM 伺服器之間的轉換,和在 xREM 用戶端和非 xREM 管理伺服器之間的轉換。

在一個實施方式中,xREM 取決於它所在之處,可以是 xREM 伺服器(管理器)、xREM 用戶端(受管理的實體)或 xREM 代理(作為管理器及受管理的實體)。

第 5 圖是示出用於執行 xREM 的用戶端-伺服器模型 500 的圖的框圖。xREM 可根據 ETSI TS 102 960 而執行。參考第 5 圖,NREM 502 可包括 xREM 伺服器 502-1,其執行與一個或多個 xREM 用戶端(例如 DREM 504 和 GREM 506 各自的 xREM 用戶端 504-1 和 506-1 的任一個)或 GREM 510 的 xREM 代理 510-1 的通信交互作用。NREM 502 可具有到第三方管理機構(authority) 512 和例如 NREM 508 的其他 NREM 的分離介面。GREM 510 可操作為 xREM 用戶端 510-2 或 xREM 代理 510-1。當 M2M 閘道操作為要由 M2M 伺服器管理的端裝置時,GREM 510 可操作為 xREM 用戶端。當 M2M 閘道操作為用於 M2M 伺服器、以管理 M2M 閘道後面的 M2M 裝置(ETSI 相容或非 ETSI)的代理時,GREM 510 操作為 xREM 代理。作為 xREM 代理 510-1,GREM 510 可包括 xREM 用戶端 510-2、xREM 伺服器 510-3、非 xREM 管理伺服器 510-4 和協定轉換單元 510-5。DREM 512 可包括 xREM 用戶端 512-1。xREM 用戶端 512-1 可與 NREM 502 中的 xREM 伺服器 502-1 或替代地與 xREM 代理 510-1 中的 xREM 伺服器 510-3 交互作用。

如果 OMA DM 被用以執行 xREM,xREM 伺服器 502-1 可以是 DM 伺服器,並且 xREM 用戶端 512-1 可以是 DM 用

戶端。xREM 代理 510-1 可以是 OMA GwMO。如果 BBF TR-069 被用以執行 xREM，xREM 伺服器 502-1 可以是 ACS，並且 xREM 用戶端 512-1 可以是 CPE。如果 SNMP 被用來執行 xREM，xREM 伺服器 502-1 可以是 SNMP 管理器，並且 xREM 用戶端 512-1 可以是 SNMP 代理。xREM 代理 510-1 可以是 SNMP 代理。

支援多管理協定

根據一個實施方式，整合的 M2M 系統（例如系統 10）可包括多個垂直的 M2M 應用，其中可部署不同的管理協定。例如，支持 DM 用戶端作為其 xREM 用戶端的 M2M 裝置可移動並與僅支援 BBF TR-069 的 M2M GW 或與僅管理 BBF TR-069 裝置的 M2M 伺服器相連接。在另一個替代選擇中，根據本文的實施方式，其他適當 OMA DM 裝置和 BBF TR-069 裝置可在整合的 M2M 系統中管理。

第 6 圖是示出使用多個不同管理協定以支援 xREM 的以通道為基礎的方法的框圖。NREM 602 和/或 GREM 604 可使用不同的管理協定來管理與不同 M2M GW 和裝置的交互作用。通道模組可執行以下說明。通道模組可執行在 D/GREM 和 NREM 之間和/或在 GREM 和 DREM 之間的協商，以確定使用哪個管理協定。通道模組還可根據該確定引導 xREM 軟體更新。

通道模組可執行資料模型轉換，以在 xREM 資料模型和 OMA DM mgmt 物件、BBF TR-069 管理參數和/或 SNMP MIB 之間轉換。通道模組還執行 OMA DM 命令或 BBF TR-069 命令和 xREM RESTful 方法之間的管理命令轉換和/或映射。通道模組還可執行協定適配以使 OMA DM、BBF TR-069 或 SNMP 協定適用於能夠使用 mId 參考點上的 RESTful 方法。

用於協商或指示管理協定的方法

被稱為例如 `mgmtProtocolType` 的參數可被用以表示管理協定的類型。`mgmtProtocolType` 可以是資源<scl>的屬性（或子資源），其可以是 M2M 裝置、M2M 閘道或 M2M 伺服器的 SCL。此外，`mgmtProtocolType` 可被包括在 M2M SCL 管理物件（稱為“SCLMO”）中而作為配置參數。在一個實施方式中，M2M 裝置可使用“`mgmtProtocolType`”以指示在 M2M 裝置和 M2M 伺服器之間或在 M2M 裝置和 M2M 閘道之間使用的管理協定類型。在一個實施方式中，M2M 閘道可使用“`mgmtProtocolType`”以表示在 M2M 閘道和 M2M 伺服器之間使用的管理協定類型。為了促進此事，M2M 閘道的<scl>可包括屬性（或子資源）“`mgmtProtocolType`”並且 M2M 閘道的 SCLMO 可包括參數“`mgmtProtocolType`”。如果 M2M 裝置在 M2M 閘道後面存在，第二屬性或（子資源）“`mgmtProtocolTypeExt`”可被用以表示在 M2M 閘道後面的 M2M 裝置與 M2M 閘道之間的管理協定類型。

在一個實施方式中，M2M 伺服器可使用“`mgmtProtocolType`”以表示它支持的用於管理 M2M 裝置和/或 M2M 閘道的管理協定類型。為了促進此事，M2M 伺服器的<scl>可包括屬性（或子資源）“`mgmtProtocolType`”。

在一個實施方式中，N-SCL（和/或 G-SCL）可包括表示相應 NREM 和/或 GREM 支援的多個管理協定的列表的屬性（或子資源）`mgmtProtocolType`。

作為一個替代選擇，`mgmtProtocolType` 可被增加為 `mgmtObjs` 資源的一個屬性和/或每個<mgmtObj>實例的一個屬性。經由使用 `mgmtProtocolType` 用作<scl>的屬性、`mgmtObjs` 資源的屬性或<mgmtObj>資源的屬性，以下方法可應用於協商或指示 DREM 和/或 GREM（“D/GREM”）與

NREM 之間和/或 DREM 與 GREM 之間的管理協定。

第 7A 圖-第 7C 圖是分別示出用於確定用於 REM 的管理協定的類型的示例流程 700、730 和 760 的流程圖。為了方便，參考第 1A 圖-第 1C 圖的系統 10 來描述流程 700、730 和 760 的每一個。流程 700、730 和 760 還可使用其他架構來實現。

在 D/GREM 和 NREM 之間

方法 1 - 在“SCL 註冊”和/或“更新 SCL 註冊”中攬負 (piggyback) mgmtProtocolType

現在參考第 7A 圖的流程圖 700，D/GREM 可發起 SCL 註冊過程 (702)，在此期間 D/GREM 可向 M2M 伺服器 (NREM) 發送一個或多個請求 (“SCL 註冊請求”) 訊息。作為 SCL 註冊過程 (702) 的一部分、之前和/或之後，D/GREM 可從 D/G-SCL 的“mgmtProtocolType”屬性/子資源獲得指示 D/GREM 支援的管理協定類型的值 (“mgmtProtocolType (值)”)。

M2M 裝置/閘道 (D/GREM) 然後可選取並將 mgmtProtocolType (值) 用作其參數而填充一個或多個 SCL 註冊請求訊息，然後將這樣的 SCL 註冊請求 [mgmtProtocolType (值)] 訊息發送給 M2M 伺服器 (NREM) (704)。響應於 SCL 註冊請求 [mgmtProtocolType (值)] 訊息，M2M 伺服器 (NREM) 可選取並將接收的 mgmtProtocolType (值) 作為其參數而填充一個或多個響應 (“SCL 註冊響應”) 訊息。此後，M2M 伺服器 (NREM) 可向 M2M 裝置/閘道 (D/GREM) 發送這樣的 SCL 註冊響應 [mgmtProtocolType (值)] 訊息，以確認 mgmtProtocolType (值) 的接收和/或由接收的 mgmtProtocolType 指示的管理協定類型的接收 (706)。在接收到 SCL 註冊響應

[mgmtProtocolType (值)] 訊息和用於完成 SCL 註冊過程的其他訊息（如果有的話）後，D/GREM 可終止 SCL 註冊過程（708）。經由執行流程 700，D/GREM 可在 SCL 註冊過程上揹負 mgmtProtocolType 屬性/子資源，以促進在執行 REM 時 M2M 伺服器(NREM)和 M2M 裝置/閘道(D/GREM)可使用的管理協定類型的通知和/或接收。

儘管未示出，M2M 伺服器（NREM）可不經請求地在 SCL 註冊過程上揹負 mgmtProtocolType 屬性/子資源，以指示 M2M 裝置/閘道（D/GREM）來使用由 N-SCL 的 mgmtProtocolType 屬性/子資源所指定或指示的管理協定類型。M2M 伺服器（NREM）例如可從 N-SCL 的 mgmtProtocolType 屬性/子資源獲取 mgmtProtocolType (值)，將該 mgmtProtocolType (值) 作為其參數而填充一個或多個 SCL 註冊響應訊息，並向 M2M 裝置/閘道（D/GREM）發送經填充的 SCL 註冊響應[mgmtProtocolType (值)] 訊息。

作為一個替代選擇，M2M 伺服器（NREM）和 M2M 裝置/閘道（D/GREM）可經由交換 SCL 註冊請求訊息和將各自的 mgmtProtocolType (值) 填充的 SCL 註冊響應訊息來協商管理協定的類型，直到 M2M 伺服器（NREM）或 M2M 裝置/閘道（D/GREM）向另一個發送它接收到的 mgmtProtocolType (值)。

第 7B 圖-第 7C 圖，用於在更新 SCL 註冊中揹負 mgmtProtocolType 的過程如下：

在更新 SCL 註冊過程期間，M2M 伺服器（NREM）可經由在發送給 M2M 裝置/閘道（D/GREM）的訊息中揹負“mgmtProtocolType”而指定將由 M2M 裝置/閘道(D/GREM)使用的“mgmtProtocolType”。

替代地，在“更新 SCL 註冊”的過程期間，M2M 裝置/閘道 (D/GREM) 可經由在發送給 M2M 伺服器 (NREM) 的該訊息中揹負“mgmtProtocolType”而向 M2M 伺服器 (NREM) 報告它們的“mgmtProtocolType”。

方法 2 - 在“創建 mgmtObj 資源”中揹負 mgmtProtocolType

ETSI M2M 功能架構定義了以下用於創建管理物件資源的過程。結果，mgmtProtocolType 可被嵌入那些過程中，以便 M2M 裝置/閘道 (D/GREM) 可向 M2M 伺服器 (NREM) 通知它的 mgmtProtocolType。

如第 8 圖所示，用於在創建 mgmtObj 資源中揹負 mgmtProtocolType 的過程如下：

如果此過程由 M2M 裝置/閘道 (D/GREM) 開始，M2M 裝置/閘道 (D/GREM) 在此過程期間將向 M2M 伺服器 (NREM) 發送請求訊息。M2M 裝置/閘道 (D/GREM) 可在此請求訊息中將“mgmtProtocolType”揹負為參數。

方法 3 - 創建用於發送“mgmtProtocolType”的新過程

與如在方法 2 和方法 3 中所述的在現有的 M2M 過程中揹負“mgmtProtocolType”相異的是，以下過程定義了用於在 M2M 伺服器 (NREM) 和 M2M 裝置/閘道 (D/GREM) 之間發送“mgmtProtocolType”的過程。如第 9 圖所示，該過程如下：

更新“mgmtProtocolType”

M2M 裝置 / 閘道 (D/GREM) 發送定址到 <scl-of-server>/scls/<scl-dg>/mgmtProtocolType 的更新訊息以更新“mgmtProtocolType”。

<scl-of-server>表示 M2M 伺服器。<scl-dg>表示用於更新其 mgmtProtocolType 的 M2M 伺服器的 M2M 裝置/閘道

(D/GREM)。“mgmtProtocolType”可以是<scl-dg>的新屬性。結果，M2M 伺服器 (NREM) 獲知向該 M2M 伺服器註冊的 M2M 裝置/閘道 (D/GREM) 的“mgmtProtocolType”。

NREM 還主動發送定址到<scl-dg>/mgmtProtocolType 的更新訊息以改變 M2M 裝置/閘道的“mgmtProtocolType”，如第 10 圖所示。

作為一個替代選擇，NREM 可發送定址到<scl-dg>/mgmtProtocolType 的獲取訊息，以獲取在 M2M 裝置/閘道中使用的當前管理協定。

方法 4 - 在 SCL 發現中揹負 mgmtProtocolType

mgmtProtocolType 還可使用以下方法在用於 SCL 發現的訊息中揹負。如第 11 圖所示，當 M2M 裝置/閘道 (D/GREM) 發佈請求訊息以發現 M2M 伺服器 (NSCL) 時，M2M 裝置/閘道將在請求訊息中包括其 mgmtProtocolType。所揹負的 mgmtProtocolType 資訊可幫助濾除不支援由該 mgmtProtocolType 表示的管理協定的 M2M 伺服器。

M2M 伺服器的 mgmtProtocolType 可在到 M2M 裝置/閘道的 SCL 發現響應訊息中揹負。所揹負的 mgmtProtocolType 可幫助 M2M 裝置/閘道選擇適當的 M2M 伺服器。M2M 裝置/閘道還可以改變來使用由所揹負的 mgmtProtocolType 表示的管理協定。

在 DREM 和 GREM 之間

在此情況下，M2M 裝置在作為代理的 M2M 閘道的後面。結果，M2M 裝置需要向 M2M 閘道指示它們的 mgmtProtocolType，或者 M2M 閘道可主動獲取 M2M 裝置的 mgmtProtocolType。以上用於情況 1 的類似方法可以被使用。

方法 1 - 在“SCL 註冊”和“更新 SCL 註冊”中揷負 **mgmtProtocolType**

ETSI M2M 功能架構定義了用於 SCL 管理的以下過程，而且結果，**mgmtProtocolType** 可被嵌入在那些過程中，以便 DREM 可向 GREM 通知它的 **mgmtProtocolType**。

SCL 註冊：

在此過程期間，DSCL 將向 NSCL 發送多個請求訊息。DREM 可在那些請求訊息的任一個中將“**mgmtProtocolType**”揷負為參數。此外，GREM 可在到 DREM 的響應訊息中揷負“**mgmtProtocolType**”，以指示 DREM 使用由“**mgmtProtocolType**”指定的管理協定。

更新 SCL 註冊

在“更新 SCL 註冊”的過程期間，GREM 可經由在發送給 DREM 的訊息中揷負“**mgmtProtocolType**”來指定將由 DREM 使用的“**mgmtProtocolType**”。在“更新 SCL 註冊”的過程期間，DREM 可經由在發送給 GREM 的訊息中揷負“**mgmtProtocolType**”向 GREM 報告它們的“**mgmtProtocolType**”。

方法 2 - 在“創建 **mgmtObj** 資源”中揷負 **mgmtProtocolType**

ETSI M2M 功能架構定義了用於創建管理物件資源的以下過程。結果，**mgmtProtocolType** 可被嵌入在那些過程中，以便 DREM 可向 GREM 通知它的 **mgmtProtocolType**。

創建 **mgmtObj** 資源：

如果此過程由 DSCL(DREM)發起，DSCL 將向 NGSCl 發送請求訊息。在此過程期間，DREM 可在該請求訊息中揷負“**mgmtProtocolType**”作為參數。

方法 3 - 創建用於發送“**mgmtProtocolType**”的新過程

與如在方法 2 和方法 3 中所述的在現有的 M2M 過程中揹負“mgmtProtocolType”相異的是，以下過程也可被用來在 GREM 和 DREM 之間發送“mgmtProtocolType”。

更新“mgmtProtocolType”

M2M 裝置 (DREM) 發送定址到 <scl-of-gw>/scls/<scl-d>/mgmtProtocolType 的更新訊息以更新“mgmtProtocolType”。<scl-of-gw>表示 M2M 閘道。<scl-d>表示向 M2M 閘道註冊的並正在用其 mgmtProtocolType 更新 M2M 閘道的 M2M 裝置。“mgmtProtocolType”是<scl-d>的屬性。結果，M2M 閘道 (GREM) 獲知向該 M2M 閘道註冊的 M2M 裝置 (DREM) 的“mgmtProtocolType”。

M2M 閘道 (GREM) 還可主動地發送定址到 <scl-d>/mgmtProtocolType 的更新訊息以改變 M2M 裝置的“mgmtProtocolType”。

獲取“mgmtProtocolType”

M2M 閘道 (GREM) 發送定址到 <scl-d>/mgmtProtocolType 的獲取訊息，以獲取在 M2M 裝置中使用的當前管理協定。

方法 4 - 在 SCL 發現中揹負 mgmtProtocolType

mgmtProtocolType 還可使用以下方法在用於 SCL 發現的訊息中揹負。

當 M2M 裝置 (DREM) 發佈請求訊息以發現 M2M 閘道 (GSCL) 時，M2M 裝置將在該請求訊息中包括它的 mgmtProtocolType。所揹負的 mgmtProtocolType 資訊可幫助過濾不支援由該 mgmtProtocolType 表示的管理協定的 M2M 閘道。

M2M 閘道的 mgmtProtocolType 可在到 M2M 裝置的 SCL 發現響應訊息中揹負。所揹負的 mgmtProtocolType 可

幫助 M2M 裝置選擇適當的 M2M 閘道。M2M 裝置還可以改變以使用由指負的 `mgmtProtocolType` 表示的管理協議。

用於所有存取歷史的示例資源定義

為了遠端實體管理的目的，對於接收器 SCL 來說，將請求者/發佈者-SCL 在接收器-SCL 的每個本地資源上具有的所有存取歷史加以記入將是重要的。雖然當前的 ETSI M2M 功能架構定義了資源存取狀態，但它僅記錄最後的存取（獲取/更新/訂閱） - 而不是過去在資源上的所有操作。為了便於存儲所有存取歷史，被稱為存取歷史（`accessHistories`）的新的資源定義在本發明中。在存取歷史上的相關操作也被定義。

資源存取歷史

基於存取狀態，存取歷史可被定義並如第 12 圖的示例所示，第 12 圖示出了資源存取歷史的示例結構。參考第 12 圖，存取歷史可包括以下元件：

- “屬性”可以具有存取權 ID
- 狀態：被用來表示它是否開始或結束記入存取歷史
- `<accessInstance>`：若干存取實例

每個`<accessInstance>`可具有以下子資源以記錄關於它的存取細節：

- 方法：在此存取中涉及的方法。它可以是創建、獲取、更新、刪除、訂閱或宣告。
- 請求者 ID：請求此存取的實體的 ID
- 資源 URI：此存取應該操作的資源的 URI
- 時間戳記：此存取發生時的時間
- 序號：代表存取順序的自動遞增的整數
- 結果：此操作的結果
- 如果方法是更新，它表示資源 URI 的新數值

○ 對於其他方法它可以是成功或失敗。其可能為故障管理定義不同類型的故障。

資源存取歷史可儲存在 M2M 裝置和 GW 處。只要在裝置/GW 的本地資源上有操作，存取實例可自動地得以創建並被加入存取歷史中。作為一個資源，存取歷史可由 NREM 出於管理目的而在許多情況下被存取。換句話說，NREM 可在 D/GREM 上創建/獲取/更新/刪除存取歷史。以下功能可得以實現：

- 創建：NREM 請求 D/GREM 在特殊“方法”和/或“資源 URI”上創建存取歷史一段特定時間間隔或永久。

- 獲取：NREM 從 D/GREM 獲取所有或一些 <accessInstance>

- 更新：NREM 請求 D/GREM 經由改變“狀態”的值而失效或恢復存取歷史功能

- 刪除：NREM 請求 D/GREM 刪除所有或一些 <accessInstance>

第 12 圖中所示的<accessInstance>是扁平結構，其可基於如第 13 圖所示的“方法”和/或如第 14 圖所示的“請求者 ID”被重構造，分別形成 2 級和 3 級階層式結構。第 13 圖示出了用於基於“方法”所構造的資源存取歷史的示例結構。第 14 圖示出了用於基於“方法”和“請求者 ID”所構造的資源存取歷史的示例結構。這樣的階層式結構對加速在資源存取歷史上的查詢/獲取操作是有用的，特別是在資源受限的 M2M 裝置處。

用於 xREM 管理許可授權的示例呼叫流程

M2M 裝置和/或 GW 的管理許可權可被授權給另一個 M2M 伺服器。M2M 裝置的管理許可權也可被授權給另一個 M2M GW。根據本發明的實施例，M2M 裝置和 GW 的管理許可權授權過程在本文描述。

管理許可權授權

授權者發起的授權

第 15 圖示出了用於管理許可權授權（授權者發起的）的過程的訊息流程圖。參考第 15 圖，N-SCL-1 將 D/G-SCL 授權給其他 N-SCL。N-SCL 1 在 D/G-SCL 上線後向 D/G-SCL 發佈“授權準備”。“授權準備”可被用以指示 D/G-SCL 為正在進行的授權操作做好準備。D/G-SCL 可向 N-SCL 1 發送回“授權響應”。D/G 之後可保持連線。N-SCL 1 可向被選作 D/G SCL 的新管理機構的 N-SCL X 發佈“授權請求”。“授權請求”可包括以下資訊：

- 要被授權的 D/G-SCL 的 URI 和/或授權相關資訊
- 請求管理授權的原因

N-SCL X 可經由對“授權請求”回答是或否來向 N-SCL 1 發送回“授權響應”。N-SCL 1 可向其他 N-SCL 重複發送“授權請求”，直到找到同意接受授權的 N-SCL 或在嘗試最大次數後。

N-SCL 1 可向 D/G-SCL 發佈“授權通知”。如果沒有 N-SCL X 同意，N-SCL 1 可使用此訊息來通知 D/G-SCL 取消正在進行的授權。然後 D/G 可如它平時那樣操作。它可以從此（onwards）開始休眠。整個授權過程然後可停止。否則，“授權通知”將包括關於同意接受授權的 N-SCL X 的資訊。D/G-SCL 可發送回“授權響應”。

N-SCL 1 向 N-SCL X 發佈“授權開始”以觸發它執行授權操作。

N-SCL X 可向 D/G-SCL 發佈“授權執行”。D/G-SCL 可在 N-SCL X 上進行授權。

D/G-SCL 可向 N-SCL X 發送回“授權響應”。D/G SCL 可經由將 N-SCL X 設定為它的新的管理機構來更新它的管理物件。N-SCL X 可在它的許可權下，經由包含 D/G-SCL

作為受管理的實體而更新它的管理物件。

N-SCL X 可向 N-SCL 1 發佈“授權結束”。

N-SCL 1 可向 N-SCL X 發送回“授權確認 (ACK)”。

N-SCL 1 可經由從它的管理許可權中移除 D/G-SCL 來更新它的管理物件。

N-SCL 1 可經由把 N-SCL X 增加為它向其授權管理許可權的實體來更新它的管理物件。

N-SCL X 可經由將 N-SCL 1 增加為它從其被授權為管理機構的實體來更新它的管理物件。

裝置發起的授權

第 16 圖示出了用於管理許可權授權（裝置發起的）的示例訊息流程圖。這可應用在 D/G 主動請求管理許可權授權的情境中。與授權者發起的授權的唯一不同是開始的兩個步驟。參考第 16 圖，D/G-SCL 向 N-SCL 1 發佈“授權請求”。D/G-SCL 可在此訊息中指示請求授權的原因。N-SCL 1 可經由通知 D/G-SCL 其進行授權的意願來向 D/G-SCL 發送回“授權響應”。N-SCL 1 可拒絕進行授權。然後整個授權過程可停止。

不同於第 16 圖並且替代地，D/G 可直接向被授權者（新的管理機構）發送“授權請求”。然後該被授權者可向授權者作出批准請求。示例詳細過程在第 17 圖中顯示，第 17 圖示出了管理許可權授權（裝置發起的直接到被授權者）的示例訊息流程圖。該情境可能需要授權者事先向裝置配置一些潛在被授權者，並且當裝置和授權者之間的通信遇到問題時可能發生情境。

被授權者發起的授權

除了授權者發起的和裝置發起的授權，管理許可權授權還可隨選地由被授權者發起。例如，第 18 圖示出了用於管

理許可權授權（被授權者發起的）示例訊息流程圖。參考第 18 圖，N-SCL X 可從 N-SCL 1 請求授權。N-SCL 1 可批准將 D/G-SCL 的管理許可權授權給 N-SCL X。每個步驟的含義與有關第 16 圖描述的那些步驟類似。

在作為代理的閘道下的授權

對於一些在閘道後面的裝置的情境，管理許可權授權可經由作為代理的閘道來實施。例如，第 19 圖示出了用於管理許可權授權（閘道作為代理）的示例訊息流程圖。參考第 19 圖，N-SCL-1 可經由作為代理的 G-SCL 將 D-SCL 授權給 N-SCL X。G-SCL 實際進行授權聚集。基本上，G-SCL 代表在它後面的 D 或 D'，可使用第 16 圖中類似的過程以 N-SCL 1 和 N-SCL X 實施管理許可權授權。當 D 或 D' 上線時，G-SCL 將通知它們授權結果，即新的管理機構 N-SCL X。

裝置到閘道授權

在一個實施例中，D-SCL 的管理許可權從 G-SCL 1 被授權給另一個 G-SCL X。例如，第 20 圖示出了用於管理許可權授權（裝置到閘道授權）的示例訊息流程圖。參考第 20 圖，在 D-SCL、G-SCL 1 和 G-SCL X 間的授權過程後，G-SCL 1 可通知 N-SCL 1 通知結果。

階層式 SCL 結構

第 21 圖示出了一般情境圖，其中：1) 所有 SCL 的管理許可權關係形成階層式結構；2) SCL 4 希望將其對 SCL 9 的管理許可權授權給 SCL 3。它基本上遵循 3 個步驟：

- 步驟 1：SCL 4 經由 SCL 2 和 SCL 1 逐跳發佈“授權請求”，最終到達 SCL 3。
- 步驟 2：SCL 3 向 SCL 4 發送回“授權響應”
- 步驟 3：SCL 4 向 SCL 9 發佈“授權通知”
- 步驟 4：SCL 3 和 SCL 9 實施“授權執行”相關的交互

作用。

● 步驟 5：SCL 3 經由 SCL 1 和 SCL 2 逐跳發送“授權通知”，最終到達 SCL 4。

○ 因此 SCL 1、SCL 2 和 SCL 4 可更新它們的管理物件而反映此管理許可權變化。

示例 xREM 許可權授權

NREM 負責管理 DREM 和 GREM。但是 ETSI xREM 既不描述遠端實體管理 (xREM) 許可權的概念，也不描述 xREM 許可權授權。出於例如 M2M 伺服器替換、負載平衡和移動性的原因，M2M 裝置或 GW 的 xREM 許可權需要從 M2M 伺服器被授權給另一個 M2M 伺服器。如果 M2M 裝置變成在 M2M GW 後面，M2M 裝置的 xREM 甚至可從 M2M 伺服器被授權到 M2M GW，反之亦然。

雖然 OMA-DM 為用戶端許可權授權定義了高級流程，但它沒有考慮休眠裝置或位於中間的 M2M GW。結果，它不能被直接應用於 ETSI M2M xREM。ETSI M2M xREM 需要有它自己的 xREM 許可權授權。

當 M2M 裝置或 GW 成功地向 M2M 伺服器註冊時，M2M 伺服器基本地處理對 M2M 裝置/GW 的 xREM 許可權。當 M2M 裝置在 M2M GW 後面時，該 GW 具有對那些 M2M 裝置的 xREM 許可權。因此，xREM 許可權授權可意指不同的情境：

情況 1：M2M 伺服器將其 xREM 許可權授權給另一個 M2M 伺服器

情況 2：M2M 伺服器將其 xREM 許可權授權給 M2M GW

情況 3：M2M GW 將其 xREM 許可權授權給 M2M 伺服器

情況 4：M2M GW 將其 xREM 許可權授權給另一個 M2M GW

xREM 的示例功能

結果，ETSI M2M xREM 需要具有新的功能來支援 xREM 許可權授權。網路遠端實體管理 (NREM) 需要具有以下功能：支援 xREM 許可權授權。NREM 具有對 M2M 裝置和 M2M 閘道的 xREM 許可權。NREM 需要具有以下功能來支援 xREM 許可權授權。將其對 M2M 裝置和 M2M 閘道的 xREM 許可權授權給另一個 M2M 伺服器或將其對 M2M 裝置的 xREM 許可權授權給 M2M 閘道。處理從另一個 M2M 伺服器或 M2M 閘道請求的 xREM 許可權授權。閘道遠端實體管理 (GREM) 需要具有以下功能：當作為一個經管理 M2M 區域網路的 M2M 裝置的遠端管理代理時。

支援 xREM 許可權授權。當 M2M 裝置在 M2M 閘道後面時，M2M 閘道具有對那些 M2M 裝置的 xREM 許可權。為了支援 xREM 許可權授權，M2M 閘道需要具有以下功能：將它對 M2M 裝置的 xREM 許可權授權於 M2M 伺服器或另一個 M2M 閘道；處理來自另一個 M2M 伺服器或 M2M 閘道的 xREM 許可權授權訊息；支援 xREM 許可權授權。M2M 伺服器具有對 M2M 閘道的 xREM 許可權。

M2M 閘道需要具有以下功能：主動請求 M2M 伺服器將其對 M2M 閘道的 xREM 許可權授權給另一個 M2M 伺服器；和被動地處理來自 M2M 伺服器的 xREM 許可權授權訊息。

裝置遠端實體管理 (DREM) 需要具有以下功能：支援 xREM 許可權授權。M2M 伺服器具有對 M2M 裝置的 xREM 許可權。

M2M 裝置需要具有以下功能：主動請求 M2M 伺服器

將其對 M2M 裝置的 xREM 許可權授權給另一個 M2M 伺服器；和被動地處理來自 M2M 伺服器的 xREM 許可權授權訊息。

示例非 RESTful 管理命令支援

在本文提供的是在其中非 RESTful 管理命令可以以 RESTful 方式表示和實現、用於 M2M xREM 的系統、設備和方法的實施方式。這樣的非 RESTful 管理命令的示例可包括以下命令的任一個：用於重新啟動裝置的重啟命令；用於指示接收下載命令來下載檔案的下載命令；用於執行特定過程的執行命令；用於將資源從一個位置複製和/或移動到另一個位置的副本命令。重啟和下載命令可以例如根據 BBF-TR069 來定義。執行和複製命令例如可以分別是 OMA-DM 的“Exec”命令和 OMA-DM 的“Copy”命令。非 RESTful 管理命令還可以包括其他命令，例如被用來控制裝備有這樣的可控元件的 M2M 裝置的可控元件（例如致動器）的一個或多個命令。

在一個或多個實施方式中，被稱為資源命令的資源可被用來表示非 RESTful 管理命令，並支援以由命令的一個或多個發佈者指定的不同方式執行非 RESTful 管理命令（“命令執行”）。資源命令可被用以在使用任意 RESTful 方法的裝置處促進非 RESTful 管理命令的發佈和命令執行。

另外，對於任何的非 RESTful 管理命令，資源命令可被用以促進立即命令執行、在（例如隨機的）延遲後的命令執行、單一命令執行（例如一次命令執行）和/或多次重複的命令執行。

參考第 22A 圖，框圖示出了具有表示資源命令的資料結構實例（“<commandInstance>”）的<sclbase>的示例資料結構。<commandInstance>可包括若干資料結構元素，如第

22A 圖所示的那些。表示屬性（““屬性””）的資料結構元素和表示子資源 `execMode`、`execParameters`、`execStartTime`、`execDuration`、`execResult`、`execStatus`、`requestorID` 和 `actorID` 的資料結構元素就在所示的這些資料結構元素之間。“屬性”可包括諸如先前討論過的存取權 ID（`accessRightID`）。

`execMode` 可被用以指定 `<commandInstance>` 的非 RESTful 管理命令的命令執行的特定模式。`execMode` 資料結構元素可包括表示用於命令執行的模式（“命令執行模式”）的一個或多個條目。命令執行模式的示例可包括以下的任一者：立即一次模式、立即和重複模式、隨機一次模式以及隨機和重複模式。

立即一次模式指定命令執行將立即發生並僅發生一次。立即和重複模式指定命令執行立即並重複地發生。在任意兩個命令執行之間的任意時間間隔可在 `execParameters` 中指定。隨機一次模式指定命令執行在（例如隨機的）延遲後發生並僅發生一次。延遲可在 `execParameters` 中指定。隨機和重複模式指定命令執行在（例如隨機的）延遲後重複地發生。延遲和任意兩個命令執行之間的任意間隔可在 `execParameters` 中指定。

`execParameters` 可以是容器，並且包括與命令執行模式相關聯的資訊。除了已經指出的資訊外，該資訊還可包括用於（1）指定在隨機一次和隨機重複模式中如何在命令執行前回退特定時間的資訊，和/或（2）指定在立即和重複模式以及隨機和重複模式中用於重複的命令執行的頻率的資訊。

`execStartTime` 可指定命令執行的實際持續時間。`execDuration` 可指定用於持續的命令執行所需的持續時間。`execResult` 可指定用於最後命令執行的返回值。`execStatus` 可指定命令執行的當前狀態。此狀態例如可以是以完成（成

功或失敗)、未決的、正在處理等。requestorID 可指定請求被調用的命令執行的請求者的 ID (例如 URI)。actorID 可指定調用命令執行的接收者的 ID (例如 URI)。

第 22B 圖是示出表示命令資源的示例資料結構 (“命令資源結構”) 的框圖。命令資源結構可包括若干資料結構元素, 例如第 22B 圖中所示的那些。並且此命令資源結構的許多資料結構元素類似於第 22A 圖的命令資源結構。定義屬性的資料結構元素 (“命令資源“屬性””) 和表示各自的命令 (“<command>”) 集合的資料結構元素組合在所示的這些資料結構元素中。<command>的每個實例作為資源表示單一的命令, 並且包括代表該<command>實例屬性和/或子資源的資料結構元素。這些<command>實例屬性和/或子資源可指定如何觸發和執行命令執行。代表 execEnable 子資源 (“execEnable”) 的資料結構元素在<command>實例屬性和/或子資源中。也可被稱為“執行”屬性的 execEnable 可促進調用命令執行狀態的改變。例如, execEnable 可以是布林變數, 並且該變數的特定值可調用命令執行、暫停命令執行、恢復命令執行和/或取消命令執行。這樣, 命令執行、暫停命令執行、恢復命令執行和取消命令執行的任一個可經由修改 execEnable 的值來調用。execEnable 的值可使用 RESTful 方法來修改。

例如, RESTful 方法更新 (UPDATE) 可被用以調用 (例如觸發) <command>實例的命令執行。經由將 RESTful 方法更新中的第一個值 (例如 “1”) 指定給 .../commands/<command>/execEnable, 而可調用命令執行。

RESTful 方法更新和刪除 (DELETE) 可被用以取消 (停止) <command>實例的命令執行。經由將 RESTful 方法更新

中的第二個值（例如“0”）指定給.../commands/<command>/execEnable，而可調用取消命令執行，以停止該<command>實例。附加地或替代地，發佈 RESTful 方法刪除.../commands/<command>/，取消命令執行可被調用，以在根據 RESTful 方法刪除這樣的<command>實例前停止該<command>實例的命令執行。

除了 execEnable，<command>實例子資源可包括代表例如 execBaseDelay、execAdditionalDelay、execFrequency、execNumber、execResource、execStatus、execResult、execIssuer 和 execParameters 的子資源的資料結構元素。

execBaseDelay 可在<command>實例的命令執行前指定最小延遲。execAdditionalDelay 可促進隨機附加延遲的生成。總延遲是 execBaseDelay 加上隨機附加延遲的和。execBaseDelay 和 execAdditionalDelay 可包括用於 (i) 指定如何在隨機一次和隨機重複模式中於命令執行前回退特定時間的資訊，和/或 (2) 指定在立即和重複模式以及隨機和重複模式中重複命令執行的頻率的資訊。

execFrequency 可指定用於<command>實例的重複命令執行的頻率。execNumber 可指定<command>實例的命令執行可重複的次數。execResource 可為<command>實例指定經過命令執行的資源 URI。該資源 URI 指向包括一組資源的組資源。這樣，多個<command>實例的命令執行可被調用為指向組資源的資源 URI 的函數。execResource 可以是隨選的。例如，如果<command>實例被從屬為在<command>實例下的子資源，execResource 可以是不必要的。

execStatus 可指定<command>實例的命令執行的當前狀態。該狀態例如可以是未決的、運行、暫停、停止、恢復、完成並成功、完成並失敗等。execResult 可儲存<command>

實例的命令執行的執行結果。如果生成多個結果（例如如果 `execResource` 指向一組資源），`execResult` 可模型化為 `<command>` 實例的子資源。

`execIssuer` 可指定發佈一請求來調用 `<command>` 實例的命令執行的發佈者的 ID。`execParameters` 可以用於儲存特定於 `<command>` 實例的參數的容器（例如佔位）。

現在參考第 22C 圖，其是示出另一個示例命令資源結構的框圖。此命令資源結構可包括若干資料結構元素，例如在第 22C 圖中所示的那些。並且此命令資源結構的許多資料結構元素類似於第 22B 圖的命令資源結構。定義若干屬性（即 `commandID` 屬性、`execDisable` 屬性、`execPause` 屬性、`execResume` 屬性和 `execResult` 屬性）的資料結構元素在第 22C 圖所示的資料結構元素間。

也可被稱為“執行”屬性的 `execEnable` 屬性可促進 `<command>` 實例的命令執行的調用。對 `execEnable` 屬性的 RESTful 方法更新可調用 `<command>` 實例的命令執行。RESTful 方法更新的酬載可以為空白的或被設定為一個值（例如 `execEnable=1`）。

也可被稱為“取消”屬性的 `execDisable` 屬性可促進 `<command>` 實例的取消命令執行的調用。對 `execDisable` 屬性的 RESTful 方法更新可調用 `<command>` 實例的取消命令執行。RESTful 方法更新的酬載可以為空白的或被設定為一個值（例如 `execDisable=1`）。

`execPause` 屬性可促進 `<command>` 實例的取消命令執行的調用。對 `execPause` 屬性的 RESTful 方法更新可調用 `<command>` 實例的暫停命令執行。RESTful 方法更新的酬載可以為空白的或被設定為一個值（例如 `execPause=1`）。

`execResume` 屬性可促進 `<command>` 實例的恢復命令執

行的調用。對 `execResume` 屬性的 RESTful 方法更新可調用這樣的恢復命令執行。RESTful 方法更新的酬載可以為空白的或被設定為一個值（例如 `execResume=1`）。

以上說明了第 23A、23B 和 23C 圖的資料結構經由使用 RESTful 方法來促進非 RESTful 管理命令的命令執行。這包括促進 BBF-TR069 和 OMA-DM 的非 RESTful 管理命令的命令執行。例如，BBF-TR069 的重啟可在資料結構中表示為 `.../commands/reboot/`。BBF-TR069 的下載可在資料結構中表示為 `.../commands/download/`。OMA-DM 的 Exec 可在資料結構中表示為 `.../commands/exec/`。OMA-DM 的複製可在資料結構中表示為 `.../commands/copy/`。

`<command>` 實例也可被用以模型化和表示其他管理 API 或遠端程序呼叫 (RPC)。資源命令可設置在現有“資源”的子資源之下且作為現有“資源”的子資源，使得每個 `commands/<command>` 被觸發時如果需要的話可自動地在“資源”上執行。此外，“`execResource`”可被用以指定 `commands/<command>` 可在其上執行的資源。在此方法中，資源命令可被設置在中央位置，並且不需要立即在“`execResource`”涉及的資源下。OMA-DM FUMO 為示例，兩個方法說明如下：

方法 1：使用“`execResource`”來指示（refer to）FUMO 中的操作

如第 23 圖所示，OMA FUMO 具有 3 個操作：下載、更新和下載並更新。在 OMA-DM 中，那 3 個操作由 DM 伺服器向 DM 用戶端發佈非 RESTful Exec 命令來觸發。為了以 RESTful 方式來調用那 3 個操作，Exec 被模型化為 `<command> exec`，其具有如第 23 圖列出的一些屬性。兩個重要的屬性是 `execEnable` 和 `execResource`。

當網路域（“NA”）的 M2M 應用需要進行“下載”操作時，它僅僅向.../mgmtObjs/commands/exec/發佈更新方法以設定：

execEnable = 1 且，

execResource = .../mgmtObjs/FUMO/Download

當 NA 需要執行“更新”操作時，它僅僅向.../mgmtObjs/commands/exec/發佈更新方法以設定：

execEnable = 1 且，

execResource = .../mgmtObjs/FUMO/Update

當 NA 需要執行“下載並更新”操作時，它僅僅向.../mgmtObjs/commands/exec/發佈更新方法以設定：

execEnable = 1 且，

execResource

= .../mgmtObjs/FUMO/DownloadAndUpdate。

方法 2：使用命令來重新模型化 FUMO 中的操作

如第 24A 圖-第 24B 圖所示，另一個方法將直接將所有 FUMO 操作重新模型化為<command>資源。為了觸發以執行一個操作，僅需要對對應<command>的 execEnable 的更新。結果，根本不需要定義附加的 Exec 命令。在此方法中，不需要“execResource”。

如第 24A 圖所示，3 個 FUMO 操作可在資源命令下分別被重新模型化為 3 個<command>資源（即下載、更新、下載並更新）。由於每個 FUMO 操作具有一些子/葉節點，它們需要被模型化為“execParameters”下的屬性。使用下載並更新作為示例，“PkgURL”可被添加為.../commands/downloadAndUpdate/execParameters 的屬性。

當 NA 需要執行“下載並更新”操作時，它僅僅向.../mgmtObjs/FUMO/commands/downloadAndUpdate/發佈

更新方法以設定：

`execEnable = 1` 且

`PkgURL` = 相應封包的 URL。

如果舊的 FUMO 操作需要被保存為它們自身，重新模型化命令可在節點 `Ext` 下被設置為子樹，如第 24B 圖所示。

在一個或多個實施例中，命令資源結構可支援相同非 RESTful 管理命令的多個發佈（例如多個 NA 或其他發佈者可請求相同非 RESTful 管理命令的命令執行）。

支援相同非 RESTful 管理命令的多個發佈的命令資源結構的兩個示例在第 25A 圖和第 25B 圖中示出。第 25A 圖中示出的命令資源結構類似於第 22A 圖的命令資源結構，並經由將 `<command>` 定義為多個實例的集合（即 `<commandInstances>`）而擴展了第 22A 圖的命令資源結構。如先前所述的過程可被用以調用 `<command>` 的命令執行並生成相應的 `<commandInstance>`。`<commandInstance>` 可維護由多個 NA 發佈的相應 `<command>` 實例。生成的 `<commandInstance>` 的每個命令執行可經由如上所述存取它的屬性和/或子資源而被停止或修改。例如，現有 `<commandInstance>` 的命令執行可經由使用 RESTful 方法更新將 `<commandInstance>/execEnable` 從 1 修改為 0 而被停止。`<commandInstance>` 和/或 `<commandInstance>` 命令執行狀態的變化也可以經由使用其他 RESTful 方法來存取和操作。

第 25B 圖中所示的命令資源結構類似於第 22B 圖的命令資源結構，並且經由將 `<command>/execRequests`（其也可被稱為 `<execInstances>`）定義為多個請求實例的集合（即 `<requestInstances>`（其也可被稱為 `<execInstances>`）），來擴展第 22B 圖的命令資源結構。前述過程將被用以調用命令執行或 `<command>` 的命令執行狀態的變化和/或生成相應的

<requestInstance>。 <requestInstance> 可維護相應的 <command> 實例（例如由 NA 發佈的那些）。生成的 <requestInstance> 的每個命令執行可如上所述經由存取它的屬性而被停止或修改。例如，現有 <requestInstance> 的命令執行可在 <requestInstance>/execEnable 上使用 RESTful 方法更新而被停止。

<requestInstance> 和 / 或 <requestInstance> 的命令執行狀態的改變也可使用其他 RESTful 方法來存取和操作。

第 26A 圖是示出資源命令的示例結構的框圖。資源命令可具有 <command> 的子資源，和以下屬性：

- 存取權 ID；
- 創建時間；
- 最後修改時間；
- 期滿時間；
- 搜索字串；
- 內容類型，其可在 FFS 處格式化；
- moID；
- 原始 MO；和
- 描述（即命令資源的文本格式描述）。

<command> 也可被模型化為屬性。

第 26B 圖是示出資源命令的示例結構的框圖。資源命令可具有子資源訂閱和 <command> 的子資源和以下屬性：

- 存取權 ID；
- 創建時間；
- 最後修改時間；
- 期滿時間；
- 搜索字串；
- 內容類型，其可在 FFS 處格式化；

moID；

原始 MO；和

描述（即命令資源的文本格式描述）。

<command>也可被模型化為屬性。

每個<command>可包括子資源（i）訂閱和（ii）execRequest。execRequest 可以用於儲存來自不同發佈者而用於調用相同非 RESTful 管理命令的命令執行的請求的佔位。在一個或多個實施方式中，這些請求對<command>實例可具有不同的引數（argument）。

每個<command>實例可具有以下屬性：

存取權 ID；

創建時間；

最後修改時間；

期滿時間；

搜索字串；

內容類型，其可在 FFS 處格式化；

moID；

原始 MO；和

描述（即命令資源的文本格式描述）。

<command>也可被模型化為屬性。

如第 27A 圖所示，每個<command>例如具有若干<commandInstance>作為子資源。每個<command>具有以下屬性：

存取權 ID：。

創建時間：。

最後修改時間：。

execEnable：用以觸發來執行命令的布林（Boolean）變數。

如果 `execEnable` 從 1 變化為 0，則 `<commandInstance>` 可被停止。`execEnable` 可被調整以支援更多的選項（例如暫停等）。

`execMode`：用以指定命令可如何被執行。

立即一次：接收器立即執行命令並僅執行一次。

立即並重複：接收器立即但重複地執行命令。兩次執行之間的間隔可由容器 `execParameters` 來指定。

隨機一次：接收器在隨機延遲後執行命令並僅執行一次。隨機延遲在 `execParameters` 中指定。

隨機並重複：接收器在隨機延遲後重複地執行命令。隨機延遲和兩次執行之間的間隔可由容器 `execParameters` 指定。

`execBaseDelay`：指定 `<commandInstance>` 可被執行前的最小延遲。

`execAdditionalDelay`：用以生成隨機附加延遲。總延遲可以是 `execBaseDelay` 加隨機附加延遲的和。

`execFrequency`：指定 `<commandInstance>` 可被重複執行的頻率。

`execNumber`：指定 `<commandInstance>` 可被重複執行的次數。

`execResource`：可表示在其上執行 `<commandInstance>` 的資源 URI。資源 URI 可指向包括一組資源的組資源；然後該命令可在那些資源的每一個上執行。`execResource` 是隨選的。如果 `<commandInstance>` 於 `<commandInstance>` 可在其上執行的資源下被設置為子資源，則可以不需要 `execResource`。

`execStatus`：可表示 `<commandInstance>` 的當前狀態。該狀態可以是未決、運行和完成（成功或失敗）的任一個。

`execIssuer`：可表示發佈 `<command>` 的發佈者。

execParameters：是用於儲存特定於每個單一命令的參數的佔位。

execBaseDelay 和 **execAdditionalDelay** 可被用以指定在“隨機一次”和“隨機重複”模式中於執行<commandInstance>前如何退回特定時間。**execFrequency** 和 **execNumber** 被用以指定在“立即並重複”和“隨機並重複”下執行相同<commandInstance>的頻率。

上述以“exec”開始的屬性也可被模型化為子資源。以上以“exec”開始的屬性可應用於普通 RESTful CRUD 操作（即創建、獲取、刪除、更新）以為操作資源增加更多的靈活性。

如第 27B 圖所示，每個<commandInstance>可具有以下屬性：

存取權 ID：。

創建時間：。

最後修改時間：。

execEnable：用以觸發執行命令的布林變數。如果 **execEnable** 從 1 變化為 0，則<commandInstance>可被停止。**execEnable** 可被修改以支援更多的選項（例如暫停等）。

execMode：被用以指定命令如何被執行。

立即一次：接收器可立即執行命令並僅執行一次。

立即並重複：接收器可立即但重複地執行命令。兩次執行之間的間隔可由容器 **execParameters** 來指定。

隨機一次：接收器可在隨機延遲後執行命令並僅執行一次。隨機延遲在 **execParameters** 中得以指定。

隨機並重複：接收器可在隨機延遲後重複地執行命令。隨機延遲和兩次執行之間的間隔可由容器 **execParameters** 指定。

execBaseDelay：用以指定<commandInstance>可被執行

前的最小延遲。

execAdditionalDelay：用以生成隨機附加延遲。總延遲可以是 **execBaseDelay** 加隨機附加延遲的和。

execFrequency：用以指定<commandInstance>可被重複執行的頻率。

execNumber：用以指定<commandInstance>可被重複執行的次數。

execResource：可表示在其上執行<commandInstance>的資源 URI。資源 URI 可指向包括一組資源的組資源；然後該命令可在那些資源的每一個上執行。**execResource** 是隨選的。如果<commandInstance>於<commandInstance>可在其上執行的資源下被設置為子資源，則可以不需要 **execResource**。

execStatus：可表示<commandInstance>的當前狀態。此狀態可以是未決、運行和完成（成功或失敗）的任一個。

execIssuer：可表示發佈<command>的發佈者。

execParameters：是用於儲存特定於每個單一命令的參數的佔位。

以上以“exec”開始的屬性也可被模型化為子資源。

現在參考第 28A 圖，其是示出另一個示例命令資源結構的框圖。此命令資源結構可包括若干資料結構元素。並且此命令資源結構的許多資料結構元素類似於第 22C 圖的命令資源結構。如上所述，命令資源結構支援多個發佈者的（例如多個 NA 的）、用於相同非 RESTful 管理命令（例如由屬性 **commandID** 標識）的請求。為了促進這樣的支援，在一個實施方式中，可為每個請求創建不同的<command>。每個<command>可具有不同的名字，但它的屬性 **commandID** 可以和另一個相同。屬性 **commandID**（其可由命令的類型來分類）指定哪個（哪些）命令引起該命令狀態的改變。可不

使用<command>的子資源 execRequest。另外，每個發佈者可在資源命令下創建特定的<command>。RESTful 方法更新可被用以調用所創建的<command>的命令執行（例如，在資源<command>的 execEnable 屬性上使用 RESTful 方法更新）。

替代地，子資源 execRequests 可被用以儲存用於相同<command>的命令執行的多個請求。發佈者可在<command>的 execEnable 屬性上使用 RESTful 方法更新，以調用<command>的命令執行。因此，接收器可為此發佈者在 .../commands/<command>/execRequests/ 下創建<requestInstance>。該發佈者可在<requestInstance>的屬性 execDisable、execPause 和 execResume 上分別使用 RESTful 方法更新，以取消、暫停和恢復此生成的<requestInstance>。

此外，該發佈者可在命令資源 .../commands/<command>/execRequests/<requestInstance>上使用 RESTful 方法刪除（DELETE）操作來刪除此生成的<requestInstance>。可不使用該<command>的屬性 execDisable、execPause、execResume。

如上詳細所述，4 個屬性（execEnable、execDisable、execPause、execResume）被配置，並被用以分別調用命令（<command>或<requestInstance>）的命令執行、暫停命令執行、恢復命令執行和取消命令執行。

作為一個替代選擇，4 個特殊的<command>資源可在資料結構中創建；

即.../commands/enable、.../commands/disable、.../commands/pause 和.../commands/resume。4 個特殊的命令可僅具有各自的“execResource”屬性。

.../commands/enable 可被用以調用一般命令執行。例

如，在其屬性 `execResource` 被設定為 `execResource = ".../commands/<command>"` 的 `".../commands/enable"` 資源上發佈 RESTful 方法創建 (CREATE) 或更新 (UPDATE) 操作，以調用 `".../commands/<command>"` 的命令執行。

`.../commands/disable` 可被用以調用 `.../commands/<command>`

(或 `.../commands-/<command>/execRequests/<requestInstance>`) 的一般取消命令執行。例如，在其屬性 `execResource` 被設定為 `execResource=.../commands/<command>`

(或 `.../commands-/<command>/execRequests/<requestInstance>`) 的 `.../commands/disable` 資源上發佈 RESTful 方法創建或更新，以調用 `.../commands/<command>`

(或 `.../commands-/<command>/execRequests/<requestInstance>`) 的取消命令執行。

`.../commands/pause` 可被用以調用現有 `.../commands/<command>`

(或 `.../commands-/<command>/execRequests/<requestInstance>`) 的暫停命令執行。例如，在其屬性 `execResource` 被設定為 `execResource=.../commands/<command>`

(或 `.../commands-/<command>/execRequests/<requestInstance>`) 的 `./commands/pause` 資源上發佈 RESTful 方法創建或更新可調用 `.../commands/<command>`

(或 `.../commands-/<command>/execRequests/<requestInstance>`) 的暫停命令執行。

`.../commands/resume` 可被用以調用現有 `.../commands/<command>`

(或 `.../commands-/<command>/execRequests/<requestInstance>`) 的恢復命令執行。例如，在其屬性 `execResource` 被設定

為 `execResource=.../commands/<command>`
 （或 `.../commands-/<command>/execRequests/<requestInstance>`）的 `./commands/resume`
 資源上發佈 RESTful 方法創建或更新可調用恢復命令執行。

用於 xREM 命令管理的示例訊息流程圖

第 29A 圖-第 29R 圖是示出用於資源命令的 xREM 的示例訊息流的訊息流程圖。第 29A 圖-第 29R 圖的訊息流參考第 22C 圖、第 25B 圖和第 28A 圖-第 28D 圖的命令資源結構而描述。訊息流也可使用資源命令或 `<commands>` 的其他結構來實現。

現在參考第 29A 圖，示出用於創建資源命令結構的示例訊息流的訊息流程圖。使用第 29A 圖的訊息流所創建的資源命令結構可包括在第 22C、25B 和 28A 圖中所示的資源命令結構的一些或全部。例如，資源命令結構可包括在這樣的圖式中所示出的所有 `<command>` 元素。替代地，資源命令結構可包括所示 `<command>` 元素的子集。這樣的子集可包括 `<command>` 屬性和/或子資源的子集。以示例的方式，資源命令結構可包括（i）屬性，例如“屬性”、`execEnable` 和 `commandID`；和（ii）子資源，例如 `execParameters`、`execRequests` 和 `subscriptions`。作為另一個示例，資源命令結構可包括（i）屬性，例如“屬性”、`execEnable`、`execDisable`、`execPause`、`execResume` 和 `commandID`；和（ii）子資源，例如 `execParameters`、`execRequests` 和 `subscriptions`。仍然作為另一個示例，資源命令結構可包括（i）屬性，例如“屬性”、`execEnable`、`execDisable` 和 `commandID`；和（ii）子資源，例如 `execParameters`、`execRequests` 和 `subscriptions`。資源命令結構可包括在第 22C、25B 和 28A 圖中所示的 `<command>`

元素（例如屬性和子資源）的其他組合。

資源命令結構還可包括未在第 22C、25B 和 29A 圖中示出的<command>的其他元素。這些其他元素可包括在與第 22C、25B 和 28A 圖中所示的<command>元素的任意組合相結合的資源命令結構中。

為了發起資源命令結構的創建，發佈者可向 M2M 伺服器發佈 RESTful 方法創建（CREATE）。該發佈者可以是例如 NA、M2M 裝置和 M2M GW 的任一個。作為發佈者，M2M 裝置或 M2M GW 可響應於註冊或作為向 M2M 伺服器註冊的部分，而發佈 RESTful 方法更新。M2M 裝置、M2M GW 或 NA 也可在其他時間發佈 RESTful 方法創建。

為了在<sclbase>的給定節點促進資源命令結構的創建，RESTful 方法創建可包括在其下資源命令結構可被創建的節點（例如資源）的識別符。例如，RESTful 方法創建可包括資源<mgmtObjs>的識別符（此後稱為“<mgmtObjs>識別符”），以便將它識別為在其下資源命令結構可被創建的節點。

作為一個替代選擇，RESTful 方法創建可包括資源<commands>的識別符（此後稱為“<commands>識別符”），以將資源<commands>識別為在其下資源命令結構可被創建的節點。如上所述，<mgmtObjs>和<commands>識別符的每一個可以是例如 URI、鏈結和位址的任一個。為了簡化說明，以下假設資源命令結構可在資源<mgmtObjs>下創建。然而，資源命令結構可在<sclbase>的任意節點下創建。

RESTful 方法創建還可包括用於填充資源命令結構的任意屬性的資訊。例如，RESTful 方法創建可包括填充 commandID 屬性的資訊，以便指示可被請求執行並使用該資源命令結構來執行的非 RESTful 命令的類型。

RESTful 方法創建可進一步包括用於填充該資源命令結構的一個或多個子資源（例如參數資源）的資訊。例如，RESTful 方法創建可包括用於填充 `execParameters` 子資源的一個或多個參數和/或引數。這些參數和/或引數可特定於資源 `<command>` 和/或非 RESTful 命令。RESTful 方法創建還可包括用於根據非 RESTful 命令來填充子資源 `execMode`、`execBaseDelay`、`execAdditionalDelay`、`execFrequency`、`execNumber` 和 `execResource` 的任一個的資訊。

在接收到 RESTful 方法創建後，M2M 伺服器可在 M2M 伺服器的資源 `<mgmtObjs>` 下創建資源命令結構。所創建的資源命令結構（此後成為“伺服器 `<command>`”）可包括在第 22C、25B 和 28A 圖中示出的所有 `<command>` 元素。替代地，伺服器 `<command>` 可包括所示 `<command>` 元素的子集，例如 `<command>` 的屬性和/或子資源的子集。伺服器 `<command>` 例如可包括“屬性”、`execEnable` 和 `commandID` 屬性；和 `execParameters`、`execRequests` 和訂閱子資源。

替代地，伺服器 `<command>` 可包括“屬性”、`execEnable`、`execDisable`、`execPause`、`execResume` 和 `commandID` 屬性；和 `execParameters`、`execRequests` 和訂閱子資源。仍然作為另一個示例，伺服器 `<command>` 可包括屬性”、`execEnable`、`execDisable` 和 `commandID` 屬性；和 `execParameters`、`execRequests` 和訂閱子資源。伺服器 `<command>` 也可包括（在第 22C、25B 和 27A 圖中示出或未示出的）`<command>` 元素（例如屬性和子資源）的其他組合。

如果用於填充伺服器 `<command>` 的屬性和/或子資源的資訊被包括在 RESTful 方法創建中，M2M 伺服器可以所提供的資訊來填充這樣的相應屬性和/或子資源。如果用於填充任意給定的屬性和/或子資源的資訊沒有被提供，M2M 伺

伺服器可將這樣的屬性和/或子資源保持初始創建的那樣（例如空白或具有預設屬性和/或參數）。如下更詳細地描述的那樣，這些屬性和子資源可在創建伺服器<command>後，經由向 M2M 伺服器發佈包括用於填充該屬性和/或子資源的 RESTful 方法更新來被填充。

響應於 RESTful 方法創建，M2M 伺服器可向發佈者發送響應訊息（“響應”）。如所示地，M2M 伺服器可在創建該伺服器<command>後發送該響應。替代地，M2M 伺服器可在伺服器<command>創建期間發送該響應。M2M 伺服器還可向發佈者發送應答（“ACK/NACK”）訊息（未示出），用於應答 RESTful 方法創建的接收。發佈者沒有接收到 ACK/NACK 訊息（例如在特定時間內）可指示以 M2M 伺服器的 RESTful 方法創建的否定應答。

該響應可包括指示 M2M 伺服器成功地創建或未能創建伺服器<command>的指示（例如代碼）。該指示可以是指示成功創建的一個值和指示失敗的另一個值。替代地，該響應可包括第一指示（例如代碼），以指示 M2M 伺服器成功地創建伺服器<command>，和第二指示（例如代碼），以指示創建伺服器<command>失敗。作為另一個替代選擇，M2M 伺服器可僅當該 M2M 伺服器成功地創建伺服器<command>時才發佈該響應。

該響應可包括在創建期間被分配給伺服器<command>的節點的識別符（“伺服器<command>識別符”）。此伺服器<command>識別符可被用以存取伺服器<command>。伺服器<command>命令識別符可以例如是該節點的 URI、鏈結或位址。

該響應還可包括在創建期間被分配給 execEnable 子資源的節點的識別符（“execEnable 識別符”）。如上和以下更

詳細地描述的那樣，execEnable 識別符可被用以引起命令執行狀態的變化（例如調用命令執行、取消命令執行、暫停命令執行和恢復命令執行的任一個）。

替代地，該響應還可包括 execEnable 識別符、在創建期間被分配給 execDisable 的節點的識別符（“execDisable 識別符”）（如果有的話）、在創建期間被分配給 execPause 的節點的識別符（“execPause 識別符”）（如果有的話）和在創建期間被分配給 execResume 的節點的識別符（“execResume 識別符”）（如果有的話）。如上和以下更詳細地描述的那樣，execEnable、execDisable、execPause 和 execResume 識別符的每一個可被用以引起命令執行狀態的變化。execEnable 可被用以調用命令執行。execDisable 可被用以調用取消命令執行。execPause 可被用以調用暫停命令執行。execResume 可被用以調用恢復命令執行。

作為一個替代選擇，M2M 伺服器可在伺服器 <command> 的屬性或子資源（例如“屬性”屬性）中儲存 execEnable、execDisable、execPause 和 execResume 識別符的任一個（如果有的話）。這樣，這麼儲存的 execEnable、execDisable、execPause 和 execResume 識別符的任一個可在隨後的時間獲取。

現在參考第 29B 圖，示出用於從資源命令結構（例如伺服器 <command>）獲取資訊的示例訊息流的訊息流程圖。為了發起獲取，NA（即發佈者）可向 M2M 伺服器發佈 RESTful 方法獲取（RETRIEVE）。RESTful 方法獲取可包括伺服器 <command> 識別符。

在接收到 RESTful 方法獲取後，M2M 伺服器可使用伺服器 <command> 識別符來定位該伺服器 <command>。一旦定位，M2M 伺服器可從具有這樣的可獲取資訊的伺服器

<command>的屬性和/或子資源來查詢和獲得可獲取資訊（例如屬性、參數和/或引數）（此後稱為“經獲取的屬性資訊”和/或“經獲取的子資源資訊”）。經獲取的屬性和/或經獲取的子資源資訊可包括所儲存的 execEnable、execDisable、execPause 和 execResume 識別符的任一個。

在獲得經獲取的屬性和/或經獲取的子資源資訊後，M2M 伺服器可向 NA 發送一個響應。此響應可包括伺服器 <command>識別符和經獲取的屬性和/或經獲取的子資源資訊。

作為一個替代選擇，RESTful 方法獲取可被用以從具有這樣的可獲取資訊的伺服器 <command>的屬性和/或子資源來獲取可獲取資訊的一個或多個選擇部分。為了促進此事，RESTful 方法獲取可包括被分配給具有可獲取資訊的選擇部分的伺服器 <command>的屬性和/或子資源的每個節點的識別符。使用該識別符（或者這些識別符，如果多於一個的話），M2M 伺服器可定位、查詢和獲得可獲取資訊的選擇部分（此後成為“所選擇經獲取的屬性和/或經獲取子資源資訊”）。M2M 伺服器然後可向 NA 發送包括該所選擇經獲取的屬性和/或經獲取的子資源資訊的響應。

雖然未示出，但 M2M 伺服器還可以向發佈者發送 ACK/NACK 訊息，用於應答 RESTful 方法獲取的接收。發佈者沒有接收到 ACK/NACK 訊息（例如在特定的時間內）可指示以 M2M 伺服器的 RESTful 方法獲取的否定應答。

現在參考第 29C 圖，示出用於刪除資源命令結構（例如伺服器 <command>或伺服器 <command>實例的集合（“伺服器 <command>實例”））的示例訊息流的訊息流程圖。為了開始刪除，發佈者可向 M2M 伺服器發佈 RESTful 方法刪除（DELETE）。該發佈者可以是 NA、M2M 裝置或 M2M GW。

M2M 裝置和/或 M2M GW 可響應於非 RESTful 命令的重啟、取消、去註冊等，而發佈 RESTful 方法刪除。

RESTful 方法刪除可包括伺服器<command>識別符。替代地，RESTful 方法刪除可包括分配給在其下已創建伺服器<command>實例的資源（例如伺服器<commands>）的節點識別符（此後成為“伺服器<commands>識別符”）。

在接收到 RESTful 方法刪除後，M2M 伺服器可分別使用伺服器<command>識別符或伺服器<commands>識別符來定位和刪除伺服器<command>或伺服器<commands>。這可包括刪除伺服器<command>的所有屬性和/或子資源，或對於伺服器<commands>，刪除這樣的伺服器<commands>的伺服器<command>實例的每一個。

響應於 RESTful 方法刪除，M2M 伺服器可向發佈者發送響應。M2M 伺服器可在刪除了伺服器<command>或伺服器<commands>後發送該響應，如所示那樣。替代地，M2M 伺服器可在刪除伺服器<command>或伺服器<commands>期間發送該響應。M2M 伺服器還可向發佈者發送 ACK/NACK 訊息（未示出），用於應答 RESTful 方法刪除的接收。發佈者沒有接收到 ACK/NACK 訊息（例如在特定的時間內）可指示以 M2M 伺服器進行的 RESTful 方法刪除的否定應答。

該響應可包括用以指示 M2M 伺服器成功地刪除或未能刪除伺服器<command>或伺服器<commands>的指示（例如代碼）。該指示可以是指示成功地刪除的一個值和指示失敗的另一個值。替代地，該響應可包括第一指示（例如代碼），以指示 M2M 伺服器成功地刪除伺服器<command>或伺服器<commands>，和第二指示（例如代碼），以指示刪除伺服器<command>或伺服器<commands>失敗。作為另一個替代選擇，M2M 伺服器可僅當該 M2M 伺服器成功地刪除伺服器

<command>或伺服器<commands>時才發佈該回應。

第 29D 圖示出以用於在執行非 RESTful 命令的資訊(如所示)來更新資源命令結構(例如伺服器<command>)的示例訊息流的訊息流程圖。第 29E 圖-第 29N 圖是示出用於更新資源命令結構(例如伺服器<command>)以調用非 RESTful 命令的命令執行的示例訊息流的流程圖。雖然第 29D 圖的訊息流和第 29E 圖-第 29N 圖的訊息流可使用 RESTful 方法更新,但第 29D 圖的訊息流可使用 RESTful 方法更新,使用可被用來執行非 RESTful 命令(即在命令執行已被調用後)的資訊(例如屬性、參數和/或引數的任一個)來填充資源命令結構(例如伺服器<command>)的元素。然而,第 29E 圖-第 29N 圖的訊息流可使用 RESTful 方法更新來引起非 RESTful 命令的命令執行的調用。

現在參考第 29D 圖,NA(即發佈者)可向 M2M 伺服器發佈 RESTful 方法更新以發起更新。RESTful 方法更新可包括伺服器<command>識別符和用於使用這樣的資訊填充伺服器<command>的屬性和/或子資源的資訊(例如屬性、參數和/或引數)。如上述 RESTful 方法創建,RESTful 方法更新可包括用以填充 commandID 屬性的資訊,以便指示可請求執行和可使用該資源命令結構來執行的非 RESTful 命令的類型。

RESTful 方法更新可進一步包括用於填充資源命令結構的一個或多個子資源(例如參數資源)的資訊。例如,RESTful 方法更新可包括用於填充 execParameters 子資源的一個或多個參數和/或引數。這些參數和/或引數可特定於資源<command>和/或非 RESTful 命令。RESTful 方法更新還可包括用於根據非 RESTful 命令來填充子資源 execMode、execBaseDelay、execAdditionalDelay、execFrequency、

execNumber 和 execResource 的任一個的資訊。

在接收到 RESTful 方法更新後，M2M 伺服器使用伺服器命令識別符來定位伺服器<command>。M2M 伺服器可此後以所提供的資訊來填充這樣的伺服器<command>的屬性和/或子資源。如果用於填充任意給定的屬性和/或子資源的資訊未被提供，M2M 伺服器可保持這樣的屬性和/或子資源為初始創建的那樣（例如空白或具有預設屬性和/或參數）或最後修改的那樣。

響應於 RESTful 方法更新，M2M 伺服器可向發佈者發送響應。M2M 伺服器可在更新伺服器<command>後發送該響應，如所示那樣。替代地，M2M 伺服器可在更新伺服器<command>期間發送該響應。M2M 伺服器還可向發佈者發送 ACK/NACK 訊息（未示出），用於應答 RESTful 方法更新的接收。發佈者沒有接收到 ACK/NACK 訊息（例如在特定的時間內）可指示以 M2M 伺服器進行的 RESTful 方法更新的否定應答。

該響應可包括用以指示 M2M 伺服器成功地更新或無法更新伺服器<command>的指示（例如代碼）。該指示可以是指示成功更新的一個值和指示失敗的另一個值。替代地，該響應可包括第一指示（例如代碼），以指示 M2M 伺服器成功地更新伺服器<command>，和第二指示（例如代碼），以指示更新伺服器<command>失敗。作為另一個替代選擇，M2M 伺服器可僅當該 M2M 伺服器成功地更新伺服器<command>時才發佈該響應。

該響應還可包括更新的伺服器<command>識別符和/或屬性和/或子資源。該響應可進一步包括被用以更新屬性和/或子資源的資訊或指示相同事件的確認。

現在參考第 29E 圖，示出用於更新資源命令結構（例如

伺服器<command>)以調用非 RESTful 命令的命令執行的示例訊息流的流程圖。為了發起該更新，NA (即發佈者)向 M2M 伺服器發佈 RESTful 方法更新。除了制定子資源 execEnable (例如.../<command>/execEnable)外，此 RESTful 方法更新可包括伺服器<command>識別符以及指定子資源 execEnable (例如.../<command>/execEnable)。替代地，RESTful 方法更新可包括 execEnable 識別符。

響應於此 RESTful 方法更新，M2M 伺服器可向發佈者發送 ACK/NACK 訊息 (以響應示出)，以應答 RESTful 方法更新的接收。發佈者沒有接收到 ACK/NACK 訊息 (例如在特定的時間內)可指示以 M2M 伺服器進行的 RESTful 方法更新的否定應答。

M2M 伺服器可根據伺服器<command>的屬性和/或子資源來調用非 RESTful 命令的命令執行。作為調用命令執行的一部分，M2M 伺服器可執行若干功能。這些功能可包括例如將在伺服器<command>中 (例如經由 CommandID) 所識別的命令轉換或翻譯為可由 M2M 裝置或 M2M GW 解釋或執行的非 RESTful 命令。該功能還可包括用於在伺服器<command>所識別的命令和可由 M2M 裝置或 M2M GW 解釋或執行的非 RESTful 命令之間映射屬性、參數和/或引數 (“經映射的屬性、參數和/或引數”) 的功能。

M2M 伺服器還可在 execRequests 子資源下創建資源 <requestInstance>，以維護和跟蹤調用的伺服器<command>。資源 <requestInstance> 可包括伺服器<command>的一個或多個屬性和/或子資源。M2M 伺服器可在創建資源<requestInstance>時從伺服器<command>繼承或引入這樣的屬性和/或子資源。給定伺服器<command>可包括各種元素的組合，資源<requestInstance>也可包括各種元

素的組合。例如，資源<requestInstance>可包括在第 25B、28C 和 28D 圖中示出的資源<requestInstance>的所有元素，其例如可以是在第 22C、25B 和 28A 圖中示出的伺服器<command>的必然結果（corollary）。

替代地，資源<requestInstance>可包括所示資源<requestInstance>的元素的子集。這樣的子集可包括資源<requestInstance>的屬性和/或子資源的子集。以示例的方式，資源<requestInstance>可包括“屬性”、execEnable、execStatus、execResult 和 execDisable 屬性；和 subscriptions 訂閱資源。作為另一個示例，資源<requestInstance>可包括“屬性”、execEnable、execStatus、execResult 和 execDisable、execPause 和 execResume 屬性；和 subscriptions 子資源。仍然作為另一個示例，資源<requestInstance>可包括“屬性”、execEnable、execStatus 和 execResult 屬性；和 subscriptions 子資源。子資源可包括在第 25B、28C 和 28D 圖中示出的伺服器<command>和/或資源<requestInstance>的元素（例如屬性和子資源）的其他組合。

資源<requestInstance>還可包括未在第 25B、28C 和 28D 圖中示出的<command>的其他元素。這些其他元素可被包括在資源<requestInstance>中，與在第 25B、28C 和 28D 圖中示出的資源<requestInstance>的元素的任意組合相結合。

一旦在 execRequests 子資源下創建資源<requestInstance>，M2M 伺服器可向資源<requestInstance>分配在其處創建資源<requestInstance>的節點的識別符（“<requestInstance> 識別符”）。如上所述，此<requestInstance>識別符例如可以是 URI、鏈結和位址的任一個。

M2M 伺服器可向資源<requestInstance>引入（或引起資

源<requestInstance>來繼承) 被填充入伺服器<command>的屬性和/或子資源的一些或全部資訊，其被引入到資源<requestInstance>或由資源<requestInstance>繼承。這樣的資訊可包括被填充到伺服器<command>的 execParameters 子資源的一個或多個參數和/或引數。該資訊還可包括在例如 execMode 、 execBaseDelay 、 execAdditionalDelay 、 execFrequency、execNumber 和 execResource 子資源中填充的一些或全部資訊。

在創建資源<requestInstance>後，M2M 伺服器可向 M2M 裝置和/或 M2M GW 發送請求訊息（“請求”），以便在這樣的 M2M 裝置和/或 M2M GW 處調用命令執行。該請求可包括非 RESTful 命令。該請求還可包括經映射的屬性、參數和引數的任一個。

響應於該請求，M2M 裝置和/或 M2M GW 可執行非 RESTful 命令。M2M 裝置和/或 M2M GW 可發送對該請求的響應（“命令執行響應”）。此命令執行響應可包括結果，如果有的話（如由“{}”符號表示）。M2M 伺服器可為隨後的獲取來擷取和儲存在資源<requestInstance>的 execResult 子資源中的結果。替代地和/或附加地，M2M 伺服器可在向 NA 的響應（“NA 響應”）中發送結果。在獲得結果後，NA 可處理它們。

命令執行響應還可包括指示（例如代碼），用以指示 M2M 裝置和/或 M2M GW 成功地執行非 RESTful 命令的命令執行或未能執行該非 RESTful 命令的命令執行。該指示可以是指示成功的命令執行的一個值，和指示失敗的另一個值。替代地，該響應可包括第一指示（例如代碼），以指示 M2M 裝置和/或 M2M GW 成功地執行命令執行，和第二指示（例如代碼），以指示 M2M 裝置和/或 M2M GW 未能執

行該命令執行。作為另一個替代選擇，M2M 裝置和/或 M2M GW 可僅當該 M2M 裝置和/或 M2M GW 成功地執行命令執行時才發佈該響應。

雖然未示出，M2M 伺服器可在 M2M 裝置和/或 M2M GW 未能執行非 RESTful 命令的命令執行時刪除資源 <requestInstance>。M2M 伺服器還可響應於被通知 M2M 裝置和/或 M2M GW 未能執行非 RESTful 命令的命令執行，而刪除伺服器 <command> 和 / 或一個或多個伺服器 <commands>。

現在參考第 29F 圖，示出用於更新資源命令結構（例如伺服器 <command>）以調用非 RESTful 命令的命令執行的另一個示例訊息流的流程圖。除了如本文所描述的，第 29F 圖的訊息流類似於第 29E 圖的訊息流。在第 29F 圖的訊息流中，M2M 伺服器可在創建 <requestInstance> 前發佈請求。同樣地，M2M 伺服器可在創建資源 <requestInstance> 後接收命令執行響應。

第 29G 圖是示出用於更新資源命令結構（例如伺服器 <command>）以調用非 RESTful 命令的命令執行的另一個示例訊息流的流程圖。除了如本文所描述的，第 29G 圖的訊息流類似於第 29E 圖的訊息流。在第 29G 圖的訊息流中，在創建包括 execDisable、execPause 和/或 execResume 屬性（或子資源）的資源 <requestInstance> 後，M2M 伺服器可向發佈者發送相應的 execDisable、execPause 和/或 execResume 識別符。

替代地，M2M 伺服器可在資源 <requestInstance> 的屬性或子資源（例如“屬性”屬性）中儲存 execEnable、execDisable、execPause 和 execResume 識別符的任一個。這樣，這麼儲存的 execEnable、execDisable、execPause 和

`execResume` 識別符可在隨後的時間被獲取。如下所述，發佈者（例如 NA）可使用 `execDisable`、`execPause` 和/或 `execResume` 識別符來分別調用取消命令執行、暫停命令執行和/或回復命令執行。

現在參考第 29H 圖，示出用於更新資源命令結構（例如伺服器<command>）以調用非 RESTful 命令的命令執行的另一個示例訊息流的流程圖。除了如本文所描述的，第 29H 圖的訊息流類似於第 29G 圖的訊息流。在第 29G 圖的訊息流中，M2M 伺服器可在創建資源<requestInstance>之前發佈請求。同樣地，M2M 伺服器可在創建資源<requestInstance>之後接收命令執行響應。

第 29I 圖和第 29J 圖是示出用於更新資源命令結構（例如伺服器<command>）以調用非 RESTful 命令的命令執行狀態的示例訊息流的流程圖。為了發起該更新，NA（即發佈者）可向 M2M 伺服器發佈 RESTful 方法更新。此 RESTful 方法更新可包括<requestInstance>識別符以及指定子資源 `execDisable`（例如 `.../<command>/execDisable`）。替代地，RESTful 方法更新可包括經由從 M2M 伺服器發送的響應而獲得的 `execDisable` 識別符（第 29I 圖），或在從資源<requestInstance>的屬性獲取它後而包括所獲得的 `execDisable` 識別符（第 29J 圖）。

響應於此 RESTful 方法更新，M2M 伺服器可向發佈者發送 ACK/NACK 訊息（顯示為響應），以應答 RESTful 方法更新的接收。發佈者沒有接收到 ACK/NACK 訊息（例如在特定的時間內）可指示以 M2M 伺服器進行的 RESTful 方法更新的否定應答。

M2M 伺服器然後可根據伺服器<command>的屬性和/或子資源來調用非 RESTful 命令的取消命令執行。作為調用

取消命令執行的一部分，M2M 伺服器可執行若干功能。這些功能可包括例如將如同在資源<requestInstance>中所識別的取消命令執行轉換或翻譯為用於調用執行非 RESTful 命令的取消命令執行的非 RESTful 命令（“非 RESTful 取消命令”）。該功能還可包括用於準備在資源<requestInstance>中所識別的命令和非 RESTful 取消命令之間經映射的屬性、參數和/或引數的功能。

在進行用於調用取消命令執行的功能後，M2M 伺服器可向 M2M 裝置和/或 M2M GW 發送請求，以便在這樣的 M2M 裝置和/或 M2M GW 處調用取消命令執行。該請求可包括非 RESTful 取消命令。該請求還可包括用於執行非 RESTful 取消命令的經映射的屬性、參數和引數的任一個。

響應於該請求，M2M 裝置和/或 M2M GW 可執行非 RESTful 取消命令。M2M 裝置和/或 M2M GW 可發送命令執行響應。此命令執行響應可包括結果，如果有的話。M2M 伺服器可為隨後的獲取來擷取和儲存在資源<requestInstance>的 execResult 子資源中的結果。替代地和/或附加地，M2M 伺服器可在 NA 響應中發送結果。在獲得結果後（如果有的話），NA 可處理它們。

命令執行響應還可包括指示（例如代碼），用以指示 M2M 裝置和/或 M2M GW 成功地執行非 RESTful 取消命令的取消命令執行或未能執行該非 RESTful 取消命令的取消命令執行。該指示可以是指示成功的取消命令執行的一個值，和指示失敗的另一個值。替代地，該響應可包括第一指示（例如代碼），以指示 M2M 裝置和/或 M2M GW 成功地執行取消命令執行，和第二指示（例如代碼），以指示 M2M 裝置和/或 M2M GW 未能執行該取消命令執行。作為另一個替代選擇，M2M 裝置和/或 M2M GW 可僅當該 M2M 裝置

和/或 M2M GW 成功地執行取消命令執行時才發佈該響應。

雖然未示出，M2M 伺服器可在 M2M 裝置和/或 M2M GW 未能執行非 RESTful 命令的取消命令執行時刪除資源 <requestInstance>。M2M 伺服器還可響應於被通知 M2M 裝置和/或 M2M GW 未能執行非 RESTful 取消命令的取消命令執行，而刪除伺服器 <command> 和/或一個或多個伺服器 <commands>。

第 29K 圖和第 29L 圖是示出用於更新資源命令結構(例如伺服器 <command>)以調用非 RESTful 命令的命令執行狀態的改變的示例訊息流的流程圖。為了發起該更新，NA(即發佈者)可向 M2M 伺服器發佈 RESTful 方法更新。此 RESTful 方法更新可包括 <requestInstance> 識別符以及指定子資源 execPause(例如.../<command>/execPause)。替代地，RESTful 方法更新可包括經由從 M2M 伺服器發送的響應所獲得的 execPause 識別符(第 29K 圖)，或在從資源 <requestInstance> 的屬性獲取它後，包括所獲得的 execPause 識別符(第 29L 圖)。

響應於此 RESTful 方法更新，M2M 伺服器可向發佈者發送 ACK/NACK 訊息(顯示為響應)，以應答 RESTful 方法更新的接收。發佈者沒有接收到 ACK/NACK 訊息(例如在特定的時間內)可指示以 M2M 伺服器進行的 RESTful 方法更新的否定應答。

M2M 伺服器然後可根據伺服器 <command> 的屬性和/或子資源來調用非 RESTful 命令的暫停命令執行。作為調用暫停命令執行的一部分，M2M 伺服器可執行若干功能。這些功能可包括例如將在資源 <requestInstance> 中所識別的暫停命令執行轉換或翻譯為用於調用執行非 RESTful 命令的暫停命令執行的非 RESTful 命令(“非 RESTful 暫停命令”)。

該功能還可包括用於準備在資源<requestInstance>中所識別的命令和非 RESTful 暫停命令之間經映射的屬性、參數和/或引數的功能。

在執行用於調用暫停命令執行的功能後，M2M 伺服器可向 M2M 裝置和/或 M2M GW 發送請求，以便在這樣的 M2M 裝置和/或 M2M GW 處調用暫停命令執行。該請求可包括非 RESTful 暫停命令。該請求還可包括用於執行非 RESTful 暫停命令的經映射的屬性、參數和引數的任一個。

響應於該請求，M2M 裝置和/或 M2M GW 可執行非 RESTful 暫停命令。M2M 裝置和/或 M2M GW 可發送命令執行響應。此命令執行響應可包括結果，如果有的話。M2M 伺服器可為隨後的獲取來擷取和儲存在資源<requestInstance>的 execResult 子資源中的結果。替代地和/或附加地，M2M 伺服器可在 NA 響應中發送結果。在獲得結果後（如果有的話），NA 可處理它們。

命令執行響應還可包括指示（例如代碼），用以指示 M2M 裝置和/或 M2M GW 成功地執行非 RESTful 暫停命令的暫停命令執行或未能執行該非 RESTful 暫停命令的暫停命令執行。該指示可以是指示成功的暫停命令執行的一個值，和指示失敗的另一個值。替代地，該響應可包括第一指示（例如代碼），以指示 M2M 裝置和/或 M2M GW 成功地執行暫停命令執行，和第二指示（例如代碼），以指示 M2M 裝置和/或 M2M GW 未能執行該暫停命令執行。作為另一個替代選擇，M2M 裝置和/或 M2M GW 可僅當該 M2M 裝置和/或 M2M GW 成功地執行暫停命令執行時才發佈該命令執行響應。

雖然未示出，M2M 伺服器可在 M2M 裝置和/或 M2M GW 未能執行非 RESTful 命令的暫停命令執行時刪除資源

<requestInstance>。M2M 伺服器還可響應於被通知 M2M 裝置和/或 M2M GW 未能執行非 RESTful 暫停命令的暫停命令執行，而刪除伺服器<command>和/或一個或多個伺服器<commands>。

第 29M 圖和第 29N 圖是示出用於更新資源命令結構(例如伺服器<command>)以調用非 RESTful 命令的命令執行狀態的改變的示例訊息流的流程圖。為了發起該更新，NA(即發佈者)可向 M2M 伺服器發佈 RESTful 方法更新。此 RESTful 方法更新可包括<requestInstance>識別符以及指定子資源 execResume(例如.../<command>/exec-Resume)。替代地，RESTful 方法更新可包括經由從 M2M 伺服器發送的響應所獲得的 execResume 識別符(第 29M 圖)，或在從資源<requestInstance>的屬性獲取它後，包括 execResume 識別符(第 29N 圖)。

響應於此 RESTful 方法更新，M2M 伺服器可向發佈者發送 ACK/NACK 訊息(顯示為響應)，以應答 RESTful 方法更新的接收。發佈者沒有接收到 ACK/NACK 訊息(例如在特定的時間內)可指示以 M2M 伺服器進行的 RESTful 方法更新的否定應答。

M2M 伺服器然後可根據伺服器<command>的屬性和/或子資源來調用非 RESTful 命令的恢復命令執行。作為調用恢復命令執行的一部分，M2M 伺服器可執行若干功能。這些功能可包括例如將在資源<requestInstance>中所識別的恢復命令執行轉換或翻譯為用於調用已暫停的非 RESTful 命令的恢復命令執行的非 RESTful 命令(“非 RESTful 恢復命令”)。該功能還可包括用於準備在資源<requestInstance>中所識別的命令和非 RESTful 恢復命令之間經映射的屬性、參數和/或引數的功能。

在執行用於調用恢復命令執行的功能後，M2M 伺服器可向 M2M 裝置和/或 M2M GW 發送請求，以便在這樣的 M2M 裝置和/或 M2M GW 處調用恢復命令執行。該請求可包括非 RESTful 恢復命令。該請求還可包括用於執行非 RESTful 恢復命令的經映射的屬性、參數和引數的任一個。

響應於該請求，M2M 裝置和/或 M2M GW 可執行非 RESTful 恢復命令。M2M 裝置和/或 M2M GW 可發送命令執行響應。此命令執行響應可包括結果，如果有的話。M2M 伺服器可為隨後的獲取來擷取和儲存在資源 <requestInstance> 的 execResult 子資源中的結果。替代地和/或附加地，M2M 伺服器可在 NA 響應中發送結果。在獲得結果後（如果有的話），NA 可處理它們。

命令執行響應還可包括指示（例如代碼），用以指示 M2M 裝置和/或 M2M GW 成功地執行非 RESTful 恢復命令的恢復命令執行或未能執行該非 RESTful 恢復命令的恢復命令執行。該指示可以是指示成功的恢復命令執行的一個值，和指示失敗的另一個值。替代地，該響應可包括第一指示（例如代碼），以指示 M2M 裝置和/或 M2M GW 成功地執行恢復命令執行，和第二指示（例如代碼），以指示 M2M 裝置和/或 M2M GW 未能執行該恢復命令執行。作為另一個替代選擇，M2M 裝置和/或 M2M GW 可僅當該 M2M 裝置和/或 M2M GW 成功地執行恢復命令執行時才發佈該命令執行響應。

雖然未示出，M2M 伺服器可在 M2M 裝置和/或 M2M GW 未能執行非 RESTful 命令的恢復命令執行時刪除資源 <requestInstance>。M2M 伺服器還可響應於被通知 M2M 裝置和/或 M2M GW 未能執行非 RESTful 暫停命令的恢復命令執行，而刪除伺服器 <command> 和/或一個或多個伺服器

<commands>。

現在參考第 29O 圖，示出用於以在執行非 RESTful 命令時使用的資訊來更新資源命令結構（例如 <requestInstance>）的示例訊息流的流程圖。和第 29D 圖的訊息流一樣，第 29O 圖的訊息流可使用 RESTful 方法更新，以可用於執行非 RESTful 命令的資訊（例如屬性、參數和/或引數中的任一個）來填充資源命令結構（例如 <requestInstance>）的元素。

如所示那樣，NA（即發佈者）可向 M2M 伺服器發佈 RESTful 方法更新以發起更新。RESTful 方法更新可包括 <requestInstance> 識別符和用於以這樣的資訊來填充 <requestInstance> 的屬性和/或子資源的資訊（例如屬性、參數和/或引數）。

在接收到 RESTful 方法更新後，M2M 伺服器使用 <requestInstance> 識別符來定位 <requestInstance>。M2M 伺服器可此後以所提供的資訊來填充這樣的 <requestInstance> 的屬性和/或子資源。如果用於填充任意給定的屬性和/或子資源的資訊沒有被提供，則 M2M 伺服器可將這樣的屬性和/或子資源保持初始創建的那樣（例如空白或具有預設屬性和/或參數）或如最後修改的那樣。

響應於 RESTful 方法更新，M2M 伺服器可向發佈者發送響應。M2M 伺服器可在更新 <requestInstance> 後發送該響應，如所示那樣。替代地，M2M 伺服器可在更新 <requestInstance> 期間發送該響應。M2M 伺服器還可向發佈者發送 ACK/NACK 訊息（未示出），以應答 RESTful 方法更新的接收。發佈者沒有接收到 ACK/NACK 訊息（例如在特定的時間內）可指示以 M2M 伺服器進行的 RESTful 方法更新的否定應答。

該響應可包括指示（例如代碼），用以指示 M2M 伺服器成功地更新 <requestInstance> 或未能更新 <requestInstance>。該指示可以是指示成功的更新的一個值，和指示失敗的另一個值。替代地，該響應可包括第一指示（例如代碼），以指示 M2M 伺服器成功地更新 <requestInstance>，和第二指示（例如代碼），以指示 M2M 伺服器未能更新 <requestInstance>。作為另一個替代選擇，M2M 伺服器可僅當該 M2M 伺服器成功地更新 <requestInstance> 時才發佈該響應。

該響應還可包括 <requestInstance> 識別符和/或所更新的屬性和/或子資源。該響應可進一步包括可被用以更新屬性和/或子資源的資訊或指示相同事件的確認。

第 29P 圖是示出用於從資源命令結構（例如 <requestInstance>）獲取資訊的示例訊息流的訊息流程圖。為了發起獲取，NA（即發佈者）向 M2M 伺服器發佈 RESTful 方法獲取。RESTful 方法獲取可包括 <requestInstance> 識別符。

在接收到 RESTful 方法獲取後，M2M 伺服器可使用 <requestInstance> 識別符來定位 <requestInstance>。一旦定位，M2M 伺服器可從具有這樣的可獲取資訊（此後稱為“經獲取的屬性資訊”和/或“經獲取的子資源資訊”）的 <requestInstance> 的屬性和/或子資源來查詢和獲得可獲取資訊（例如屬性、參數和/或引數）。經獲取的屬性和/或經獲取的子資源資訊可包括例如所儲存的 execDisable、execPause 和 execResume 識別符的任一個和/或來自其他屬性和/或子資源（例如 execStatus 和 execResult）的任意資訊。

在獲得經獲取的屬性和/或經獲取的子資源資訊後，M2M 伺服器可向 NA 發送響應。此響應可包括 <requestInstance> 識別符

和經獲取的屬性和/或經獲取的子資源資訊。

作為一個替代選擇，RESTful 方法獲取可被用以從具有這樣的可獲取資訊的<requestInstance>的屬性和/或子資源來獲取可獲取資訊的一個或多個選取部分。為了促進這事，RESTful 方法獲取可包括被分配給具有可獲取資訊的選取部分的伺服器<command>的屬性和/或子資源的每個節點的識別符。使用該識別符（或這些識別符，如多於一個的話），M2M 伺服器可定位、查詢和獲得所選取的經獲取的屬性和/或經獲取的子資源資訊。M2M 伺服器然後可向 NA 發送包括所選取的經獲取的屬性和/或經獲取的子資源資訊的響應。

雖然未示出，但 M2M 伺服器還可向發佈者發送 ACK/NACK 訊息，以應答 RESTful 方法獲取的接收。發佈者沒有接收到 ACK/NACK 訊息（例如在特定的時間內）可指示以 M2M 伺服器進行的 RESTful 方法獲取的否定應答。

第 29Q 圖是示出用於刪除資源命令結構（例如<requestInstance>）的示例訊息流的訊息流程圖。為了發起該刪除，發佈者可向 M2M 伺服器發佈 RESTful 方法刪除。發佈者可以是 NA、M2M 裝置或 M2M GW。M2M 裝置和/或 M2M GW 可響應於非 RESTful 命令的重啟、取消、去註冊等而發佈 RESTful 方法刪除。

RESTful 方法刪除可包括<requestInstance>識別符。替代地，RESTful 方法刪除可包括被分配給在其下已創建<requestInstance>實例集合的資源（例如<requestInstances>）的節點的識別符（此後稱為“<requestInstances>識別符”）。

在接收到 RESTful 方法刪除後，M2M 伺服器可使用<requestInstance>識別符或<requestInstances>識別符，以分別定位和刪除該<requestInstance>或<requestInstances>。這可包

括刪除<requestInstance>的所有屬性和/或子資源，或對於<requestInstances>來說，刪除這樣的<requestInstances>的每個<requestInstance>。

響應於 RESTful 方法刪除，M2M 伺服器可向發佈者發送響應。M2M 伺服器可在刪除<requestInstance>或<requestInstances>後發送該響應，如所示的那樣。替代地，M2M 伺服器可在刪除<requestInstance>或<requestInstances>的期間發送該響應。M2M 伺服器還可向發佈者發送 ACK/NACK 訊息（未示出），以應答 RESTful 方法刪除的接收。發佈者沒有接收到 ACK/NACK 訊息（例如在特定的時間內）可指示以 M2M 伺服器的 RESTful 方法刪除的否定應答。

該響應可包括指示（例如代碼），用以指示 M2M 伺服器成功地刪除<requestInstance>或<requestInstances>或未能刪除<requestInstance>或<requestInstances>。該指示可以是指示成功更新的一個值，和指示失敗的另一個值。替代地，該響應可包括第一指示（例如代碼），以指示 M2M 伺服器成功地刪除<requestInstance>或<requestInstances>，和第二指示（例如代碼），以指示未能刪除<requestInstance>或<requestInstances>。作為另一個替代選擇，M2M 伺服器可僅當該 M2M 伺服器成功地刪除<requestInstance>或<requestInstances>時才發佈該響應。

現在參考第 29R 圖，示出用於刪除例如伺服器<command>、伺服器<commands>和/或<requestInstance>的資源命令結構的示例訊息流的訊息流程圖。第 29R 圖的訊息流類似於第 29C 圖的訊息流和第 29I 圖（或第 29J 圖）的訊息流的組合，或第 29Q 圖的訊息流和第 29I 圖（或第 29J 圖）的訊息流的組合。如所示那樣，當發佈者向 M2M 伺服器發

佈 RESTful 方法刪除時，M2M 伺服器確定是否任意 <requestInstance> 正被用於命令執行或暫停命令執行，然後 M2M 伺服器可在刪除這樣的 <requestInstance> 之前執行它的取消命令執行。M2M 伺服器可根據第 29I 圖或第 29J 圖的訊息流來執行取消命令執行。

除了前述的之外，M2M 伺服器可創建和/或刪除伺服器 <command> 或另一個資源命令結構，而不發佈 RESTful 方法創建或 RESTful 方法刪除。

示例基於開道的機器對機器（“M2M”）裝置管理

根據一個實施方式，第 30A 圖示出了用於藉由槓桿操縱 OMA GwMO “透明”模式經由 M2M GW (G') 來管理 D 類型 ETSI M2M 裝置的示例架構。參考第 30A 圖，NREM 使用 GwMO 特定的介面和元件來從 GREM 獲取裝置資訊，為了執行此任務，GREM 反過來經由 GwMO 介面與 DREM 對話。然後，NREM 可使用 OMA DM 協定（用於訊息交換的用戶端-伺服器交互作用模型）直接地與 DREM 對話。應注意的是，在“透明”模式中，GREM 不需要支援 OMA DM 協定命令來管理裝置。

根據一個實施方式，第 30B 圖示出了 ETSI M2M xREM 的示例架構。參考第 30B 圖，xREM 被劃分為功能子模組。例如，REM 伺服器在裝置管理中扮演伺服器的角色（類似於 OMA DM 中的 DM 伺服器或 ACS）。REM 用戶端在裝置管理中扮演用戶端的角色（類似於在 OMA DM 中的 DM 用戶端或 CPE）。REM 代理元件僅位於分別與 REM 代理伺服器（在 NREM 中）和 REM 代理用戶端（在 DREM 中）對話的 GREM 中。REM 代理伺服器位於被用來與 REM 代理元件對話、以在 GW 後面得到裝置資訊的 NREM 中。REM 代理用戶端位於被用來對 REM 代理元件作出響應、以向

GW 報告最終被轉發給 NREM 的裝置資訊的 DREM 中。

在代理模式中，M2M GW 可作用如“中間人”一樣。它可支援 GwMO 和 DM 用戶端以及 DM 伺服器，如示出用於槓桿操縱 OMA GwMO 1.0 的示例架構的第 31A 圖所示。在此模式中，NREM 並未直接與 DREM 通信，反而由 GREM 轉換和中繼。對於 DREM，GREM 看起來像伺服器；對於 NREM，GREM 表現得像用戶端。此模式向 M2M GW 增加更多的特徵和功能。示例性的優點包括更多的加值服務（例如命令搗出（command fan-out）和響應聚集），且接下來在 M2M 核心中更低的業務負載，和更有效地用於管理受限的 M2M 裝置，特別是休眠的節點。第 31B 圖示出了用於這種模式的 xREM 的示例架構。

根據一個實施方式，M2M 裝置（D'）可能不具有 ETSI M2M 服務能力。可假設它們不具有 OMA DM 用戶端功能，但其他非 OMA 裝置管理協定具有。因此，M2M GW 可採用“適應”模式以在 OMA 協定和其他管理協定之間轉換。第 32A 圖示出用於槓桿操縱 OMA GwMO 1.0 的示例架構。第 32B 圖示出 ETSI M2M xREM 的示例架構。與“代理”模式相比，一些非 OMA 管理協議（例如 SNMP）可用於 dIa 介面或另一個 NDM-1 介面上的 M2M GW 和 M2M 裝置之間的交互作用。應注意，協定轉換不僅涉及管理命令映射/轉換，還涉及管理樹狀結構（或資源）映射/轉換等。如果 D' 類型裝置的確具有 OMA DM 用戶端功能，也可應用“透明”和“代理”模式。

根據一個實施方式，可假設 d 類型非 ETSI M2M 裝置可使用不同的管理協定。因此，M2M GW 可使用“適應”模式來管理它們。第 33A 圖和第 33B 圖示出的架構類似於第 32A 圖和第 32B 圖中的架構。應注意，d 類型裝置可以是非 ETSI

M2M GW，例如 ZigBee 協調器，儘管它非 ETSI M2M 區域網路可使用和存取 ETSI M2M 服務能力而被連接在一起。

在一個實施方式中，第 34 圖示出了槓桿操縱 OMA GwMO 的以 GW 為基礎的裝置管理的圖。如第 34 圖所示，單一的 M2M GW 可連接具有不同類型的 M2M 裝置的多個 M2M 區域網路。M2M GW 可為不同的區域網路施加透明、代理或適應管理模式。

在第 30 圖-第 34 圖的所有圖中，額外的 OMA GwMO 和 DM 邏輯實體（例如 OMA DM 伺服器）不必是 xREM 的一部分；但是相反的是，可以是 xREM 外的分離實體。然而也可應用本文公開的架構。例如，第 35 圖示出了用於部分緊密整合的 OMA DM 和 M2M GW 的示例架構。第 36 圖示出了用於鬆散整合的 OMA DM 和 M2M GW 的示例架構。

用於管理 M2M 區域網路和在 M2M 閘道後面的 M2M 裝置的示例資料模型

如以下更詳細地描述的那樣，用於管理機器對機器通信（M2M）區域網路和 M2M 閘道後面的 M2M 裝置的管理物件（MO）。M2M 區域網路管理可提供功能，例如裝置目錄和配置管理、區域網路配置管理、區域網路性能管理和裝置的組管理。這可包括應用（A）、M2M 直接裝置（D）、M2M 本地裝置（d 類型、D' 類型或 D 類型裝置）和 M2M 閘道（G）。

管理物件（MO）可被定義為以 M2M 閘道來管理 M2M 區域網路。一個這樣的 MO 可以是 etsiAreaNwkInfo 資源，其可以用於區域網路資訊和配置的管理物件。另一個 MO 可以是 etsiAreaNwkDeviceInventory 資源，其可以用於 M2M 本地裝置目錄的 MO。用於 M2M 本地裝置組的管理物件例如可以是 etsiAreaNwkDeviceGroups 資源。用於操作 M2M 本地裝置組的管理物件可以是

etsiGroupMgmtOperations 和 etsiAreaNwkGroupOperations 的任一個，並且用於整合到 M2M 裝置中的感測器的管理物件資源可以是 etsiSensors 資源。

MO 可以被組織和/或設置於不同的從屬等級。例如，MO 可以在 M2M 伺服器處 `<sclBase-of-Server>/scls/<GSCL>` 下的相同佔位的子資源。以示例的方式，在 `<sclBase-of-Server>/scls/<GSCL>/attachedDevices` 下的 `mgmtObjs` 子資源可被創建用於作為整體管理 M2M 開道後面的附屬裝置。該 MO 可例如設置在 `<sclBase-of-Server>/scls/<GSCL>/attachedDevices/mgmtObjs` 下。MO 可以是另一個 MO 的子資源。例如，`etsiSensors` 可以是 `<deviceInventory>` 的子資源，`<deviceInventory>` 可以是 `etsiAreaNwkDeviceInventory` 的子資源。

如第 37 圖所示，用於區域網路資訊和配置（`etsiAreaNwkInfo`）的 MO 可包括一個或多個屬性和複數子資源，例如訂閱集合和 `<areaNwkInstance>`。`<areaNwkInstance>` 可以是 M2M 區域網路的實例。`etsiAreaNwkInfo` 的屬性可包括例如期滿時間（`expirationTime`）、存取權 ID（`accessRightID`）、搜尋字串（`searchStrings`）、創建時間（`creationTime`）、最後修改時間（`lastModifiedTime`）、內容類型（`contentType`）、moID、原始 MO（`originalMO`）、區域網路數目（`numOfAreaNwks`）和描述（`description`）。`numOfAreaNwks` 屬性可以代表 M2M 區域網路的數目。`description` 屬性可以是 `mgmtObj` 的文本格式描述。該屬性（和/或變數）可遵循 ETSI M2M TS 102 690。作為一個替代選擇，該屬性和/或描述集合可包括、維護和/或儲存 `expirationTime`、`accessRightID`、`searchStrings`、

creationTime 、 lastModifiedTime 、 contentType 、 molD 、 originalMO 、 numOfAreaNwks 和 description 。

如第 38A 圖所示， etsiAreaNwkInfo 資源的 <areaNwkInstance>子資源可包括一個或多個屬性和複數子資源。<areaNwkInstance>子資源可包括 (i) areaNwkID 屬性，以儲放 M2M 區域網路的標識，(ii) areaNwkType 屬性，以指定 M2M 區域網路的類型，例如藍牙/6LoWPAN、802.15.4/6LoWPAN、Wi-Fi 網路，(iii) workingChannelFrequency 屬性，以指定 M2M 區域網路的工作頻道頻率和 (iv) addressMode 屬性，以指定 M2M 區域網路的位址模式（例如 IPv4、IPv6、短 MAC（媒體存取控制）、或長 MAC）。

<areaNwkInstance>子資源的複數子資源可包括描述集合、<deviceInstance>子資源、attachedDevices 子資源、groups 子資源、6LoWPAN 子資源、Wi-Fi 子資源、RFID 子資源和 ZigBee 子資源。

<deviceInstance>子資源可包括用於在區域網路實例中單一個裝置的資訊。attachedDevices 子資源可以用於區域網路的所有附屬裝置的佔位，並且 groups 子資源可以用於為組操作或操作搗出（operation fan-out）的所定義的裝置組的佔位。6LoWPAN 子資源可以用於包括與 6LoWPAN 網路相關的參數的佔位；因為，當區域網路是 6LoWPAN 網路時，這樣的資訊可被使用。Wi-Fi 子資源可以用於包括與 Wi-Fi 網路相關的參數的佔位；因為，當區域網路是 Wi-Fi 網路時，這樣的資訊可被使用。RFID 子資源可以用於包括與 RFID 網路相關的參數的佔位，因為，當區域網路是 RFID 網路時，可需要此資訊，並且 ZigBee 子資源可以用於包括與 ZigBee 網路相關的參數的佔位，因為，當區域網

路是 ZigBee 網路時，可需要此資訊。擴展也可被包含為子資源且可提供用於擴展的佔位。

與<areaNwkInstance>子資源相關聯的屬性可包括 (i) 裝置數目 (numOfDevice)，其可表示區域網路實例中裝置的數目；(ii) 睡眠間隔 (sleepInterval)，其可表示兩次睡眠之間的間隔；(iii) 睡眠持續時間 (sleepDuration)，其可表示每次睡眠的持續時間；(iv) MTU，其可表示在此區域網路<areaNwkInstance>子資源中的最大傳輸單元；和 (v) 區塊大小 (blockSize)，其可表示在 IETF CoAP 按區塊 (blockwise) 傳輸中所使用的區塊大小。當區域網路<areaNwkInstance>子資源支援受限應用協定 (CoAP 協定) 時，blockSize 將是有用的。sleepInterval 屬性可被用作一般時間間隔，經由 M2M 伺服器例如可指示 M2M 裝置和/或 M2M 開道週期地 (例如每個 sleepInterval 時間單元) 發送回特定的報告或響應。

如第 38B 圖所示，6LoWPAN 子資源可具有 subscriptions 子資源、subscription 和與在 6LoWPAN 中的定址、路由和相鄰的發現相關的複數屬性。複數屬性可包括 (i) 用於指定區域網路 IP 位址字首的 ipAddrPrefix 屬性，和 (ii) 用於指定 M2M 區域網路的路由模式的 routingMode 屬性。對於 6LoWPAN 網路而言，routingMode 屬性可將路由模式指定為 mesh-under 或 route-over。

複數屬性還可包括 (i) 用於指定在準備改變時繼續傳播標頭壓縮上下文資訊的最小時間的 minContextChangeDelay 屬性；(ii) 用於儲存和/或指定將即將發送的未經請求的 (unsolicited) 路由器通告的最大次數的 maxRtrAdvertisements 屬性；(iii) 用於指定發送給所有節點多播地址的兩次連貫的路由器通告之間的最小間隔的 minDelayBetweenRas 屬性；(iv) 用於指定用於發送路由器

通告訊息而作為對所接收的路由器請求訊息的響應的最大延遲的 `maxRaDelayTime` 屬性；(v) 用於儲存和提供試探鄰居快取記憶體計時器的 `entativeNceLifetime` 屬性；(vii) 用於儲存和/或指定複製位址檢測訊息的跳數限制值的 `hopLimit` 屬性；(viii) 用於儲存和/或指定第一 `maxRtxSolications` 路由器請求的初始重傳的間隔的 `rtrSolicitationInterval` 屬性；(ix) 用於儲存和/或指定如由 `rtrSolicitationInterval` 所定義的初始重傳的數目的 `maxRtrSolicitations` 屬性；和 (x) 用於儲存和/或指定路由器請求的最大重傳間隔的 `maxRtrSolicitationInterval` 屬性，因為裝置可在初始重傳後使用二進位指數倒回 (binary exponential backoff)。 `hopLimit`、`rtrSolicitationInterval`、`maxRtrSolicitations` 和 `maxRtrSolicitationInterval` 屬性可應用於裝置。其他參數可應用於裝置及 M2M 開道兩者。

如第 38C 圖所示，`etsiAreaNwkInfo` 資源的 `<areaNwkInstance>` 子資源可包括一個或多個屬性和 `subscriptions` 子資源。`<areaNwkInstance>` 子資源可包括 (i) `areaNwkID` 屬性，用以儲存 M2M 區域網路的標識；(ii) `areaNwkType` 屬性，用以指定 M2M 區域網路的類型，例如藍牙/6LoWPAN、802.15.4/6LoWPAN、Wi-Fi 網路；(iii) `workingChannelFrequency` 屬性，用以指定 M2M 區域網路的工作頻道頻率；和 (iv) `addressMode` 屬性，用以指定 M2M 區域網路的位址模式（例如 IPv4、IPv6、短 MAC（媒體存取控制）或長 MAC）；(v) 6LoWPAN 屬性、Wi-Fi 屬性、RFID 屬性、ZigBee 屬性。

6LoWPAN 屬性可以用於包括與 6LoWPAN 網路相關的參數的佔位，例如當區域網路是 6LoWPAN 網路時可使用的資訊。Wi-Fi 屬性可以用於包括與 Wi-Fi 網路相關的參

數的佔位，例如當區域網路是 Wi-Fi 網路時可使用的資訊。RFID 屬性可以用於包括與 RFID 網路相關的參數的佔位，並且 ZigBee 屬性可以用於包括與 ZigBee 網路相關的參數的佔位。

此外，<areaNwkInstance>子資源可具有與特定類型的區域網路相關的更多屬性。例如，如果 M2M 區域網路是 6LoWPAN 網路，<areaNwkInstance>子資源可包括與在 6LoWPAN 中定址、路由和/或相鄰發現相關的若干屬性。複數屬性可包括 (i) ipAddrPrefix 屬性；和 (ii) routingMode 屬性；(iii) minContextChangeDelay 屬性；(iv) maxRtrAdvertisements 屬性；(v) minDelayBetweenRas 屬性；(vi) maxRaDelayTime 屬性；(vii) tentativeNceLifetime 屬性；(ix) hopLimit 屬性；(x) rtrSolicitationInterval 屬性；(xi) 用於儲存和/或指定如由 rtrSolicitationInterval 所定義的初始重傳的數目的 maxRtrSolicitations 屬性；和 (xii) maxRtxSolicitationInterval 屬性。hopLimit、rtrSolicitationInterval、maxRtrSolicitations 和 maxRtrSolicitationInterval 屬性可應用於裝置。其他參數既可應用於裝置及 M2M 閘道兩者。

如第 39 圖所示，etsiAreaNwkDeviceInventory MO 可具有子資源，包括 (i) <deviceInstance>子資源，其可包括用於附屬於 M2M 閘道的每個活動裝置各自<deviceInstance>資訊的集合；(ii) <areaNwkInstance>子資源，其可標識每個區域網路；(iii) groups 子資源，其可提供用於經定義的裝置組的佔位；和 (iv) subscriptions 集合子資源。Subscriptions 集合子資源可包括屬性，例如 expirationTime、accessRightID、searchStrings、creationTime、lastModifiedTime、可被包括在 FSS 格式中的 contentType、

moID、originalMO、description（例如 mgmtObj 的文本格式描述）和附屬於 M2M 開道的裝置的數目（numOfDevices）。

在相同 M2M 開道下的 M2M 區域網路可對應於單一的 etsiAreNwkDeviceInventory 子資源，或者每個區域網路具有它自己的 etsiAreNwkDeviceInventory 子資源。etsiAreNwkDeviceInventory 子資源可從屬於 etsiAreaNwkInfo 子資源。作為一個替代選擇，etsiAreNwkDeviceInventory MO 可不包括 <areaNwkInstance> 子資源和/或 gourps 子資源。

如第 40A 圖所示，每個 <deviceInstance> 可包括例如 deviceGroupsList 的該裝置所屬組的標識列表、在此裝置上的例如 D'A 或 DA 的所有應用的應用標識的組列表 deviceApplicationList、鄰居節點的標識列表 deviceNeighborsList、電池資訊 etsiBattery、記憶體資訊 etsiMemory 和感測器/激發器資訊 estiSensor、可提供用於包括與 6LoWPAN 網路相關的參數的佔位的 6LoWPAN、可提供用於包括與 Wi-Fi 網路相關的參數的佔位的 Wi-Fi、可提供用於包括與 RFID 網路相關的參數的佔位的 RFID、可提供用於包括與 ZigBee 網路相關的參數的佔位的 ZigBee 和可以是用於擴展的佔位的 extensions。

此組可提供標準組資源，例如在 ETSI TS 102 609 中所提出。<deviceInstance> 可包括屬性，例如可以是裝置類型的 deviceTpye、可以是裝置識別符的 deviceID、可以是裝置的位址類型的 addressType、可包括該裝置所屬 M2M 區域網路的標識的 areaNwkID、可以是在 M2M 區域網路內部使用的裝置的內部 IP 位址的內部位址、和可以是在 M2M 區域網路外部使用的裝置的外部 IP 位址的外部位址。如果埠號在 M2M 開道處用於位址轉換，此位址可包括埠資訊。另外，

兩次睡眠之間的間隔 `sleepInterval`、每次睡眠的持續時間 `sleepDuration`、裝置的狀態 `status`（例如睡眠或喚醒）、可包括在區域網路中最大傳輸單元的 MTU 和可包括在 IETF CoAP 按區塊傳輸中使用的區塊大小的 `blockSize`。

如第 40B 圖所示，每個 `<deviceInstance>` 可包括例如 `deviceGroupsList` 的該裝置所屬組的標識列表、在此裝置上的例如 D'A 或 DA 的所有應用的應用標識的組列表 `deviceApplicationList`、鄰居節點的標識列表 `deviceNeighborsList`、電池資訊 `etsiBattery`、記憶體資訊 `etsiMemory` 和感測器/激發器資訊 `estiSensor`。描述集合可包括屬性，例如可以是裝置類型的 `deviceType`、可以是裝置識別符的 `deviceId`、可以是裝置的位址類型的 `addressType`、可包括該裝置所屬 M2M 區域網路的標識的 `areaNwkID`、可以是在 M2M 區域網路內部使用的裝置的內部 IP 位址的內部位址、和可以是在 M2M 區域網路外部使用的裝置的外部 IP 位址的外部位址。如果埠號在 M2M 閘道處用於位址轉換，此位址可包括埠資訊。另外，兩次睡眠之間的間隔 `sleepInterval`、每次睡眠的持續時間 `sleepDuration`、裝置的狀態（例如睡眠或喚醒）`status`、即將發送的未經請求的路由器通告的最大數目 `maxRtrAdvertisements`、發送給所有節點多播地址的兩個連貫的路由器通告之間的最小間隔 `minDelayBetweenRas`、用於發送作為響應於所接收的路由器請求訊息的路由器通告訊息的最大延期 `maxRaDelayTime`、用於試探鄰居快取記憶體的計時器 `tentativeNceLifetime`、用於複製位址檢測訊息的跳數限制值 `hopLimit`、用於第一 `maxRtxSolications` 路由器請求的初始重傳的間隔的 `rtrSolicitationInterval`、初始重傳的數目 `maxRtrSolicitations` 和用於路由器請求的最大重傳間隔

maxRtrSolicitationInterval，因為裝置可在初始重傳後使用二進位指數倒回。

如第 41A 圖所示，etsiAreaNwkDeviceGroups MO 可包括例如定義裝置組的 <deviceGroupInstance> 的子資源和 subscriptions。該訂閱的集合和屬性可包括 expirationTime、accessRightID、searchStrings、creationTime、lastModifiedTime、可以被格式化為 FFS 的 contentType、moID、originalMO 和包括 mgmtObj 文本格式描述的 description。

如第 41B 圖所示，etsiAreaNwkDeviceGroups MO 可包括子資源 descriptions 和被定義為多個裝置組的集合的組。etsiAreaNwkDeviceGroups 可包括屬性，例如 expirationTime、accessRightID、searchStrings、creationTime、lastModifiedTime、可以被格式化為 FFS 的 contentType、moID、originalMO 和包括 mgmtObj 文本格式描述的 description。

每個子資源 <group> 可包括屬於相同組的裝置的標識列表。此外，裝置可屬於和存在於多個 <group> 實例。

如第 42 圖所示，每個 <deviceGroupInstance> 可包括例如 subscriptions 的子資源，具有訂閱的集合和屬性（包括 expirationTime、accessRightID、searchStrings、creationTime、lastModifiedTime、可以被格式化為 FFS 的 contentType、moID、originalMO 和包括 mgmtObj 文本格式描述的 description）。另外，groupID 可包括組標識、可包括組類型的 groupType、例如在組中裝置的數目的 groupSize、定義為在組中裝置集合的 members 和指定要為此組成員的裝置的條件的 condition。

如第 43A 圖所示，etsiGroupMgmtOperations MO 可包括

例如<operationInstance>的子資源，它可代表即將在組或多個組上執行的操作或動作，其可包括用於每個裝置組（包括裝置列表）的經定義的裝置組的佔位，並且可由一個或多個<operationInstance>操作。訂閱集合和屬性可包括expirationTime、accessRightID、searchStrings、creationTime、lastModifiedTime、contentType、moID、originalMO 和包括mgmtObj 文本格式描述的 description。

除了被用來管理/操作在 M2M 閘道後面的 M2M 裝置組以外，etsiGroupMgmtOperations 可被用來管理與 M2M 伺服器直接連接的 M2M 裝置組的操作。在此情況下，etsiGroupMgmtOperations 可設置在<sclBase-of-Server>/scls/mgmtObjs/etsiGroupMgmtOperations 下。相應的<operationInstance>可包括去註冊 M2M 裝置或閘道組、將 M2M 裝置或閘道組發送至睡眠模式、重啟 M2M 裝置或閘道組和在 M2M 裝置或閘道組上進行相同的軟體/韌體更新。

同樣地，etsiGroupMgmtOperations 可被用以管理在 M2M 裝置或 M2M 閘道上的應用組。在此情況下，etsiGroupMgmtOperations 可設置在<sclBase-of-Server>/scls/<scl>/applications/mgmtObjs/etsiGroupMgmtOperations 下。相應的<operationInstance>可包括去註冊 M2M 應用組和在 M2M 應用組上進行相同的軟體/韌體更新。

如第 43B 圖所示，etsiAreaNwkDeviceGroupOperations MO 可包括例如<operationInstance>的子資源，其可代表即將在組上執行的操作或動作。訂閱集合和屬性可包括expirationTime、accessRightID、searchStrings、creationTime、lastModifiedTime、contentType、moID、originalMO 和包括

mgmtObj 文本格式描述的 description。

如第 44A 圖所示，每個<operationInstance>可包括子資源，例如具有訂閱集合和屬性（包括 expirationTime、accessRightID、searchStrings、creationTime、lastModifiedTime、可在 FSS 格式中的 contentType、moID、originalMO 和包括 mgmtObj 文本格式描述的 description）的 subscriptions。<operationInstance>可包括為子資源組，其可包括可以是<operationInstance>可在其上執行的組的標識列表 deviceGroupsList、可以是<operationInstance>集合結果的 execResults，並且可具有包括<resultInstance>和訂閱以及屬性 aggregatedResult 的其他子資源。

<resultInstance>可表示來自單一裝置的結果，並且可具有子資源 subscriptions 和屬性，屬性包括可以是裝置標識的 deviceID、可以是來自裝置 deviceID 的結果的 resultValue、可以是在裝置 deviceID 上的<operationInstance>的狀態的 execStatus、可在 deviceID 上開始<operationInstance>的 execEnable、可在 deviceID 上暫停<operationInstance>的 execPasue、可在 deviceID 上恢復<operationInstance>的 execResume、可在 deviceID 上停止<operationInstance>的 execDisable、可以是聚集結果的 aggregatedResult、可包括<operationInstance>所需的引數並且可是操作上特定的 execParameters。

另外，可以是<operationInstance>的識別並且指定<operationInstance>代表什麼的 operationID。groupID 可提供在其上執行<operationInstance>的組的標識，並且如果<operationInstance>可在多個經定義的裝置組上執行，則 groupID 可包括多個識別。替代地，那些多個組識別可被包括在 deviceGroupsList 資源中。execEnable 可在 groupID 中

開始裝置上的<operationInstance>。execPause 可在 groupId 中暫停裝置上的<operationInstance>，execResume 可在 groupId 中恢復裝置上的<operationInstance>，execDisable 可在 groupId 中停止裝置上的<operationInstance>，execStatus 可提供<operationInstance>的狀態。該狀態可包括未決、運行、停止、暫停、恢復、在裝置上成功地執行、完成且成功的裝置數目和/或在其上未能被執行的裝置數目。

如第 43B 圖所示，每個<operationInstance>可包括子資源，例如具有訂閱集合和屬性（包括 expirationTime、accessRightID、searchStrings、creationTime、lastModifiedTime、可在 FSS 格式中的 contentType、moID、originalMO 和包括 mgmtObj 文本格式描述的 description）的 subscriptions。另外，groupId 可提供在其上可被執行<operation>的組標識，致能（enable）可被提供以開始<operation>，失效（disable）以停止<operation>，結果（results）可包括<operation>的集合結果，並且 aggregatedResult 可包括聚集的結果。每個<resultInstance>可具有兩個屬性，例如可指示裝置標識及結果的 deviceId，result 可以是來自裝置 deviceId 的結果。

如第 44 圖所示，etsiSensors MO 可包括子資源，例如用於感測器實例的<sensorInstance>和包括訂閱集合和屬性的 subscriptions，例如 expirationTime、accessRightID、searchStrings、creationTime、lastModifiedTime、可在 FSS 格式中的 contentType、moID、originalMO 和包括 mgmtObj 文本格式描述的 description。etsiSensors 可應用於具有 M2M 服務能力的 D 類型 M2M 裝置。

如第 45 圖所示，<sensorInstance>可包括子資源（例如具有相關組的組）。例如，一個組可以是 applicationList，用

以代表在使用<sensorInstance>的此裝置上的 D'A 或 DA 應用。容器可儲存感測器讀取。訂閱集合和屬性可包括 expirationTime、accessRightID、searchStrings、creationTime、lastModifiedTime、可在 FSS 格式中的 contentType、moID、originalMO 和包括 mgmtObj 文本格式描述的 description。sensorID 可描述<sensorInstance>的標識。sensorType 可描述<sensorInstance>的類型，例如溫度、壓力、定義<sensorInstance>製造商的製造商和操作，操作可包括能在<sensorInstance>上操作的操作。致能（enable）可包括致能感測器讀取。如果它是開關感測器，致能可表示打開開關。致能可具有作為它的屬性的結果，以儲存操作結果。失效（disable）可包括失效感測器讀取。如果它是開關感測器，失效可表示關閉開關。失效可具有作為它的屬性的結果，以儲存操作結果。其他操作對於特定的感測器是可能的。

示例操作環境

第 46 圖是用於根據由 DA 和/或 GA 使用的管理物件來執行 REM 以管理向 M2M 伺服器（在此是<scl>）註冊的另一個 D/G 的系統的示例架構的框圖。

M2M 伺服器（即<scl>）可向 D/G 通知它的管理物件，如在 4702 所示。然後 DA/GA 可存取在 D/G 中的這樣的經通知的管理物件，並且然後能夠經由在 M2M 伺服器處的訊息中繼而管理其他 D/G。

第 48A 圖是在其中一個或多個公開實施方式可被實現的示例通信系統 100 的圖。通信系統 100 可以是向多個無線用戶提供例如語音、資料、視頻、訊息、廣播等內容的多存取系統。通信系統 100 可使多個無線用戶能夠經由共享的系統資源（包括無線頻寬）來存取這樣的內容。例如，通信系統 100 可採用一個或多個頻道存取方法，例如碼分多址

(CDMA)、時分多址 (TDMA)、頻分多址 (FDMA)、正交 FDMA (OFDMA)、單載波 FDMA (SC-FDMA) 等。

如第 48A 圖所示，通信系統 100 可包括無線發射/接收單元 (WTRU) 102a、102b、102c、102d、無線電存取網路 (RAN) 104、核心網路 106、公共交換電話網路 (PSTN) 108、網際網路 110 和其他網路 112，雖然將理解的是所公開的實施方式考量任意數目的 WTRU、基站、網路和/或網路元件。WTRU 102a、102b、102c、102d 的每一個可以是任意類型的、被配置以在無線環境中操作和/或通信的裝置。以示例的方式，WTRU 102a、102b、102c、102d 可被配置以發送和/或接收無線信號，並且可包括用戶設備 (UE)、移動站、固定或移動用戶單元、呼叫器、行動電話、個人數位助理 (PDA)、智慧型電話、膝上型電腦、輕省筆電 (netbook)、個人電腦、無線感測器、消費電子產品等。

通信系統 100 還可包括基站 114a 和基站 114b。基站 114a、114b 的每一個可以是任意類型的、被配置以與 WTRU 102a、102b、102c、102d 的至少一個無線介面的裝置，以促進存取一個或多個通信網路 (例如核心網路 106、網際網路 110 和/或網路 112)。以示例的方式，基站 114a、114b 可以是基地收發信機站台 (BTS)、節點 B、e 節點 B、家用節點 B、家用 e 節點 B、站點控制器、存取點 (AP)、無線路由器等。雖然基站 114a、114b 每一個被圖示為單一元件，應理解的是基站 114a、114b 可包括任意數目的互連基站和/或網路元件。

基站 114a 可以是 RAN 104 的一部分，RAN 104 還可包括其他基站和/或網路元件 (未示出)，例如基站控制器 (BSC)、無線電網路控制器 (RNC)、中繼節點等。基站 114a 和/或基站 114b 可被配置以在可被稱為胞元 (未示出) 的特

定地理區域內發送和/或接收無線信號。胞元可進一步被劃分為胞元扇區。例如，與基站 114a 相關聯的胞元可被劃分為 3 個扇區。因此，在一個實施方式中，基站 114a 可包括 3 個收發器，即胞元的每個扇區使用一個收發器。在另一個實施方式中，基站 114a 可使用多輸入多輸出 (MIMO) 技術，因此可為胞元的每個扇區使用多個收發器。

基站 114a、114b 可在空氣介面 116 與 WTRU 102a、102b、102c、102d 的一個或多個通信，空氣介面 116 可以是任意適當的無線通信鏈路（例如射頻 (RF)、微波、紅外線 (IR)、紫外線 (UV)、可見光等）。空氣介面 116 可使用任意適當的無線電存取技術 (RAT) 來建立。

更具體地，如上所述，通信系統 100 可以是多存取系統，並且可使用一個或多個頻道存取方案，例如 CDMA、TDMA、FDMA、OFDMA、SC-FDMA 等。例如，RAN 104 中的基站 114a 和 WTRU 102a、102b、102c 可執行例如通用行動通信系統 (UMTS) 地面無線電存取 (UTRA) 的無線電技術，其可使用寬頻 CDMA (WCDMA) 來建立空氣介面 116。WCDMA 可包括例如高速封包存取 (HSPA) 和/或演進型 HSPA (HSPA+) 的通信協定。HSPA 可包括高速下行鏈路封包存取 (HSDPA) 和/或高速上行鏈路封包存取 (HSUPA)。

在另一個實施方式中，基站 114a 和 WTRU 102a、102b、102c 可執行例如演進型 UMTS 地面無線電存取 (E-UTRA) 的無線電技術，其可使用長期演進 (LTE) 和/或高級 LTE (LTE-A) 來建立空氣介面 116。

在其他實施方式中，基站 114a 和 WTRU 102a、102b、102c 可執行例如 IEEE 802.16（即全球互通微波存取 (WiMAX)）、CDMA2000、CDMA2000 1X、CDMA2000

EV-DO、臨時標準 2000 (IS-2000)、臨時標準 95 (IS-95)、臨時標準 856 (IS-856)、全球行動通信系統 (GSM)、增強型資料速率 GSM 演進技術(EDGE)、GSM EDGE(GERAN)等無線電技術。

第 48A 圖中的基站 114b 可以是例如無線路由器、家用節點 B、家用 e 節點 B 或存取點，並且可使用任意適當的 RAT 以促進局部區域中（例如商業地點、家庭、車輛、校園等）的無線連接性。在一個實施方式中，基站 114b 和 WTRU 102c、102d 可執行例如 IEEE 802.11 的無線電技術，以建立無線局部區域網路（WLAN）。在另一個實施方式中，基站 114b 和 WTRU 102c、102d 可執行例如 IEEE 802.15 的無線電技術，以建立無線個人區域網路（WPAN）。仍然在另一個實施方式中，基站 114b 和 WTRU 102c、102d 可使用基於胞元的 RAT（例如 WCDMA、CDMA2000、GSM、LTE、LTE-A 等）來建立微微胞元（picocell）或毫微微胞元（femtocell）。如第 48A 圖所示，基站 114b 可與網際網路 110 有直接連接。因此，基站 114b 不需要經由核心網路 106 來存取網際網路 110。

RAN 104 可與核心網路 106 通信，核心網路 106 可以是任意類型的、被配置以向 WTRU 102a、102b、102c、102d 的一個或多個提供語音、資料、應用和/或經由網際網路協定的語音（VoIP）服務的網路。例如，核心網路 106 可提供呼叫控制、計費服務、基於行動位置的服務、預付費呼叫、網際網路連接、視頻發佈等，和/或執行例如用戶認證的高級安全功能。雖然未在第 48A 圖中示出，應理解的是 RAN 104 和/或核心網路 106 可與使用與 RAN 104 相同 RAT 或不同 RAT 的其他 RAN 直接或間接通信。例如，除了與可採用 E-UTRA 無線電技術的 RAN 104 連接之外，核心網路 106

還可使用 GSM 無線電技術與另一個 RAN (未示出) 通信。

核心網路 106 還可作為用於 WTRU 102a、102b、102c、102d 的閘道，以存取 PSTN 108、網際網路 110 和/或其他網路 112。PSTN 108 可包括提供普通老式電話服務 (POTS) 的電路交換電話網絡。網際網路 110 可包括使用通用通信協定 (例如 TCP/IP 網際網路協定系列中的傳輸控制協定 (TCP)、用戶資料報協定 (UDP) 和網際網路協定 (IP)) 的互連電腦網路和裝置的全球系統。網路 112 可包括由其他服務供應商所有和/或營運的有線或無線通信網路。例如，網路 112 可包括與可使用與 RAN 104 相同 RAT 或不同 RAT 的一個或多個 RAN 相連接的另一個核心網路。

在通信系統 100 中的 WTRU 102a、102b、102c、102d 的一些或所有可包括多模式能力，例如 WTRU 102a、102b、102c、102d 可包括用於經由多個無線鏈路與不同無線網路通信的多個收發器。例如，第 48A 圖中示出的 WTRU 102c 可被配置以與可使用基於胞元的無線電技術的基站 114a 和與可使用 IEEE 802 無線電技術的基站 114b 通信。

第 48B 圖是示例 WTRU 102 的系統圖。如第 48B 圖所示，WTRU 102 可包括處理器 118、收發器 120、發射/接收元件 122、揚聲器/麥克風 124、鍵盤 126、顯示器/觸控板 128、非移除式記憶體 130、可移除式記憶體 132、電源 134、全球定位系統 (GPS) 晶片組 136 和其他週邊設備 138。應理解的是，WTRU 102 可包括前述元件的任意子組合，而與實施方式保持一致。

處理器 118 可以是通用處理器、專用處理器、傳統處理器、數位信號處理器 (DSP)、複數微處理器、與 DSP 核心相關聯的一個或多個微處理器、控制器、微控制器、專用積體電路 (ASIC)、現場可編程閘陣列 (FPGA) 電路、任意

其他類型的積體電路 (IC)、狀態機等。處理器 118 可執行信號編碼、資料處理、功率控制、輸入/輸出處理和/或使 WTRU 102 能夠在無線環境中操作的任意其他功能。處理器 118 可與收發器 120 相耦合，收發器 120 可與發射/接收元件 122 相耦合。雖然第 48B 圖將處理器 118 和收發器 120 圖示為分離的部件，將理解的是處理器 118 和收發器 120 可在電子封裝或晶片中整合在一起。

發射/接收元件 122 可被配置以經由空氣介面 116 向基站（例如基站 114a）傳送信號或從基站接收信號。例如，在一個實施方式中，發射/接收元件 122 可以是被配置以傳送和/或接收 RF 信號的天線。在另一個實施方式中，發射/接收元件 122 可以是被配置以例如發送和/或接收 IR、UV 或可見光信號的發射器/檢測器。在另一個其他實施方式中，發射/接收元件 122 可以被配置以發送和接收 RF 和光信號兩者。將理解的是，發射/接收元件 122 可被配置以傳送和/或接收無線信號的任意組合。

此外，雖然發射/接收元件 122 在第 48B 圖中被圖示為單一元件，WTRU 102 可包括任意數目的發射/接收元件 122。更具體地，WTRU 102 可使用 MIMO 技術。因此，在一個實施方式中，WTRU 102 可包括兩個或更多個用於經由空氣介面 116 傳送和接收無線信號的發射/接收元件 122（例如多個天線）。

收發信機 120 可被配置以調制即將由發射/接收元件 122 傳送的信號並解調由發射/接收元件 122 接收的信號。如上所述，WTRU 102 可具有多模式能力。因此，收發信機 120 可包括例如用於使 WTRU 102 能夠經由例如 UTRA 和 IEEE 802.11 的多個 RAT 進行通信的多個收發信機。

WTRU 102 的處理器 118 可與揚聲器/麥克風 124、鍵盤

126 和/或顯示器/觸控板 128 (例如液晶顯示器 (LCD) 顯示單元或有機發光二極體 (OLED) 顯示單元) 相耦合, 並可從它們接收用戶輸入資料。處理器 118 還可以向揚聲器/麥克風 124、鍵盤 126 和/或顯示器/觸控板 128 輸出用戶資料。此外, 處理器 118 可從例如非移除式記憶體 130 和/或可移除式記憶體 132 的任意類型的適當記憶體存取資訊, 並將資料儲存在其中。非移除式記憶體 130 可包括隨機存取記憶體 (RAM)、唯讀記憶體 (ROM)、硬碟或任意其他類型的記憶體儲存設備。可移除式記憶體 132 可包括用戶識別模組 (SIM) 卡、記憶棒、安全數位 (SD) 記憶卡等。在其他實施方式中, 處理器 118 可從物理上不位於 WTRU 102 上 (例如在伺服器或家用電腦 (未示出) 上) 的記憶體存取資訊, 並將資料儲存在其中。

處理器 118 可從電源 134 接收功率, 並可被配置以分配和/或控制給 WTRU 102 中其他元件的功率。電源 134 可以是任意適當的用於向 WTRU 102 供電的設備。例如, 電源 134 可包括一個或多個乾電池 (鎳鎘 (NiCd)、鎳鋅 (NiZn)、鎳金屬氫化物 (NiMH)、鋰離子 (Li-ion) 等)、太陽能電池、燃料電池等。

處理器 118 還可以與可被配置以提供關於 WTRU 102 當前位置的位置資訊 (例如經度和緯度) 的 GPS 晶片組 136 相耦合。除了來自 GPS 晶片組 136 的資訊或替代來自 GPS 晶片組 136 的資訊, WTRU 102 可經由空氣介面 116 從基站 (例如基站 114a、114b) 接收位置資訊, 和/或基於信號從兩個或更多個附近基站所接收的時序來確定它的位置。將理解的是, WTRU 102 可經由任何適當的位置確定方法來獲取位置資訊而與實施方式保持一致。

處理器 118 可進一步與其他週邊設備 138 相耦合, 其他

週邊設備 138 可包括提供附加特徵、功能和/或有線或無線連接的一個或多個軟體和/或硬體模組。例如，週邊設備 138 可包括加速計、電子羅盤、衛星收發器、數位照相機（用於相片或視頻）、通用串列匯流排（USB）埠、振動設備、電視收發器、免持聽筒耳機、藍牙®模組、調頻（FM）無線電單元、數位音樂播放器、媒體播放器、視頻遊戲玩家模組、網際網路瀏覽器等。

第 48C 圖是根據一個實施方式 RAN 104 和核心網路 106 的系統圖。如上所述，RAN 104 可使用 UTRA 無線電技術而通過空氣介面 106 與 WTRU 102a、102b、102c 通信。RAN 104 還可以與核心網路 106 通信。如第 48C 圖所示，RAN 104 可包括節點-B 140a、140b、140c，節點-B 140a、140b、140c 的每一個可包括經由空氣介面 116 而與 WTRU 102a、102b、102c 通信的一個或多個收發器。節點-B 140a、140b、140c 的每一個可與在 RAN 104 中的特定胞元（未示出）相關聯。RAN 104 還可包括 RNC 142a、142b。將理解的是，RAN 104 可包括任意數目的節點-B 和 RNC 而保持與實施方式一致。

如第 48C 圖所示，節點-B 140a、140b 可與 RNC 142a 通信。附加地，節點-B 140c 可與 RNC 142b 通信。節點-B 140a、140b、140c 可經由 Iub 介面而與各自的 RNC 142a、142b 通信。RNC 142a、142b 可經由 Iur 介面互相通信。RNC 142a、142b 的每一個可被配置以控制與其連接的各個節點-B 140a、140b、140c。此外，RNC 142a、142b 的每一個可被配置以執行或支援其他功能，例如外環功率控制、負載控制、許可控制、封包調度、移交控制、宏分集、安全功能、資料加密等。

第 48C 圖所示的核心網路 106 可包括媒體閘道(MGW) 144、行動交換中心(MSC)146、服務 GPRS 支援節點(SGSN)

148 和/或開道 GPRS 支援節點 (GGSN) 150。雖然上述元件的每一個都被圖示為核心網路 106 的一部分，將理解的是，這些元件的任意一個可由不是核心網路營運商的實體所有和/或營運。

RAN 104 中的 RNC 142a 可經由 IuCS 介面與核心網路 106 中的 MSC 146 相連接。MSC 146 可與 MGW 144 相連接。MSC 146 和 MGW 144 可向 WTRU 102a、102b、102c 提供到電路交換網路（例如 PSTN 108）的存取，以促進於 WTRU 102a、102b、102c 和傳統陸地通信設備之間的通信。

RAN 104 中的 RNC 142a 還可以經由 IuPS 介面與核心網路 106 中的 SGSN 148 相連接。SGSN 148 可與 GGSN 150 相連接。SGSN 148 和 GGSN 150 可向 WTRU 102a、102b、102c 提供到封包交換網路（例如網際網路 110）的存取，以促進於在 WTRU 102a、102b、102c 和致能 IP 的設備之間的通信。

如上所述，核心網路 106 還可與網路 112 相連接，網路 112 可包括由其他服務供應商所有和/或營運的其他有線或無線網路。

第 48D 圖是示出根據一個實施方式的 RAN 104 和核心網路 106 的系統圖。如上所述，RAN 104 可使用 E-UTRA 無線電技術來經由空氣介面 116 而與 WTRU 102a、102b、102c 通信。RAN 104 還與核心網路 106 通信。

RAN 104 可包括 e 節點-B 140a、140b、140c，雖然將理解的是 RAN 104 可包括任意數目的 e 節點-B 而與實施方式保持一致。e 節點-B 140a、140b、140c 每一個可包括用於經由空氣介面 116 而與 WTRU 102a、102b、102c 通信的一個或多個收發器。在一個實施方式中，e 節點-B 140a、140b、140c 可執行 MIMO 技術。因此 e 節點-B 140a 例如可使用多

個天線來向 WTRU 102a 發送無線信號並從它接收無線信號。

e 節點-B 140a、140b、140c 的每一個可與特定的胞元(未示出)相關聯，並且可被配置以處理無線電資源管理決策、移交決策、調度在上行鏈路和/或下行鏈路中的用戶等。如第 48D 圖所示，e 節點-B 140a、140b、140c 可經由 X2 介面互相通信。

第 48D 圖中示出的核心網路 106 可包括行動管理閘道 (MME) 142、服務閘道 144 和封包資料網路 (PDN) 閘道 146。雖然上述元件被圖示為核心網路 106 的一部分，但將理解的是這些元件的任意一個可由除了核心網路營運商以外的實體所有和/或營運。

MME 142 可經由 S1 介面與 RAN 104 中的 e 節點-B 140a、140b、140c 的每一個相連接，並且可作為控制節點。例如，MME 142 可負責認證 WTRU 102a、102b、102c 的用戶、承載啟動/去啟動、在 WTRU 102a、102b、102c 初始附屬期間選擇特定的服務閘道等。MME 142 還可提供用於在 RAN 104 和使用其他無線電技術 (例如 GSM 或 WCDMA) 的其他 RAN (未示出) 之間切換的控制面功能。

服務閘道 144 可經由 S1 介面與 RAN 104 中的 e 節點-B 140a、140b、140c 的每一個相連接。服務閘道 144 通常可路由和轉發用戶資料封包到 WTRU 102a、102b、102c/來自 WTRU 102a、102b、102c 的用戶資料封包。服務閘道 144 還可以執行其他功能，例如在 e 節點 B 間移交期間錨定用戶面、當下行鏈路數據對 WTRU 102a、102b、102c 可用時觸發呼叫、管理和儲存 WTRU 102a、102b、102c 的上下文等。

服務閘道 144 還可與 PDN 閘道 146 相連接，PDN 閘道

146 可向 WTRU 102a、102b、102c 提供到例如網際網路 110 的封包切換網路的存取，以促進於 WTRU 102a、102b、102c 和致能 IP 的裝置之間的通信。

核心網路 106 可促進於與其他網路的通信。例如，核心網路 106 可向 WTRU 102a、102b、102c 提供到例如 PSTN 108 的電路交換網路的存取，以促進於 WTRU 102a、102b、102c 和傳統陸地通信設備之間的通信。例如，核心網路 106 可包括作為核心網路 106 和 PSTN 108 之間的介面的 IP 閘道（例如 IP 多媒體子系統（IMS）伺服器）或與 IP 閘道通信。此外，核心網路 106 可向 WTRU 102a、102b、102c 提供到網路 112 的存取，網路 112 可包括由其他服務供應商所有和/或營運的其他有線或無線網路。

第 48E 圖是示出根據實施方式的 RAN 104 和核心網路 106 的系統圖。RAN 104 可以是使用 IEEE 802.16 無線電技術以經由空氣介面 116 與 WTRU 102a、102b、102c 通信的存取服務網路（ASN）。如將於下文進一步討論的那樣，在 WTRU 102a、102b、102c、RAN 104 和核心網路 106 的不同功能實體之間的通信鏈路可被定義為參考點。

如第 48E 圖所示，RAN 104 可包括基站 140a、140b、140c 和 ASN 閘道 142，雖然將理解的是 RAN 104 可包括任意數目的基站和 ASN 閘道而與實施方式保持一致。基站 140a、140b、140c 的每一個可與 RAN 104 中的特定胞元（未示出）相關聯，並且每一個可包括經由空氣介面 116 與 WTRU 102a、102b、102c 通信的一個或多個收發器。在多個實施方式中，基站 140a、140b、140c 可實施 MIMO 技術。因此，基站 140a 例如可使用多個天線來向 WTRU 102a 發送無線信號，並從 WTRU 102a 接收無線信號。基站 140a、140b、140c 還可提供移動管理功能，例如切換觸發、通道

建立、無線電資源管理、業務分類、服務品質 (QoS) 策略執行等。ASN 閘道 142 可作為業務聚集點，並且可負責呼叫、暫存用戶簡檔、到核心網路 106 的路由等。

WTRU 102a、102b、102c 和 RAN 104 之間的空氣介面 116 可被定義為實施 IEEE 802.16 規範的 R1 參考點。此外，WTRU 102a、102b、102c 的每一個可與核心網路 106 建立邏輯介面（未示出）。WTRU 102a、102b、102c 和核心網路 106 之間的邏輯介面可被定義為可用於認證、許可、IP 主機配置管理和/或移動管理的 R2 參考點。

基站 140a、140b、140c 的每一個之間的通信鏈路可被定義為包括用於促進 WTRU 移交和基站之間的資料傳輸的協定的 R8 參考點。基站 140a、140b、140c 和 ASN 閘道 215 之間的通信鏈路可被定義為 R6 參考點。R6 參考點可包括用於基於與 WTRU 102a、102b、102c 的每一個相關聯的移動事件而促進移動管理的協定。

如第 48E 圖所示，RAN 104 可與核心網路 106 相連接。RAN 104 和核心網路 106 之間的通信鏈路可被定義為 R3 參考點，R3 參考點包括用於促進例如資料傳輸和移動管理能力的協定。核心網路 106 可包括移動 IP 本地代理者 (MIP-HA) 144、認證、許可、計費 (AAA) 伺服器 146 和閘道 148。雖然上述元件的每一個被圖示為核心網路 106 的一部分，將理解的是這些元件的任意一個可由除了核心網路營運商以外的實體所有和/或營運。

MIP-HA 可負責 IP 地址管理，並且可使 WTRU 102a、102b、102c 能在不同的 ASN 和/或不同的核心網路之間漫遊。MIP-HA 144 可向 WTRU 102a、102b、102c 提供到封包交換網路（例如網際網路 110）的存取，以促進 WTRU 102a、102b、102c 和致能 IP 的裝置之間的通信。AAA 伺服器 146

可負責用戶認證和支援用戶服務。閘道 148 可促進與其他網路的交互影響。例如，閘道 148 可向 WTRU 102a、102b、102c 提供到例如 PSTN 108 的電路交換網路的存取，以促進 WTRU 102a、102b、102c 和傳統陸地線路通信裝置之間的通信。此外，閘道 148 可向 WTRU 102a、102b、102c 提供到網路 112 的存取，網路 112 可包括由其他服務供應商所有和/或營運的其他有線或無線網路。

雖然在第 48E 圖中未示出，將理解的是 RAN 104 可與其他 ASN 相連接，而且核心網路 106 可與其他核心網路相連接。RAN 104 和其他 ASN 之間的通信鏈路可被定義為 R4 參考點，R4 參考點可包括用於協調 RAN 104 和其他 ASN 之間的 WTRU 102a、102b、102c 的移動的協定。核心網路 106 和其他核心網路之間的通信鏈路可被定義為 R5 參考，其可包括用於促進本地核心網路和外地核心網路之間的交互影響的協定。

儘管以上以特定的組合描述了特徵和元素，但是本領域通常技術人員將理解，每個特徵或元素可以單獨地或與其它的特徵和元素任意組合地使用。此外，在本文描述的方法可在併入由電腦或處理器執行的電腦可讀取媒體中的電腦程式、軟體或韌體中執行。電腦可讀取媒體的示例包括電子信號（經由有線或無線連接傳送）和電腦可讀取儲存媒體。電腦可讀取儲存媒體的示例包括但不限制為唯讀記憶體（ROM）、隨機存取記憶體（RAM）、暫存器、快取記憶體、半導體記憶體設備、磁性媒體（例如內部硬碟和可移除磁片）、磁光媒體和例如 CD-ROM 光碟片和數位化多功能光碟（DVD）的光學媒體。與軟體相關聯的處理器可用來執行在 WTRU、UE、終端、基站、RNC 或任何主電腦中使用的無線電頻率收發信機。

結論

草案 ETSI TS 102 690 V0.12.3 (2011-06) 及其更早的版本（總稱為“用於 M2M 通信的草案 ETSI 標準”）包括具有給定定義的若干術語，這些定義在用於 M2M 通信的草案 ETSI 標準的上下文中定義了這樣的術語的含義。這些術語和由草案 ETSI TS 102 690 V0.12.3 (2011-6) 及其更早版本所指定、公開和/或引用的相關聯的定義藉由引用被合併於本文。

實施例

在一個實施例中，一種方法可包括執行用於管理 M2M 環境中 M2M 實體的一個或多個管理層。該方法還可包括使用複數管理層來管理 M2M 區域網路，其中該 M2M 區域網路可包括一個或多個 M2M 端裝置。M2M 端裝置可包括例如 M2M 閘道和/或 M2M 裝置。管理層可包括應用管理層、服務管理層、網路管理層和裝置管理層的任一個。管理層可提供 M2M 實體的配置管理、故障管理和性能管理的任一個。

在一個實施例中，一種方法可包括以用於遠端實體管理（“REM”）的服務能力（“SC”）和以具有用於根據複數管理層的其中之一來執行第二 M2M 實體的 REM 的從屬資源結構的資源結構來配置第一 M2M 實體。該方法可進一步包括經由操縱從屬資源結構的資源和屬性的任一個來執行第二 M2M 實體的 REM。

如在前面實施例中的方法，其中第一 M2M 實體可以是 M2M 伺服器，並且其中第二 M2M 實體可以是 M2M 應用、M2M 服務能力、M2M 區域網路、M2M 閘道或 M2M 裝置。

如前面至少一個實施例的方法，其中複數管理層可包括裝置管理層、網路管理層、服務管理層和應用管理層的至少兩個。

如前面至少一個實施例的方法，其中用於第二 M2M 實體的遠端實體管理的從屬資源結構可包括根據應用管理層而用於 M2M 應用的遠端實體管理的資源結構。

如前面實施例的方法，其中應用管理層可包括用於執行應用生命週期管理的一個或多個功能。

如前面至少一個實施例的方法，其中用於第二 M2M 實體的遠端實體管理的從屬資源結構可包括根據服務管理層而用於 M2M 服務能力的遠端實體管理的資源結構。

如前面實施例的方法，其中服務管理包括用於執行軟體管理和/或韌體管理的功能。

如前面至少一個實施例的方法，其中用於第二 M2M 實體的遠端實體管理的從屬資源結構可包括根據網路管理層而用於 M2M 區域網路的遠端實體管理的資源結構。

如前面至少一個實施例的方法，其中用於第二 M2M 實體的遠端實體管理的從屬資源結構可包括根據裝置管理層而用於 M2M 裝置的遠端實體管理的資源結構。

如前面至少一個實施例的方法，其中複數管理層的每一個可定義用於執行第二 M2M 實體的配置管理、故障管理和性能管理的一個或多個功能。

如前面至少一個實施例的方法可進一步包括在第一 M2M 實體處接收命令，以操縱從屬資源結構的資源和屬性的任一個，其中第二 M2M 實體的 REM 可響應於該命令而被執行。

如前面實施例的方法，其中該命令包括 RESTful 方法。

如前面至少一個實施例的方法可進一步包括在第一 M2M 實體處接收第一命令，以操縱從屬資源結構的資源和屬性的任一個。

如前面實施例的方法，其中執行第二 M2M 實體的 REM

可包括向第二 M2M 實體發送第二命令，以操縱從屬資源結構的資源和屬性的任一個，和響應於所接收的第二命令而執行第二 M2M 實體的 REM。

如前面實施例的方法，其中第一和第二命令兩者都可包括 RESTful 方法。

如前面至少一個實施例的方法，其中第一命令可包括 RESTful 方法，並且其中第二命令包括非 RESTful 方法。

如前面至少一個實施例的方法，其中執行第二 M2M 實體的 REM 可包括向第二 M2M 實體發送命令，以操縱從屬資源結構的資源和屬性的任一個，並響應於所接收的命令而執行第二 M2M 實體的 REM。

如前面實施例的方法，其中該命令可以是非 RESTful 方法或 RESTful 方法。

如前面至少一個實施例的方法，其中第二 M2M 實體可包括從屬資源結構的副本，並且其中執行第二 M2M 實體的 REM 包括操縱從屬資源結構的副本，以在操縱從屬資源結構的資源和屬性的任一者後複製該從屬資源結構。

在一個實施例中，一種設備可包括第一 M2M 實體，其以具有用於 REM 的 SC 和具有從屬資源結構的資源結構加以配置，從屬資源結構用於根據複數管理層其中之一而執行第二 M2M 實體的 REM。該設備可進一步包括適用於經由操縱從屬資源結構的資源和屬性的任一個來執行第二 M2M 實體的 REM 的處理器。

如前面實施例的設備，其中第一 M2M 實體可包括 M2M 伺服器，並且其中第二 M2M 實體可包括 M2M 應用、M2M 服務能力、M2M 區域網路、M2M 閘道或 M2M 裝置。

如前面至少一個實施例的設備，其中複數管理層可包括裝置管理層、網路管理層、服務管理層和應用管理層的兩個

或更多。

如前面至少一個實施例的設備，其中用於第二 M2M 實體的 REM 的從屬資源結構可包括根據應用管理層而用於 M2M 應用的 REM 的資源結構。

如前面至少一個實施例的設備，其中應用管理層可包括用於執行應用生命週期管理的功能。

如前面至少一個實施例的設備，其中用於第二 M2M 實體的遠端實體管理的從屬資源結構可包括根據服務管理層而用於 M2M 服務能力的遠端實體管理的資源結構。

如前面實施例的設備，其中服務管理可包括用於執行軟體管理和/或韌體管理的功能。

如前面至少一個實施例的設備，其中用於第二 M2M 實體的遠端實體管理的從屬資源結構可包括根據網路管理層而用於 M2M 區域網路的遠端實體管理的資源結構。

如前面至少一個實施例的設備，其中用於第二 M2M 實體的遠端實體管理的從屬資源結構可包括根據裝置管理層而用於 M2M 裝置的遠端實體管理的資源結構。

如前面至少一個實施例的設備，其中複數管理層的每一個可定義用於執行第二 M2M 實體的配置管理、故障管理和性能管理的任一個的功能。

如前面至少一個實施例的設備，其中第一 M2M 實體可包括：用於接收命令以操縱從屬資源結構的資源和屬性的任一個的 SC，其中第二 M2M 實體的 REM 可響應於該命令而被執行。

如前面至少一個實施例的設備，其中該命令可以是 RESTful 方法。

如前面至少一個實施例的設備，其中第一 M2M 實體可包括 (i) 用於接收第一命令以操縱從屬資源結構的資源和

屬性的任一個的 SC，和 (ii) 用於向第二 M2M 實體通信第二命令以操縱在第二 M2M 實體上所維護的從屬資源結構的副本的資源副本和屬性副本的任一個的 SC。

如前面實施例的設備，其中處理器可適用於向第二 M2M 實體發送第二命令，以引起響應於第二命令，在第二 M2M 實體處所維護的從屬資源結構的副本的資源副本和屬性副本的任一個的操縱。

如前面至少一個實施例的設備，其中第一和第二命令兩者都可以是 RESTful 方法，或者替代地，其中第一命令可以是 RESTful 方法，並且第二命令可以是非 RESTful 方法。

如前面至少一個實施例的設備，其中第一 M2M 實體可包括(i)用於向第二 M2M 實體通信命令以操縱在第二 M2M 實體處所維護的從屬資源結構的副本的資源副本和屬性副本的任一個的 SC。

如前面實施例的設備，其中處理器可適用於向第二 M2M 實體發送命令，以引起響應於第二命令，在第二 M2M 實體處所維護的從屬資源結構的副本的資源副本和屬性副本的任一個的操縱。

如前面實施例的設備，其中該命令可以是非 RESTful 方法，或者替代地，是 RESTful 方法。

在一個實施例中，一種系統可包括伺服器，伺服器具有以具有用於 REM 的服務能力和以具有用於根據複數管理層的其中之一來執行第二 M2M 實體的 REM 的從屬資源結構的資源結構來配置的第一 M2M 實體。該系統還可包括處理器，適用於經由操縱從屬資源結構的資源和屬性的任一者來執行第二 M2M 實體的 REM。該系統可進一步包括一裝置及處理器，該裝置具有以從屬資源結構的副本來配置的第二 M2M 實體，該處理器適用於在伺服器操縱之後操縱從屬資

源結構的副本，以複製從屬資源結構。

在一個實施例中，一種實體電腦可讀取儲存媒體可在其上儲存可執行的指令，這些指令用於 (i) 以用於 REM 的 SC 和以具有用於根據複數管理層的其中之一來執行第二 M2M 實體的 REM 的從屬資源結構的資源結構來配置第一 M2M 實體；和 (ii) 經由操縱從屬資源結構的資源和屬性的任一個來執行第二 M2M 實體的 REM。這些可執行的指令可載入至計算裝置的記憶體中，並且可由該計算裝置執行。

在一個實施例中，一種實體電腦可讀取儲存媒體可在其上儲存可執行的指令，這些指令用於 (i) 以用於 REM 的 SC 和以具有用於根據複數管理層的其中之一來執行第二 M2M 實體的 REM 的從屬資源結構的資源結構來配置第一 M2M 實體；(ii) 經由操縱從屬資源結構的資源和屬性的任一個來執行第二 M2M 實體的 REM；和 (iii) 操縱由第二 M2M 實體所維護的從屬資源結構的副本，以便在從屬資源結構的資源和屬性的任一個的操縱之後複製從屬資源結構。

在一個實施例中，一種方法可包括執行用於在 M2M 環境中執行管理功能的基於用戶端/伺服器的遠端實體管理 (xREM) 模型。該方法還可包括將該模型應用於 M2M 環境中的 M2M 裝置。

在一個實施例中，一種方法可包括使用基於通道的技術，以在使用多個管理協定的 M2M 環境中執行 xREM；和將該模型應用於 M2M 環境中的 M2M 裝置。

在一個實施例中，一種方法可包括提供包括屬性元件的資源命令，並將該資源命令通信給 M2M 端設備。

如前面實施例的方法，其中該資源命令可以是非 RESTful 命令。

在一個實施例中，一種方法可包括提供具有屬性元件和

可能的一些子參數的資源命令結構。該方法還可包括將該資源命令通信給 M2M 端裝置。傳統的非 RESTful 管理命令可使用 RESTful 方法而在 M2M 端裝置上簡單地操作。

在一個實施例中，一種方法可包括在 M2M 裝置或 M2M 閘道處儲存存取歷史資源結構，在 M2M 裝置或 M2M 閘道處檢測操作，和操縱存取歷史資源結構以儲存與該操作相關聯的至少一個細節。

在一個實施例中，一種方法可包括接收用於對機器對機器 (M2M) 裝置或閘道授予許可權的請求，並授予對 M2M 裝置或閘道的許可權。

在一個實施例中，一種方法可在 M2M 端裝置處執行。該方法可包括接收具有用於管理該 M2M 端裝置的一個或多個屬性元件的資源命令。

如前面實施例的方法還可包括基於該資源命令而應用一個或多個管理功能。

在一個實施例中，一種方法可包括提供包括一個或多個屬性元件的資源命令，並將該資源命令通信給 M2M 端裝置。

在一個實施例中，一種用於在第一系統中執行 xREM 的方法可包括例如適用於根據用於 M2M 通信的協定 (“M2M 通信協定”) (例如草案 ETSI TS 102 690) 而通信的第一、第二和第三裝置。M2M 通信協定可定義用於管理駐留在每一個邏輯層 (“管理層”) 的實體的邏輯層的棧或集合。這些管理層可包括例如 M2M 應用管理層、M2M 服務能力層、M2M 網路管理層和 M2M 裝置管理層。第一裝置可定義駐留在管理層的第一邏輯層 (“第一管理層”) 的實體。該實體可以是例如 M2M 應用、M2M 服務能力、M2M 區域網路、M2M 閘道或 M2M 裝置。第一裝置可包括第一資料結構，其定義用於根據第一管理層來管理實體的資源 (“第一管理

層資源”)。第二裝置可包括也定義第一管理層資源的第二資料結構。並且，第二裝置可通信地與第一和第三裝置相耦合。

該方法可包括從第二裝置向第三裝置提供標識該第一管理層資源的識別符。該方法還可包括在第二裝置處從第三裝置接收該識別符和用於第一管理層資源的應用的資訊，並將該資訊應用於第一管理層資源。

在一個或多個實例中，將資訊應用於第一管理層資源可包括操縱第二資料結構。替代地，將資訊應用於第一管理層資源可包括將該資訊從第二裝置發送給第一裝置，以引起第一裝置來操縱第一資料結構。

識別符可包括和/或是例如統一資源識別符、鏈結和位址的任一者。第一管理層資源可包括和/或定義管理物件。另外，第一、第二和第三裝置的每一個可包括用於根據 M2M 通信協定來通信的模組。第三裝置可進一步包括應用（例如 M2M 應用），其適用於提供用於第一管理層資源的應用的資訊，並且其中應用的執行涉及了根據 M2M 通信協定的資訊通信。第一裝置也可包括應用，並且這樣的應用的執行涉及了根據 M2M 通信協定的資訊通信。

在一個或多個實施例中，第一裝置可以是設備，第三裝置可以是無線發射/接收單元（“WTRU”）並且第二裝置可包括伺服器。

作為另一個示例，公開了一種用於 xREM 的設備。該設備包括適用於根據 M2M 通信協定來通信的第一裝置。如上所述，M2M 通信協定定義了管理層的棧或集合。第一裝置可包括第一資料結構。第一資料結構定義了第一管理層資源，其根據第一管理層而用於管理駐留在第一管理層的第二裝置的實體。第一裝置還可通信地與第二裝置和第三裝置相耦合。第一裝置可進一步包括適用於儲存可執行的指令的記

記憶體，這些指令適用於：向第三裝置提供標識第一管理層資源的識別符；從第三裝置接收該識別符和用於第一管理層資源的應用的資訊；和將該資訊應用於第一管理層資源。第一裝置還可包括適用於從記憶體獲取該可執行的指令並執行該可執行的指令的處理器。

識別符可包括和/或是例如統一資源識別符、鏈結和位址的任一者。第一管理層資源可包括管理物件。

第一裝置可包括用於根據 M2M 通信協定來通信的模組。用於第一管理層資源的應用的資訊可從第三裝置的應用接收。第一裝置可包括伺服器，第二裝置可以是裝置，並且第三裝置可以是 WTRU。

公開了用於在第二系統中執行 xREM 的方法的另一個示例。第二系統可包括適用於根據 M2M 通信協定來通信的第一和第二裝置。M2M 通信協定定義了管理層。第一裝置定義了駐留在第一管理層的實體，並包括定義了第一管理層資源的第一資料結構。第二裝置包括了也定義第一管理層資源的第二資料結構。第一裝置可通信地與第二裝置相耦合。

該方法可包括與第二裝置協商的第一裝置，以定義用於根據第一管理層來管理實體的管理協定的類型。與第二裝置協商可包括例如從第一裝置向第二裝置發送第一訊息，以請求在第二裝置處的服務能力層（“SCL”）的註冊，其中第一訊息包括定義管理協定類型的屬性和分配給該屬性的第一值。與第二裝置協商還可包括在第一裝置處從第二裝置接收響應於第一訊息所發送的第二訊息。第二訊息可包括分配給定義管理協定類型的屬性的第二值。

在一個或多個實施例中，與第二裝置協商可進一步包括在第一裝置處從第二裝置接收第三訊息，以請求更新 SCL 的註冊，其中第三訊息包括分配給定義管理協定類型的屬性

的第二值。

作為一個替代選擇，與第二裝置協商可包括從第一裝置向第二裝置發送第一訊息，以請求在第二裝置的 SCL 中物件的創建。為了促進此事，第一訊息可包括定義管理協定的類型的屬性和分配給該屬性的第一值。與第二裝置協商可進一步包括在第一裝置處從第二裝置接收響應於第一訊息所發送的第二訊息。第二訊息可包括分配給該屬性的第二值。

作為另一個替代選擇，與第二裝置協商可包括從第一裝置向第二裝置發送第一訊息，以請求在第二裝置對 SCL 中的物件的更新。第一訊息可包括識別符，用於標識定義管理協定類型的屬性和分配給該屬性的第一值。與第二裝置協商還可包括在第一裝置處從第二裝置接收響應於第一訊息而發送的第二訊息。第二訊息可包括分配給該屬性的第二值。

在仍然另一個選擇中，與第二裝置協商可包括在第一裝置處從第二裝置接收第一訊息，以請求在第二裝置對 SCL 中的物件的更新。為了促進該更新，第一訊息可包括識別符，標識定義管理協定類型的屬性的識別符和分配給該屬性的第一值。與第二裝置協商還可包括從第一裝置向第二裝置發送響應於第一訊息的第二訊息。第二訊息可包括分配給該屬性的第二值。

在另一個替代選擇中，與第二裝置協商可包括從第一裝置向第三裝置發送第一訊息，以發現第二裝置。第一訊息可包括定義管理協定類型的屬性和分配給該屬性的第一值。與第二裝置協商還可包括在第一裝置處從第三裝置接收響應於第一訊息而發送的第二訊息。此第二訊息可包括分配給定義協定類型的屬性的第二值。分配給該屬性的第二值可從第二裝置的 SCL 獲得。與第二裝置協商可進一步包括如果第一和第二值相等，選擇用於註冊第一裝置的 SCL 的第二裝

置。

管理協定的類型可以是簡易網路管理協定(“SNMP”)、寬頻論壇(“BBF”) TR-069 CPE WAN 管理協定和開放行動聯盟(OMA) 裝置管理(DM) 協定的任一個。

公開了用於在第二系統中執行 xREM 的方法的另一個示例。該方法可包括通知第二裝置用於根據第一邏輯層管理實體的管理協定的類型。

公開了用於執行 xREM 的方法的另一個示例。該方法可包括第一裝置，其從第二裝置接收用於向第一裝置授予用於第三裝置的 xREM 的許可權的請求；和響應於該請求，第二裝置可將許可權傳遞給第一裝置。在獲得許可權後，第一裝置可執行對(over) 第三裝置的許可權。

在一個或多個實施例中，一種方法可包括在第一實體處從第二實體接收用於執行 RESTful 方法的請求。第一實體可包括定義命令的資源(“命令資源”)的資料結構。該請求可包括用於標識命令資源的識別符和用於執行該命令的資訊。該方法還可包括執行命令，作為識別符的功能和用於執行該命令的資訊。該識別符可以是統一資源識別符、鏈結和位址的任一個。

在一個或多個實施例中，第一實體可分別包括第一和第二命令資源的第一和第二資料結構，並且該識別符可包括和/或是對第二資源的指標。

在一個或多個實施例中，一種方法可包括在第一實體處從第二實體接收用於執行 RESTful 方法的請求。第一實體可包括定義命令的命令資源的資料結構。該請求可包括用於標識命令資源的識別符和用於執行該命令的資訊。該方法還可包括響應於該請求而生成資源的第一實例，並更新該識別符以標識資源的第一實例，並且執行該命令作為識別符的功能

和用於執行該命令的資訊。

在一個或多個實施例中，一種方法可包括在第一實體處從第二實體接收用於執行 RESTful 方法的第一請求。第一實體可包括命令資源的第一資料結構，第一請求可包括用於標識命令資源的第一識別符和用於執行該命令的第一資訊。該方法還可包括響應於第一請求而生成命令資源的第一實例；更新第一識別符，以標識命令資源的第一實例；在第一實體處從第二實體接收用於執行 RESTful 方法的第二請求。第二請求可包括用於標識命令資源的第二識別符和用於執行該命令的第二資訊。該方法可進一步包括響應於第二請求而生成命令資源的第二實例；更新第二識別符，以標識命令資源的第二實例；執行該命令作為第一識別符的功能和用於執行命令的第一資訊；和執行該命令作為第二識別符的功能和用於執行命令的第二資訊。

在一個實施例中，公開了用於根據用於機器對機器通信的協定而執行 xREM 的方法。該方法可包括在第一實體處從第二實體接收用於執行 RESTful 方法的請求（“RESTful 方法請求”）。第一實體可包括可由 RESTful 方法修改的資料結構。此資料結構可以是例如代表服務能力層（“SCL”）的資料結構，例如包括在本文被稱為 sclbase 等的任何資料結構。RESTful 方法請求可標識與可由第一實體所執行的命令相關聯的資源（此後稱為“資源命令”）。該方法還可包括執行 RESTful 方法，以根據資源命令而調用對資料結構的修改。

在一個或多個實施例中，RESTful 方法可以是 RESTful 方法創建、RESTful 方法獲取、RESTful 方法更新和/或 RESTful 方法刪除。在 RESTful 方法是例如 RESTful 方法創建的實施例中，執行 RESTful 方法可包括在資料結構中實例化代表資源命令的從屬資料結構（此後稱為“命令資源結

構”）。

在 RESTful 方法是 RESTful 方法刪除的實施例中，執行 RESTful 方法可包括從資料結構刪除命令資源結構。在 RESTful 方法是 RESTful 方法獲取的實施例中，執行 RESTful 方法可包括向第二實體發送命令資源結構的一些或所有的副本和/或命令資源結構的狀態（“命令資源狀態”）。

在 RESTful 方法是 RESTful 方法更新的實施例中，執行 RESTful 方法可包括修改命令資源結構以調用命令的狀態（“命令狀態”）的改變。在一個或多個實施例中，修改從屬資料結構可包括修改命令資源結構，以調用命令的執行（“命令執行”）。例如，響應於檢測到對命令資源結構的這樣的修改，命令執行可由第一實體調用。

在一個或多個實施例中，資源命令可定義用於調用命令執行的從屬資源（“子資源”）（“命令執行子資源”）。此命令執行子資源可以是例如在下文被稱作 `execEnable` 等的子資源的一個或多個實施例。命令資源結構的一個或多個元素（“命令資源結構元素”）可代表命令執行子資源。

在一個或多個實施例中，RESTful 方法請求可包括用於修改命令執行子資源以調用命令執行的資訊。在這些實施例中，修改命令執行子資源可包括命令資源結構元素，代表帶有該資訊的命令執行子資源，以便調用命令執行。

用於修改命令執行子資源以調用命令執行的資訊可以是可被分配和解釋以調用命令執行的數字、整數、字元、代碼等。以示例的方式，用於修改命令執行子資源以調用命令執行的資訊可以是“0”，並且因此以“0”修改代表命令執行子資源的命令資源結構元素調用了命令執行。

在一個或多個實施例中，資源命令可定義用於調用命令執行的屬性。在這些實施例中，命令資源結構元素可代表屬

性，這樣的元素可由識別符（“屬性識別符”）標識。此命令執行屬性識別符可以是例如在下文被稱為 `execEnable` 等的屬性的一個或多個實施例。

RESTful 方法請求可包括屬性識別符。另外，代表該屬性的命令資源結構元素的選擇可調用命令執行。並且，修改命令資源結構以調用命令的執行可包括使用屬性識別符來選擇代表該屬性的命令資源結構元素，這反過來調用命令的執行。

在一個或多個實施例中，修改命令資源結構可包括修改命令資源結構，以調用對命令執行的暫停。響應於檢測到對命令資源結構的這樣的修改，對命令執行的暫停例如可由第一實體來調用。

在一個或多個實施例中，資源命令可定義用於調用對命令執行的暫停的子資源。此子資源可以是例如在下文被稱為 `execEnable` 等的子資源的一個或多個實施例。在一個或多個實施例中，RESTful 方法請求可包括用於修改命令執行子資源以調用對命令執行的暫停的資訊。在這些實施例中，修改命令資源結構以調用對命令執行的暫停可包括修改代表命令執行子資源的命令資源結構元素，來調用該暫停。用於修改命令執行子資源的資訊可以是可被分配並解釋以調用對命令執行的暫停的數字、整數、字元、代碼等。以示例的方式，用於修改命令執行子資源以調用暫停的資訊可以是“1”，並且因此以“1”修改代表命令執行子資源的命令資源結構元素調用了對命令執行的暫停。

在一個或多個實施例中，資源命令可定義用於調用對命令執行的暫停的屬性（“暫停執行屬性”）。一個或多個命令資源結構元素可代表暫停執行屬性，並這樣的元素可由相應的屬性識別符來標識。此暫停執行屬性可以是例如在下文被

稱為 `execPause` 等的屬性的一個或多個實施例。RESTful 方法請求可包括暫停執行屬性識別符。另外，代表暫停執行屬性的命令資源結構元素的選擇可調用對命令執行的暫停。並且修改命令資源結構以調用對命令執行的暫停可包括使用暫停執行屬性識別符，來選擇代表暫停執行屬性的命令資源結構元素，這反過來調用對命令執行的暫停。

在一個或多個實施例中，修改命令資源結構可包括修改命令資源結構，以引起經暫停執行的命令恢復執行（“恢復命令執行”）。響應於檢測到對命令資源結構的修改以引起恢復命令執行，恢復命令執行例如可由第一實體調用。

在一個或多個實施例中，資源命令可定義用於調用恢復命令執行的子資源。此子資源可以是例如在下文被稱為 `execEnable` 等的子資源的一個或多個實施例。在一個或多個實施例中，RESTful 方法請求可包括用於修改命令執行子資源以調用恢復命令執行的資訊。在這些實施例中，修改命令資源結構以調用恢復命令執行可包括修改代表命令執行子資源的命令資源結構元素，以調用恢復命令執行。此資訊可以是可被分配並解釋以調用恢復命令執行的數字、整數、字元、代碼等。以示例的方式，用於修改命令執行子資源以調用恢復命令執行的資訊可以是“2”，並且因此以“2”修改代表恢復命令執行子資源的命令資源結構元素調用了恢復命令執行。

在一個或多個實施例中，資源命令可定義用於調用恢復命令執行的屬性（“恢復執行屬性”）。一個或多個命令資源結構元素可代表恢復執行屬性，並這樣的元素可由相應的屬性識別符來標識。此恢復執行屬性可以是例如在下文被稱為 `execResume` 等的屬性的一個或多個實施例。RESTful 方法請求可包括恢復執行屬性識別符。另外，代表恢復執行屬性的

命令資源結構元素的選擇可調用恢復命令執行。修改命令資源結構以調用恢復命令執行可包括使用恢復執行屬性識別符來選擇代表恢復執行屬性的命令資源結構元素，這反過來調用恢復命令執行。

在一個或多個實施例中，修改命令資源結構可包括修改命令資源結構，以調用命令執行的取消（“取消命令執行”）。響應於檢測到對命令資源結構的修改以調用取消命令執行，取消命令執行例如可由第一實體調用。

在一個或多個實施例中，資源命令可定義用於調用取消命令執行的子資源。此子資源可以是例如在下文被稱為 `execEnable` 等的子資源的一個或多個實施例。在一個或多個實施例中，RESTful 方法請求可包括用於修改命令執行子資源以調用取消命令執行的資訊。在這些實施例中，修改命令資源結構以調用取消命令執行可包括修改代表命令執行子資源的命令資源結構元素，以調用取消命令執行。此資訊可以是可被分配並解釋以調用取消命令執行的數字、整數、字元、代碼等。以示例的方式，用於修改命令執行子資源以調用取消命令執行的資訊可以是“3”，並且因此以“3”修改代表取消命令執行子資源的命令資源結構元素調用了取消命令執行。

在一個或多個實施例中，資源命令可定義用於調用取消命令執行的屬性（“取消執行屬性”）。一個或多個命令資源結構元素可代表取消執行屬性，並且這樣的元素可由相應的屬性識別符來標識。此取消執行屬性可以是例如在下文被稱為 `execDisable` 等的屬性的一個或多個實施例。RESTful 方法請求可包括取消執行屬性識別符。另外，代表取消執行屬性的命令資源結構元素的選擇可調用取消命令執行。修改命令資源結構以調用取消命令執行可包括使用取消執行屬性識

別符來選擇代表取消執行屬性的命令資源結構元素，這反過來調用取消命令執行。

在 RESTful 方法是 RESTful 方法刪除的一個或多個實施例中，執行 RESTful 方法可包括修改命令資源結構，以調用命令執行狀態的變化，並從資料結構刪除該命令資源結構。

在一個或多個實施例中，修改命令資源結構可包括修改命令資源結構，以調用取消命令執行。響應於檢測到對命令資源結構的修改以調用取消命令執行，取消命令執行還可由第一實體調用。替代地，響應於 RESTful 方法刪除，取消命令執行例如可由第一實體調用。

在一個或多個實施例中，修改命令資源結構以調用取消命令執行可包括修改代表命令執行子資源的命令資源結構元素，以調用取消命令執行，如上所述。在一個或多個實施例中，修改命令資源結構以調用取消命令執行可包括使用取消執行屬性識別符來選擇代表取消執行屬性的命令資源結構元素，這反過來調用取消命令執行。

公開了用於根據用於機器對機器通信的協定而執行遠端實體管理的替代方法。此方法可包括在第一實體處從第二實體接收 RESTful 方法請求。第一實體可包括可由 RESTful 方法修改的資料結構。該資料結構可包括代表第一資源的從屬資料結構，其中該第一資源定義了用於調用命令資源狀態的操作。該從屬資料結構可由識別符標識，並且其中 RESTful 方法請求可包括該識別符，並且可標識該資源命令。該方法還可包括執行該 RESTful 方法，以根據該識別符和資源命令調用對資料結構的修改。

在一個或多個實施例中，用於調用命令資源狀態的改變的操作可包括調用命令執行、調用對命令執行的暫停、調用恢復命令執行或調用取消命令執行的操作。

在一個實施例中，用於根據用於機器對機器通信的協定而執行 xREM 的方法可包括在第一實體處從第二實體接收用於執行 RESTful 方法的請求（“RESTful 方法請求”）。第一實體可包括可由 RESTful 方法修改的資料結構。此資料結構可以是例如代表服務能力層（“SCL”）的資料結構，包括例如在本文被稱為 sclbase 等的任何資料結構。RESTful 方法請求可標識與可由第三實體所執行的命令相關的資源（此後稱為“資源命令”）。該方法還可包括執行該 RESTful 方法，以根據資源命令調用對資料結構的修改。

在一個或多個實施例中，RESTful 方法可以是 RESTful 方法創建、RESTful 方法獲取、RESTful 方法更新和/或 RESTful 方法刪除。在 RESTful 方法是例如 RESTful 方法創建的實施例中，執行 RESTful 方法可包括在資料結構中實例化代表資源命令的從屬資料結構（此後稱為“命令資源結構”）。

在 RESTful 方法是 RESTful 方法刪除的實施例中，執行 RESTful 方法可包括從資料結構刪除命令資源結構。在 RESTful 方法是 RESTful 方法獲取的實施例中，執行 RESTful 方法可包括向第二實體發送命令資源結構的一些或全部的副本和/或命令資源結構的狀態（“命令資源狀態”）。

在 RESTful 方法是 RESTful 方法更新的實施例中，執行 RESTful 方法可包括修改命令資源結構以調度命令的狀態（“命令狀態”）的變化。在一個或多個實施例中，修改從屬資料結構可包括修改命令資源結構，以調用命令的執行（“命令執行”）。響應於檢測到對命令資源結構的這樣的修改，命令執行可例如由第一實體調用。

在一個或多個實施例中，資源命令可定義用於調用命令執行的從屬資源（“子資源”）（“命令執行子資源”）。此命令

執行子資源可以是例如下文被稱為 `execEnable` 等的子資源的一個或多個實施例。命令資源結構的一個或多個元素(“命令資源結構元素”)可代表命令執行子資源。

在一個或多個實施例中，RESTful 方法請求可包括用於修改命令執行子資源的資訊，以調用命令執行。在這些實施例中，修改命令執行子資源可包括以該資訊代表命令執行子資源的命令資源結構元素，以便調用命令執行。

用於修改命令執行子資源以調用命令執行的資訊可以是可被分配和解釋以調用命令執行的數字、整數、字元、代碼等。以示例的方式，用於修改命令執行子資源以調用命令執行的資訊可以是“0”，並且因此以“0”修改代表命令執行子資源的命令資源結構元素調用了命令執行。

在一個或多個實施例中，資源命令可定義用於調用命令執行的屬性。在這些實施例中，命令資源結構元素可代表屬性，這樣的元素可由識別符(“屬性識別符”)標識。此命令執行屬性識別符可以是例如在下文被稱為 `execEnable` 等的屬性的一個或多個實施例。

RESTful 方法請求可包括屬性識別符。另外，代表該屬性的命令資源結構元素的選擇可調用命令執行。並且，修改命令資源結構以調用命令的執行可包括使用屬性識別符來選擇代表該屬性的命令資源結構元素，這反過來調用命令執行。

在一個或多個實施例中，修改命令資源結構可包括修改命令資源結構，以調用對命令執行的暫停。響應於檢測到對命令資源結構的這樣的修改，對命令執行的暫停例如可由第一實體調用。

在一個或多個實施例中，資源命令可定義用於調用對命令執行的暫停的子資源。此子資源可以是例如在下文稱為

`execEnable` 等的子資源的一個或多個實施例。在一個或多個實施例中，RESTful 方法請求可包括用於修改命令執行子資源以調用對命令執行的暫停的資訊。在這些實施例中，修改命令資源結構以調用對命令執行的暫停可包括修改代表命令執行子資源的命令資源結構元素來調用該暫停。用於修改命令執行子資源的資訊可以是可被分配並解釋以調用對命令執行的暫停的數字、整數、字元、代碼等。以示例的方式，用於修改命令執行子資源以調用暫停的資訊可以是“1”，並且因此以“1”修改代表命令執行子資源的命令資源結構元素調用了對命令執行的暫停。

在一個或多個實施例中，資源命令可定義用於調用對命令執行的暫停的屬性（“暫停執行屬性”）。一個或多個命令資源結構元素可代表暫停執行屬性，並且這樣的元素可由相應的屬性識別符來標識。此暫停執行屬性可以是例如在下文被稱為 `execPause` 等的屬性的一個或多個實施例。RESTful 方法請求可包括暫停執行屬性識別符。另外，代表暫停執行屬性的命令資源結構元素的選擇可調用命令執行的暫停。並且修改命令資源結構以調用命令執行的暫停可包括使用暫停執行屬性識別符來選擇代表暫停執行屬性的命令資源結構元素，這反過來調用對命令執行的暫停。

在一個或多個實施例中，修改命令資源結構可包括修改命令資源結構以引起經暫停執行的命令恢復執行（“恢復命令執行”）。響應於檢測到對命令資源結構的修改以引起恢復命令執行，恢復命令執行例如可由第一實體調用。

在一個或多個實施例中，資源命令可定義用於調用恢復命令執行的子資源。此子資源可以是例如在下文被稱為 `execEnable` 等的子資源的一個或多個實施例。在一個或多個實施例中，RESTful 方法請求可包括用於修改命令執行子資

源以調用恢復命令執行的資訊。在這些實施例中，修改命令資源結構以調用恢復命令執行可包括修改代表命令執行子資源的命令資源結構元素，以調用恢復命令執行。此資訊可以是可被分配並解釋以調用恢復命令執行的數字、整數、字元、代碼等。以示例的方式，用於修改命令執行子資源以調用恢復命令執行的資訊可以是“2”，並且因此以“2”修改代表恢復命令執行子資源的命令資源結構元素調用了恢復命令執行。

在一個或多個實施例中，資源命令可定義用於調用恢復命令執行的屬性（“恢復執行屬性”）。一個或多個命令資源結構元素可代表恢復執行屬性，並且這樣的元素可由相應的屬性識別符來標識。此恢復執行屬性可以是例如在下文被稱為 `execResume` 等的屬性的一個或多個實施例。RESTful 方法請求可包括恢復執行屬性識別符。另外，代表恢復執行屬性的命令資源結構元素的選擇可調用恢復命令執行。修改命令資源結構以調用恢復命令執行可包括使用恢復執行屬性識別符來選擇代表恢復執行屬性的命令資源結構元素，這反過來調用恢復命令執行。

在一個或多個實施例中，修改命令資源結構可包括修改命令資源結構，以調用命令執行的取消（“取消命令執行”）。響應於檢測到對命令資源結構的修改以調用取消命令執行，取消命令執行例如可由第一實體調用。

在一個或多個實施例中，資源命令可定義用於調用取消命令執行的子資源。此子資源可以是例如在下文被稱為 `execEnable` 等的子資源的一個或多個實施例。在一個或多個實施例中，RESTful 方法請求可包括用於修改命令執行子資源以調用取消命令執行的資訊。在這些實施例中，修改命令資源結構以調用取消命令執行可包括修改代表命令執行子

資源的命令資源結構元素，以調用取消命令執行。此資訊可以是可被分配並解釋以調用取消命令執行的數字、整數、字元、代碼等。以示例的方式，用於修改命令執行子資源以調用取消命令執行的資訊可以是“3”，並且因此以“3”修改代表取消命令執行子資源的命令資源結構元素調用了取消命令執行。

在一個或多個實施例中，資源命令可定義用於調用取消命令執行的屬性（“取消執行屬性”）。一個或多個命令資源結構元素可代表取消執行屬性，並且這樣的元素可由相應的屬性識別符來標識。此取消執行屬性可以是例如在下文被稱為 `execDisable` 等的屬性的一個或多個實施例。RESTful 方法請求可包括取消執行屬性識別符。另外，代表取消執行屬性的命令資源結構元素的選擇可調用取消命令執行。修改命令資源結構以調用取消命令執行可包括使用取消執行屬性識別符來選擇代表取消執行屬性的命令資源結構元素，這反過來調用取消命令執行。

在 RESTful 方法是 RESTful 方法刪除的一個或多個實施例中，執行 RESTful 方法可包括修改命令資源結構以調用命令執行狀態的變化，並從資料結構刪除該命令資源結構。

在一個或多個實施例中，修改命令資源結構可包括修改命令資源結構以調用取消命令執行。響應於檢測到對命令資源結構的修改以調用取消命令執行，取消命令執行還可由第一實體調用。替代地，響應於 RESTful 方法刪除，取消命令執行例如可由第一實體調用。

如上所述，在一個或多個實施例中，修改命令資源結構以調用取消命令執行可包括修改代表命令執行子資源的命令資源結構元素來調用取消命令執行。在一個或多個實施例中，修改命令資源結構以調用取消命令執行可包括使用取消

執行屬性識別符來選擇代表取消執行屬性的命令資源結構元素，這反過來調用取消命令執行。

公開了用於根據用於機器對機器通信的協定而執行遠端實體管理的一種替代方法。此方法可包括在第一實體處從第二實體接收 RESTful 方法請求。第一實體可包括可由 RESTful 方法修改的資料結構。該資料結構可包括代表第一資源的從屬資料結構，其中第一資源定義用於調用命令資源狀態的操作。從屬資料結構可由識別符標識，並且其中 RESTful 方法請求可包括該識別符並且可標識資源命令。該方法還可包括執行 RESTful 方法，以根據該識別符和資源命令來調用對資料結構的修改。

在一個或多個實施例中，用於調用命令資源狀態的改變的操作可包括調用命令執行、調用對命令執行的暫停、調用恢復命令執行或調用取消命令執行的操作。

在一個或多個實施例中，用於調用命令（例如 <command>、<commandInstance>或<requestInstance>，如下所述）的狀態的變化的任意從屬資料結構可包括從資料結構輸入到被從屬的這樣的從屬資料結構的屬性、子資源、參數和引數的任一個。

在一個實施例中，一種方法可包括維護裝置的位址映射，並使用該位址映射來向裝置發送通知。該裝置可以是裝置管理裝置。

在一個實施例中，一種方法可包括在裝置管理閘道處管理具有服務能力（D）的 M2M 裝置，使用用於管理 M2M 裝置的透明模式和代理模式的其中之一。

在一個實施例中，一種方法可包括在裝置管理閘道處管理 M2M 裝置，並使用用於管理 M2M 裝置的適應模式。

在一個實施例中，一種方法可包括在裝置管理閘道處管

理非 ETSI M2M 裝置，並使用用於管理非 ETSI M2M 裝置的適應模式。

在一個實施例中，一種方法可包括在閘道處維護裝置的位址映射，並使用該位址映射來向裝置發送通知。

如前面實施例的方法，其中裝置時裝置管理裝置。

如前面至少一個實施例的方法，其中裝置可被配置具有服務能力 (D)。

在一個實施例中，一種方法可包括在裝置管理閘道處管理具有服務能力 (D) 的 M2M 裝置，並使用用於管理 M2M 裝置的透明模式和代理模式的其中之一。

在一個實施例中，一種方法可包括在裝置管理閘道處管理 M2M 裝置，並使用用於管理 M2M 裝置的適應模式。

在一個實施例中，一種方法可包括在裝置管理閘道處管理非 ETSI M2M 裝置，並使用用於管理非 ETSI M2M 裝置的適應模式。

在一個實施例中，一種用於資料模型化 M2M 區域網路和 M2M 裝置的資料結構包括包含 etsiAreaNwkInfo 的至少一個管理物件，至少一個管理物件可包括 etsiAreaNwkDeviceInventory，至少一個管理物件包括 etsiAreaNwkDeviceGroups，至少一個管理物件包括 etsiAreaNwkGroupOperations，並且至少一個管理物件包括 etsiSensors。該資料結構可提供裝置目錄和配置管理、區域網路配置管理、區域網路性能管理或裝置的組管理的至少一個。

在一個實施例中，一種用於機器對機器 (M2M) 區域網路和 M2M 裝置的資料模型化的方法可包括管理 M2M 的資料模型，M2M 包括用於 M2M 網路和至少一個裝置的至少一個管理物件。

如前面實施例的方法，其中該管理可提供裝置目錄和配置管理。

如前面至少一個實施例的方法，其中該管理可提供區域網路配置管理。

如前面至少一個實施例的方法，其中該管理可提供區域網路性能管理。

如前面至少一個實施例的方法，其中該管理可提供裝置的組管理。

如前面至少一個實施例的方法，其中至少一個管理物件可包括 etsiAreaNwkInfo。

如前面至少一個實施例的方法，其中至少一個管理物件可包括 etsiAreaNwkDeviceInventory。

如前面至少一個實施例的方法，其中至少一個管理物件可包括 etsiAreaNwkDeviceGroups。

如前面至少一個實施例的方法，其中至少一個管理物件可包括 etsiAreaNwkGroupOperations。

如前面至少一個實施例的方法，其中至少一個管理物件可包括 etsiSensors。

如前面至少一個實施例的方法，其中至少一個管理物件可包括子資源。

如前面至少一個實施例的方法，其中至少一個管理物件可包括另一個管理物件的子資源。

如前面至少一個實施例的方法，其中 etsiAreaNwkInfo 可包括作為子資源的 areaNwkInstance。

如前面至少一個實施例的方法，其中 areaNwkInstance 可包括 areaNwkID、areaNwkType、workingChannelFrequency 和 addressMode 的任一者，作為子資源。

如前面至少一個實施例的方法，其中

etsiAreaNwkDeviceInventory 可包括 deviceInstance 和 deviceApplicationList 的任一者，作為組（groups）。

如前面至少一個實施例的方法，其中 deviceInstance 可包括 deviceGroupList、etsiBattery、etsiMemory 和 etsiSensor 的至少一個，作為子資源。

如前面至少一個實施例的方法，其中 deviceInstance 可包括 deviceType、deviceId、addressType、areaNwkID、internal address、external address、sleepInterval、sleepDuration、status、maxRtrAdvertisements、minDelayBetweenRas、maxRaDelayTime、tentativeNceLifetime、hopLimit、rtrSolicitationInterval、maxRtrSolicitations 或 maxRtrSolicitationInterval 的至少一個，作為子資源。

如前面至少一個實施例的方法，其中 etsiAreaNwkDeviceGroups 可包括作為子資源的 deviceGroupInstance。

如前面至少一個實施例的方法，其中 deviceGroupInstance 可包括 groupId、groupType、groupSize、members 或 condition 的至少一個，作為子資源。

如前面至少一個實施例的方法，其中 etsiAreaNwkGroupOperations 包括作為子資源的 operationInstance。

如前面至少一個實施例的方法，其中 operationInstance 可包括 groupId、enable、disable、results 或 description 的至少一個，作為子資源。

如前面至少一個實施例的方法，其中 etsiSensors 可包括作為子資源的 sensorInstance。

如前面至少一個實施例的方法，其中 sensorInstance 可包括 sensorID、sensorType、manufacturer 或 operations 的至

少一個，作為子資源。

如前面至少一個實施例的方法，其中 operations 可包括 enable、disable 或 result 的至少一個，作為子資源。

在一個實施例中，用於資料模型化機器對機器（M2M）區域網路和 M2M 裝置的資源結構可包括包含 etsiAreaNwkInfo 的至少一個管理物件；至少一個管理物件包括 etsiAreaNwkDeviceInventory；至少一個管理物件包括 etsiAreaNwkDeviceGroups；至少一個管理物件包括 etsiAreaNwkGroupOperations；和至少一個管理物件包括 etsiSensors，該資源結構提供裝置目錄和配置管理、區域網路配置管理、區域網路性能管理或裝置的組管理的至少一個。

在一個實施例中，用於資料模型化 M2M 區域網路和 M2M 裝置的資料結構包括包含 etsiAreaNwkInfo 的至少一個管理物件、包含 etsiAreaNwkDeviceInventory 的至少一個管理物件、包含 etsiAreaNwkDeviceGroups 的至少一個管理物件、包含 etsiGroupMgmtOperations 的至少一個管理物件和包含 etsiSensors 的至少一個管理物件。該資料結構提供裝置目錄和配置管理、區域網路配置管理、區域網路性能管理或裝置的組管理的至少一個。

在一個實施例中，一種用於機器對機器（M2M）區域網路和 M2M 裝置的資料模型化的方法可包括管理 M2M 的資料模型，M2M 包括用於 M2M 網路和至少一個裝置的至少一個管理物件。

如前面實施例的方法，其中該管理提供裝置目錄和配置管理。

如前面至少一個實施例的方法，其中該管理提供區域網路配置管理。

如前面至少一個實施例的方法，其中該管理提供區域網路性能管理。

如前面至少一個實施例的方法，其中該管理提供裝置的組管理。

如前面至少一個實施例的方法，其中至少一個管理物件包括 etsiAreaNwkInfo。

如前面至少一個實施例的方法，其中至少一個管理物件包括 etsiAreaNwkDeviceInventory。

如前面至少一個實施例的方法，其中至少一個管理物件包括 etsiAreaNwkDeviceGroups。

如前面至少一個實施例的方法，其中至少一個管理物件包括 etsiGroupMgmtOperations。

如前面至少一個實施例的方法，其中至少一個管理物件包括 etsiSensors。

如前面至少一個實施例的方法，其中至少一個管理物件包括子資源。

如前面至少一個實施例的方法，其中至少一個管理物件包括另一個管理物件的子資源。

如前面至少一個實施例的方法，其中 etsiAreaNwkInfo 包括作為子資源的 areaNwkInstance。

如前面至少一個實施例的方法，其中 areaNwkInstance 包括 areaNwkID、areaNwkType、workingChannelFrequency、addressMode、sleepInterval、sleepDuration、numOfDevices 和 attachedDevices 的至少一個，作為子資源。

如前面至少一個實施例的方法，其中 etsiAreaNwkDeviceInventory 包括 deviceInstance 和 areaNwkInstance 的至少一個，作為組 (groups)。

如前面至少一個實施例的方法，其中 deviceInstance 包

括 groups、deviceType、deviceID、addressType、areaNwkID、internalAddress、externalAddress、sleepInterval、sleepDuration、status、etsiBattery、etsiMemory、etsiSensor、blockSize 和 MTU 的至少一個，作為子資源。

如前面至少一個實施例的方法，其中 deviceInstance 包括 6LoWPAN、Wi-Fi 和 ZigBee 的至少一個，作為子資源。

如前面至少一個實施例的方法，其中 etsiAreaNwkDeviceGroup 包括作為子資源的 deviceGroupInstance。

如前面至少一個實施例的方法，其中 deviceGroupInstance 包括 groupID、groupType、groupSize、members 或 condition 的至少一個，作為子資源。

如前面至少一個實施例的方法，其中 etsiGroupMgmtOperations 包括 groups、subscriptions 和 operationInstance 的至少一個，作為子資源。

如前面至少一個實施例的方法，其中 operationInstance 包括 groupID、execEnable、execDisable、execPause、execResume、execStatus、OperationID、execResults 和 execParameters 的至少一個，作為子資源。

如前面至少一個實施例的方法，其中 etsiSensors 包括作為子資源的 sensorInstance。

如前面至少一個實施例的至少一個的方法，其中 sensorInstance 包括 sensorID、sensorType、manufacturer 或 operations 的至少一個，作為子資源。

如前面至少一個實施例的方法，其中 operations 包括 enable、disable 或 result 的至少一個，作為子資源。

在一個實施例中，無線發射/接收單元可被配置以執行如前面任一個實施例的方法。

在一個實施例中，基站可被配置以執行如前面任一個實施例的方法。

在一個實施例中，一種實體電腦可讀取儲存媒體可在其上儲存，可載入到計算裝置的記憶體中並可由其執行，用於執行如前面任一個實施例的方法的可執行的指令。

在一個實施例中，用於資料模型化機器對機器 (M2M) 區域網路和 M2M 裝置的資源結構包括包含 etsiAreaNwkInfo 的至少一個管理物件、包含 etsiAreaNwkDeviceInventory 的至少一個管理物件、包含 etsiAreaNwkDeviceGroups 的至少一個管理物件、包含 etsiGroupMgmtOperations 的至少一個管理物件和包含 etsiSensors 的至少一個管理物件，該資源結構提供裝置目錄和配置管理、區域網路配置管理、區域網路性能管理或裝置的組管理的至少一個。

上述方法、設備和系統的變型是可能的，而不脫離本發明的範圍。以可應用的各種實施例的觀點，應理解的是闡述的實施例僅是示例性的，不應被理解為對接著的申請專利範圍的限制。例如，在本文描述的示例實施例中，包括手持設備，其可包括或使用任何適當的、提供任意適當電壓的電源，例如電池等。

儘管以上以特定的組合描述了特徵和元素，但是一個本領域普通技術人員將理解，每個特徵或元素可以單獨地使用或與其它的特徵和元素任意組合地使用。此外，本文描述的方法可在包括在由電腦或處理器所執行的併入於電腦可讀取媒體中的電腦程式、軟體或韌體中執行。電腦可讀取媒體的示例包括電子信號（經由有線或無線連接來傳送）和電腦可讀取儲存媒體。電腦可讀取儲存媒體的示例包括但不限制為唯讀記憶體 (ROM)、隨機存取記憶體 (RAM)、寄存器、快取記憶體、半導體記憶體設備、磁性媒體（例如內部硬碟

和可移除磁片)、磁光媒體和例如 CD-ROM 光碟片和數位化多功能光碟 (DVD) 的光學媒體。與軟體相關聯的處理器可用來執行在 WTRU、UE、終端、基站、RNC 或任何主電腦中使用的無線電頻率收發信機。

此外，在上述的實施例中，應注意處理平台、計算系統、控制器和其他包括處理器的裝置。這些裝置可包括至少一個中央處理單元 (“CPU”) 和記憶體。根據電腦編程領域的技術人員的實務，涉及動作和操作或指令的符號表示可由各種 CPU 和記憶體來執行。這樣的動作和操作或指令可被稱為“執行”、“電腦執行”或“CPU 執行”。

一個本領域的技術人員將理解動作和符號表示的操作或指令包括由 CPU 操作的電子信號。電子系統表示能引起電子信號的結果變換或減少和在記憶體系統中的儲存位置維持資料位元，從而重新配置或改變 CPU 操作的資料位元，和信號的其他處理。資料位元要被維持的儲存位置是具有相應於或代表該資料位元的特殊電性、磁性、光學或有機性質的物理位置。應當理解的是，示例性實施例並不限於上述平台或 CPU 以及可支援上述方法的其他平台和 CPU。

資料位元還可以在電腦可讀取媒體上維持，電腦可讀取媒體包括 CPU 可讀取的磁片、光碟和任何其他揮發性（例如隨機存取記憶體 (“RAM”)）或非揮發性（例如唯讀記憶體 (“ROM”)）大量儲存系統。電腦可讀取媒體可包括協作或互連的電腦可讀取媒體，其專屬地存在於處理系統上，或分佈在多個可以是本地或遠端於處理系統的互連處理系統之間。應當理解的是，示例性實施例並不限於上述記憶體和其他可支援上述方法的平台和記憶體。

在本申請案說明中使用的元件、動作或指令都不應當被解釋為對本發明至關重要或必要的，除非明確地這麼描述。

同樣地，如本文使用的那樣，冠詞“一”和“一個”旨在包括一個或多個目。在旨在僅一個項目的情況下，則使用術語“單一”或類似語言。另外，跟隨在多個項目和/或多個項目類別列表後的術語“任一個”，如本文使用的那樣，旨在包括項目和/或項目類別、獨立於或結合於其他項目和/或其他項目類別的“任一個”、“任意組合”、“任意多個”和/或“多個的任意組合”。另外，如本文使用的那樣，術語“集合”旨在包括項目的任意數目，包括 0。此外，如本文使用的那樣，術語“數目”旨在包括任意數目，包括 0。

並且，申請專利範圍不應當被理解為受限於所描述的順序或元件，除非說明該效果。此外，在任何的申請專利範圍中使用術語“裝置 (means)”旨在援引 35 U.S.C. §112, ¶ 6，沒有措詞“裝置”的任何申請專利範圍則沒有這樣的含義。

【圖式簡單說明】

更詳細的理解可以從下述結合附圖以示例的方式給出的詳細說明中得到。在這樣的附圖中的圖式如詳細說明一樣是示例。這樣，圖式和詳細說明不應當被理解為限制，而且其他等同有效的示例是可能和適當的。並且，在圖式中相同的編號表明相同的元件，並且其中：

第 1A 圖-第 1C 圖是示出在其中一個或多個實施例，包括那些針對機器對機器 (“M2M”) 通信和/或操作的實施例，可得實現或執行的系統的示例的框圖；

第 2 圖是示出定義用於執行遠端實體管理 (“REM”) 的功能架構的示例邏輯管理層集合的框圖；

第 3A 圖是示出根據管理層集合及其功能而用於向 SCL 提供資源結構的示例資源結構框架的框圖；

第 3B 圖是示出根據管理層集合及其功能而用於向 SCL 提供資源結構的示例資源結構框架的框圖；

第 3C 圖是示出根據管理層集合及其功能而用於向 SCL 提供資源結構的示例資源結構框架的框圖；

第 4A 圖是示出用於向 SCL 提供管理物件 (“mgmtObjs”) 資源的示例資源結構框架的框圖；

第 4B 圖是示出用於向 SCL 提供管理物件 (“mgmtObjs”) 資源的示例資源結構框架的框圖；

第 5 圖是示出用於執行 xREM 的用戶端-伺服器模型的圖的框圖；

第 6 圖是示出使用多個不同管理協定以支援 xREM 的以通道為基礎的方法的框圖；

第 7A 圖-第 7C 圖是分別示出用於確定用於 REM 的管理協定類型的示例流程 700、730 和 760 的流程圖；

第 8 圖是示出用於協商和/或通知裝置用於 xREM 的管

理協定類型的過程的訊息流程圖；

第 9 圖是示出用於協商和/或通知裝置用於 xREM 的管理協定類型的過程的訊息流程圖；

第 10 圖是示出用於協商和/或通知裝置用於 xREM 的管理協定類型的過程的訊息流程圖；

第 11 圖是示出用於協商和/或通知裝置用於 xREM 的管理協定類型的過程的訊息流程圖；

第 12 圖是資源存取歷史的示例結構；

第 13 圖是基於“方法”所構建的資源存取歷史的示例結構；

第 14 圖是基於“方法”和“請求或 ID”所構建的資源存取歷史的示例結構；

第 15 圖是用於管理許可權（authority）授權（授權者發起的）的流的訊息流程圖；

第 16 圖是用於管理許可權授權（裝置發起的）的示例訊息流程圖；

第 17 圖是管理許可權授權（機器發起直接到被授權者）的示例訊息流程圖；

第 18 圖是管理許可權授權（被授權者發起的）的示例訊息流程圖；

第 19 圖是管理許可權授權（閘道作為代理）的示例訊息流程圖；

第 20 圖是用於管理許可權授權（機器到閘道授權）的示例訊息流程圖；

第 21 圖示出了用於管理許可權授權的示例過程的圖；

第 22A 圖-第 22C 圖是示出資源命令的示例結構的框圖；

第 23 圖是示出資源命令的示例結構的框圖；

第 24A 圖-第 24B 圖是示出資源命令的示例結構的框圖；

第 25A 圖-第 25B 圖是示出資源命令的示例結構的框圖；

第 26A 圖-第 26B 圖是示出資源命令的示例結構的框圖；

第 27A 圖-第 27B 圖是示出資源命令的示例結構的框圖；

第 28A 圖-第 28B 圖是示出資源命令示例結構的框圖；

第 28C 圖-第 28D 圖是示出資源命令實例的示例結構的框圖；

第 29A 圖-第 29R 圖是示出用於資源命令 xREM 的示例訊息流的訊息流程圖；

第 30A 圖是示出透過藉由槓桿操縱 (leveraging) OMA GwMO“透明”模型的 M2M GW(G')而用於管理 D 類型 ETSI M2M 裝置的示例架構的框圖；

第 30B 圖是示出 ETSI M2M xREM 的示例架構的框圖；

第 31A 圖是示出用於槓桿操縱 OMA GwMO1.0 的示例架構的框圖；

第 31B 圖是示出 xREM 的示例架構的框圖；

第 32A 圖是示出用於槓桿操縱 OMA GwMO1.0 的示例架構的框圖；

第 32B 圖是示出 ETSI M2M xREM 的示例架構的框圖；

第 33A 圖是示出用於槓桿操縱 OMA GwMO1.0 的示例架構的框圖；

第 33B 圖是根據本發明的具體實施例示出 ETSI M2M xREM 的示例架構的框圖；

第 34 圖是示出槓桿操縱 OMA GwMO 的以 GW 為基礎

的裝置管理的示例的框圖；

第 35 圖是示出部分緊密整合的 OMA DM 和 M2M GW 的示例架構的框圖；

第 36 圖是示出鬆散整合的 OMA DM 和 M2M GW 的示例架構的框圖；

第 37 圖是示出用於 etsiAreaNwkInfo 的示例資源結構的框圖；

第 38A 圖是示出用於 areaNwkInstance 的示例資源結構的框圖；

第 38B 圖是示出用於第 38A 圖的 areaNwkInstance 的 6LoWPAN 的子資源描述的示例資源結構的框圖；

第 38C 圖是示出 areaNwkInstance 的示例資源結構的框圖；

第 39 圖是示出 etsiAreaNwkDeviceInventory 的示例資源結構的框圖；

第 40A 圖是示出 deviceInstance 的示例資源結構的框圖；

第 40B 圖是示出 deviceInstance 的示例資源結構的框圖；

第 41A 圖是示出 etsiAreaNwkDeviceGroup 的示例資源結構的框圖；

第 41B 圖是示出 etsiAreaNwkDeviceGroup 的示例資源結構的框圖；

第 42 圖是示出 deviceGroupInstance 的示例資源結構的框圖；

第 43A 圖是示出 etsiGroupMgmtOperations 的示例資源結構的框圖；

第 43B 圖是示出 etsiareaNwkGroupOperations 的示例資源結構的框圖；

第 44A 圖是示出 operationInstance 的示例資源結構的框

圖；

第 44B 圖是示出 operationInstance 的示例資源結構的框圖；

第 45 圖是示出 etsiSensor 的示例資源結構的框圖；以及

第 46 圖是示出 sensorInstance 的示例資源結構的框圖；

第 47 圖是用於執行 xREM 的系統的示例架構的框圖；

第 48A 圖是其中一個或多個公開實施例可被實現的示例通信系統圖；

第 48B 圖是可用在第 48A 圖所示的通信系統中的示例無線發射/接收單元（WTRU）的系統圖；

第 48C 圖是可用在第 48A 圖所示的通信系統中的示例無線電存取網路和示例核心網路的系統圖；

第 48D 圖是可用在第 48A 圖所示的通信系統中的另一個示例無線電存取網路和示例核心網路的系統圖；以及
第 48E 圖是可用在第 48A 圖所示的通信系統中的另一個示例無線電存取網路和示例核心網路的系統圖。

【主要元件符號說明】

QOS：服務品質

M2M：機器對機器通信

dIa、mId、mIa：參考點

SC：服務能力

CoAP：受限應用協定

3GPP：第 3 代合作夥伴計畫

mgmtObjs、402、460：管理物件

REM：遠端實體管理

NREM、38a、38b、502、508、602：M2M 區域網路

ETSI：歐洲電信標準協會

DM：裝置管理

SNMP：簡易網路管理協定

mgmtProtocolType（值）：管理協定類型的值

<scl-of-server>：表示 M2M 伺服器

URI：統一資源識別字

ID：存取權

ACK：授權確認

N-SCL、26a、26b、26c：網路服務能力層

G-SCL：各自的 SCL

<commandInstance>：資源命令的資料結構實例

<command>：各自的命令

PKGURL：相應封包的 URL

<requestInstances>：多個請求實例的集合

execDisable、execPause、execResume、execResult：屬性

execStause：表示<commandInstance>的當前狀態

DA、GA：M2M 應用

OMA：管理協定和開放行動聯盟

（D）：M2M 直接裝置

MO：管理物件

etsiAreaNwkInfo：區域網路資訊和配置

<areaNwkIntance>、<deviceInstance>、6LoWPAN、WiFi、

RFID、ZigBee：子資源

areaNwkID：屬性，用以儲放 M2M 區域網路的標識

areaNwkType：屬性，用以指定 M2M 區域網路的類型

MTU：表示在此區域網路<areaNwkInstance>子資源中的最大傳輸單元

maxRtrAdvertisements：即將發送的未經請求的路由器通告的最大數目

minDelayBetweenRas：發送給所有節點多播地址的兩個連貫的路由器通告之間的最小間隔

maxRaDelayTime：用於發送作為響應於所接收的路由器請求訊息的路由器通告
訊息的最大延期

tentativeNceLifetime：用於試探鄰居快取記憶體計時器

rtrSolicitationInterval：用於第一 maxRtxSolications 路由器請求的初始重傳的間隔

maxRtrSolicitationInterval：初始重傳的數目
maxRtrSolicitations 和用於路由器請求的最大重傳間隔

maxRtrSolicitations：初始重傳的數目

group：組

12：裝置域

14：網路域

16：網路應用域

18a、18b：M2M 網路應用

20：存取和/或核心網路

22：傳輸網路

24a、24b、24c：M2M 伺服器

30a、30b、30c：N-SC

32a、32b、32c：N-SCL 資源結構

DREM、34a、34b、34c、34d、34e、34f、34g、504、512：
M2M 裝置

GREM、36a、36b、(G)、506、510、604：M2M 閘道

40a、40b、40c、40d、40f、40g：M2M 應用 (“DA”)

28a、28b、28c、42a、42b、42c、42d、42e、42f、42g、52a、
52b：網路通信協定棧

44a、44b、44c、44d：D-SCL
50a、50b：M2M 應用（“GA”）
54a、54b：G-SCL
56a、56b：G-SC
58a、58b：G-SCL 資源結構
100：通信系統
102、102a、102b、102c、102d：無線發射/接收單元（WTRU）
104、RAN：無線電存取網路
106：核心網路
108、PSTN：公共交換電話網路
110：網際網路
112：其他網路
114a、114b：基站
116：空氣介面
118：處理器
120：收發器
122：發射/接收元件
124：揚聲器/麥克風
126：鍵盤
128：顯示器/觸控板
130：非移除式記憶體
132：可移除式記憶體
134：電源
136：GPS 晶片組
GPS：全球定位系統
138：週邊設備
140a、140b、140c：節點-B
142：ASN 閘道；行動管理閘道

MME：行動管理閘道
 ASN：存取服務網路
 142a、142b、RNC：無線電網路控制器
 144：媒體閘道；服務閘道；移動 IP 本地代理者
 MIP-HA：移動 IP 本地代理者
 MGW：媒體閘道
 146、MSC：行動交換中心
 146：封包資料網路 (PDN) 閘道；認證、許可、計費 (AAA) 伺服器
 PDN：封包資料網路
 144、MIP-HA
 AAA：認證、許可、計費
 148：服務 GPRS 支援節點；閘道
 SGSN：服務 GPRS 支援節點
 150、GGSN：閘道 GPRS 支援節點
 IuCS、IuPS、Iur、Iub：介面
 200：邏輯管理層
 202：應用管理層
 204：服務管理層
 206：網路管理層
 208：裝置管理層
 210：配置管理
 212：故障 (fault) 管理
 214：性能管理
 300：資源結構架構
 302、sclBase、342、378：SCL 的根資源
 304、344、350、360、404、416、418、430、442：屬性
 306、scls：SCL 子資源

308、346、352：應用（“applications”）子資源
310、362：容器（“containers”）子資源
312、364：組（“groups”）子資源
314、366、410、424、436、448：存取權（“accessRights”）子資源
316、356、370、414、428、440、452：訂閱（“subscriptions”）子資源
318：發現（“discovery”）子資源
320、354、368、412、426、438、450：存取狀態（“accessStatus”）子資源
322、348、358、372、380、382、384、386、406：mgmtObjs子資源
324：appMgmtObjects 子資源
appMgmtObjects：應用管理物件子資源
324-1：從屬於 appMgmtObjects 子資源
326：sclMgmtObjects 子資源
sclMgmtObjects：SCL 管理物件子資源
326-1：從屬於 sclMgmtObjects 子資源
328：nwkMgmtObjects 子資源
nwkMgmtObjects：網路管理物件子資源
328-1：從屬於 nwkMgmtObjects 子資源
330：devMgmtObjects 子資源
devMgmtObjects：裝置管理物件子資源
330-1：從屬於 devMgmtObjects 子資源
332：omaMgmtObjects 子資源
omaMgmtObjects：OMA-DM 管理物件子資源
332-1：從屬於 omaMgmtObjects 子資源
334：bbfMgmtObjects 子資源

bbfMgmtObjects：BBF-TR069 管理物件子資源
334-1：從屬於 bbfMgmtObjects 子資源
340：資源結構構架
400、458：資源結構框架
408：管理物件宣告（“<mgmtObjAnnc>”）子資源
420、422、432、434：參數（<parameters>）子資源
444：預設值的<defaultValue>子資源
446：現行值的<currentValue>子資源
500：執行 xREM 的用戶端-伺服器模型
502-1、510-3：xREM 伺服器
504-1、506-1：xREM 用戶端
510-1：xREM 代理
510-2、512-1：xREM 用戶端
510-4：非 xREM 管理伺服器

發明專利說明書

(本說明書格式、順序，請勿任意更動，※記號部分請勿填寫)

※申請案號：101104398

※申請日：101.2.10

※IPC 分類：H04L 29/06 (2006.01)

一、發明名稱：(中文/英文)

管理機器對機器 (M2M) 實體系統、方法及裝置/Systems,
Methods And Apparatus For Managing Machine-To-Machine
(M2M) Entities

二、中文發明摘要：

公開了用於管理機器對機器 (M2M) 實體的系統、方法和設備。本文所包括的是可包括於管理 M2M 環境中用於管理 M2M 實體的一個或多個管理層的方法。該方法還可包括使用複數管理層來管理 M2M 區域網路，其中 M2M 區域網路可包括一個或多個 M2M 端裝置。M2M 端裝置可包括例如 M2M 閘道和/或 M2M 裝置。管理層可包括應用管理層、服務管理層、網路管理層和裝置管理層的任一個。管理層可提供 M2M 實體的配置管理、故障管理和性能管理的任一個。

三、英文發明摘要：

Systems, methods and apparatus for managing machine-to-machine (M2M) entities are disclosed. Included herein is a method that may include implementing one or more management layers for managing M2M entities in an M2M environment. The method may also include using a plurality of management layers to manage a M2M area network, wherein the M2M area network may include one or more M2M end devices. The M2M end devices may include, for example, an M2M gateway and/or an M2M device. The management layers may include any of an application management layer, service management layer, network management layer and a device management layer. The management layers may provide any of configuration management, fault management, and performance management of the M2M entities.

七、申請專利範圍：

1. 一種方法，包括：

配置一第一機器對機器（M2M）實體，具有用於遠端實體管理（REM）的一服務能力（SC）和具有一資源結構，該資源結構具有用於根據複數管理層中的一個管理層執行一第二 M2M 實體的 REM 的一從屬資源結構；以及

藉由操縱該從屬資源結構的一資源和屬性的任一者來執行該第二 M2M 實體的 REM。

2. 如申請專利範圍第 1 項所述的方法，其中該第一 M2M 實體包括一 M2M 伺服器，並且其中該第二 M2M 實體包括一 M2M 應用、一 M2M 服務能力、一 M2M 區域網路、一 M2M 閘道或一 M2M 裝置。

3. 如申請專利範圍第 1 項所述的方法，其中該複數管理層包括一裝置管理層、一網路管理層、一服務管理層和一應用管理層中至少兩者。

4. 如申請專利範圍第 1 項所述的方法，其中用於一第二 M2M 實體的遠端實體管理的該從屬資源結構包括根據一應用管理層的用於一 M2M 應用的遠端實體管理的一資源結構。

5. 如申請專利範圍第 4 項所述的方法，其中該應用管理層包括用於執行應用生命週期管理的功能。

6. 如申請專利範圍第 1 項所述的方法，其中用於一第二 M2M 實體的遠端實體管理的該從屬資源結構包括根據一

服務管理層的用於一 M2M 服務能力的遠端實體管理的一資源結構。

7· 如申請專利範圍第 6 項所述的方法，其中該服務管理包括用於執行軟體管理和/或韌體管理的功能。

8· 如申請專利範圍第 1 項所述的方法，其中用於一第二 M2M 實體的遠端實體管理的該從屬資源結構包括根據一網路管理層的用於一 M2M 區域網路的遠端實體管理的一資源結構。

9· 如申請專利範圍第 1 項所述的方法，其中用於一第二 M2M 實體的遠端實體管理的該從屬資源結構包括根據一裝置管理層的用於一 M2M 裝置的遠端實體管理的一資源結構。

10· 如申請專利範圍第 1 項所述的方法，其中該複數管理層中的每一個管理層定義用於執行該第二 M2M 實體的配置管理、該第二 M2M 實體的故障管理和該第二 M2M 實體的性能管理中任一者。

11· 如申請專利範圍第 1 項所述的方法，還包括：在該第一 M2M 實體處接收一命令，以操縱該從屬資源結構的該資源和屬性中任一者，其中該第二 M2M 實體的 REM 響應於該命令而執行。

12· 如申請專利範圍第 11 項所述的方法，其中該命令包括一 RESTful 方法。

13. 如申請專利範圍第 1 項所述的方法，還包括：在該第一 M2M 實體處接收一第一命令，以操縱該從屬資源結構的該資源和屬性中任一者，其中執行該第二 M2M 實體的 REM 包括：向該第二 M2M 實體發送一第二命令，以操縱該從屬資源結構的該資源和該屬性中任一者，並且響應於該所接收的第二命令而執行該 M2M 實體的 REM。

14. 如申請專利範圍第 13 項所述的方法，其中該第一和該第二命令兩者都包括一 RESTful 方法。

15. 如申請專利範圍第 13 項所述的方法，其中該第一命令包括一 RESTful 方法，並且其中該第二命令包括一非 RESTful 方法。

16. 如申請專利範圍第 1 項所述的方法，其中執行該第二 M2M 實體的 REM 包括：向該第二 M2M 實體發送一命令，以操縱該從屬資源結構的該資源和屬性中任一者，並且響應於該所接收的命令而執行該第二 M2M 實體的 REM。

17. 如申請專利範圍第 16 項所述的方法，其中該命令包括一非 RESTful 方法。

18. 如申請專利範圍第 16 項所述的方法，其中該命令包括一 RESTful 方法。

19. 如申請專利範圍第 1 項所述的方法，其中該第二 M2M 實體包括該從屬資源結構的一副本，並且其中執行該

第二 M2M 實體的 REM 包括操縱該從屬資源結構的該副本以在操縱該從屬資源結構的所述資源和屬性中任一者之後複製該從屬資源結構。

20．一種設備，包括：

一第一機器對機器（M2M）實體，被配置具有用於遠端實體管理（REM）的服務能力和具有一資源結構，該資源結構具有用於根據複數管理層中的一個管理層執行一第二 M2M 實體的 REM 的一從屬資源結構；以及

一處理器，適於藉由操縱該從屬資源結構的一資源和屬性的任一者來執行該第二 M2M 實體的 REM。

21．如申請專利範圍第 20 項所述的設備，其中該第一 M2M 實體包括一 M2M 伺服器，並且其中該第二 M2M 實體包括一 M2M 應用、一 M2M 服務能力、一 M2M 區域網路、一 M2M 閘道或一 M2M 裝置。

22．如申請專利範圍第 20 項所述的設備，其中該複數管理層包括一裝置管理層、一網路管理層、一服務管理層和一應用管理層中至少二者。

23．如申請專利範圍第 20 項所述的設備，其中用於一第二 M2M 實體的遠端實體管理的該從屬資源結構包括根據一應用管理層的用於一 M2M 應用的遠端實體管理的一資源結構。

24．如申請專利範圍第 23 項所述的設備，其中該應用管理層包括用於執行應用生命週期管理的功能。

25．如申請專利範圍第 20 項所述的設備，其中用於一第二 M2M 實體的遠端實體管理的該從屬資源結構包括根據一服務管理層的用於一 M2M 服務能力的遠端實體管理的一資源結構。

26．如申請專利範圍第 25 項所述的設備，其中該服務管理包括用於執行軟體管理和/或韌體管理的功能。

27．如申請專利範圍第 20 項所述的設備，其中用於一第二 M2M 實體的遠端實體管理的該從屬資源結構包括根據一網路管理層的用於一 M2M 區域網路的遠端實體管理的一資源結構。

28．如申請專利範圍第 20 項所述的設備，其中用於一第二 M2M 實體的遠端實體管理的該從屬資源結構包括根據一裝置管理層的用於一 M2M 裝置的遠端實體管理的一資源結構。

29．如申請專利範圍第 20 項所述的設備，其中該複數管理層中的每一個管理層定義用於執行該第二 M2M 實體的配置管理、該第二 M2M 實體的故障管理和該第二 M2M 實體的性能管理中任一者。

30．如申請專利範圍第 20 項所述的設備，其中該第一 M2M 實體還包括：用於接收一命令以操縱該從屬資源結構的該資源和屬性中任一者的一 SC，其中該第二 M2M 實體的 REM 響應於該命令而執行。

31·如申請專利範圍第 20 項所述的設備，其中該命令包括一 RESTful 方法。

32·如申請專利範圍第 20 項所述的設備，其中：

該第一 M2M 實體還包括：

用於接收一第一命令以操縱該從屬資源結構的該資源和屬性中任一者的一 SC；以及

用於向該第二 M2M 實體傳送一第二命令以操縱在該第二 M2M 實體處所維護的該從屬資源結構的一副本的一資源副本和一屬性副本中任一者的 SC；以及

該處理器還適於向該第二 M2M 實體發送該第二命令，以引起對響應於該第二命令，在該第二 M2M 實體處所維護的該從屬資源結構的該副本的該資源副本和該屬性副本中任一者的操縱。

33·如申請專利範圍第 32 項所述的設備，其中該第一和該第二命令兩者都包括一 RESTful 方法。

34·如申請專利範圍第 32 項所述的設備，其中該第一命令包括一 RESTful 方法，並且其中該第二命令包括一非 RESTful 方法。

35·如申請專利範圍第 20 項所述的設備，其中：

該第一 M2M 實體還包括：

一 SC，用於向該第二 M2M 實體傳送一命令，以操縱在該第二 M2M 實體處被維護的該從屬資源結構的一副本的一資源副本和一屬性副本中任一者；以及

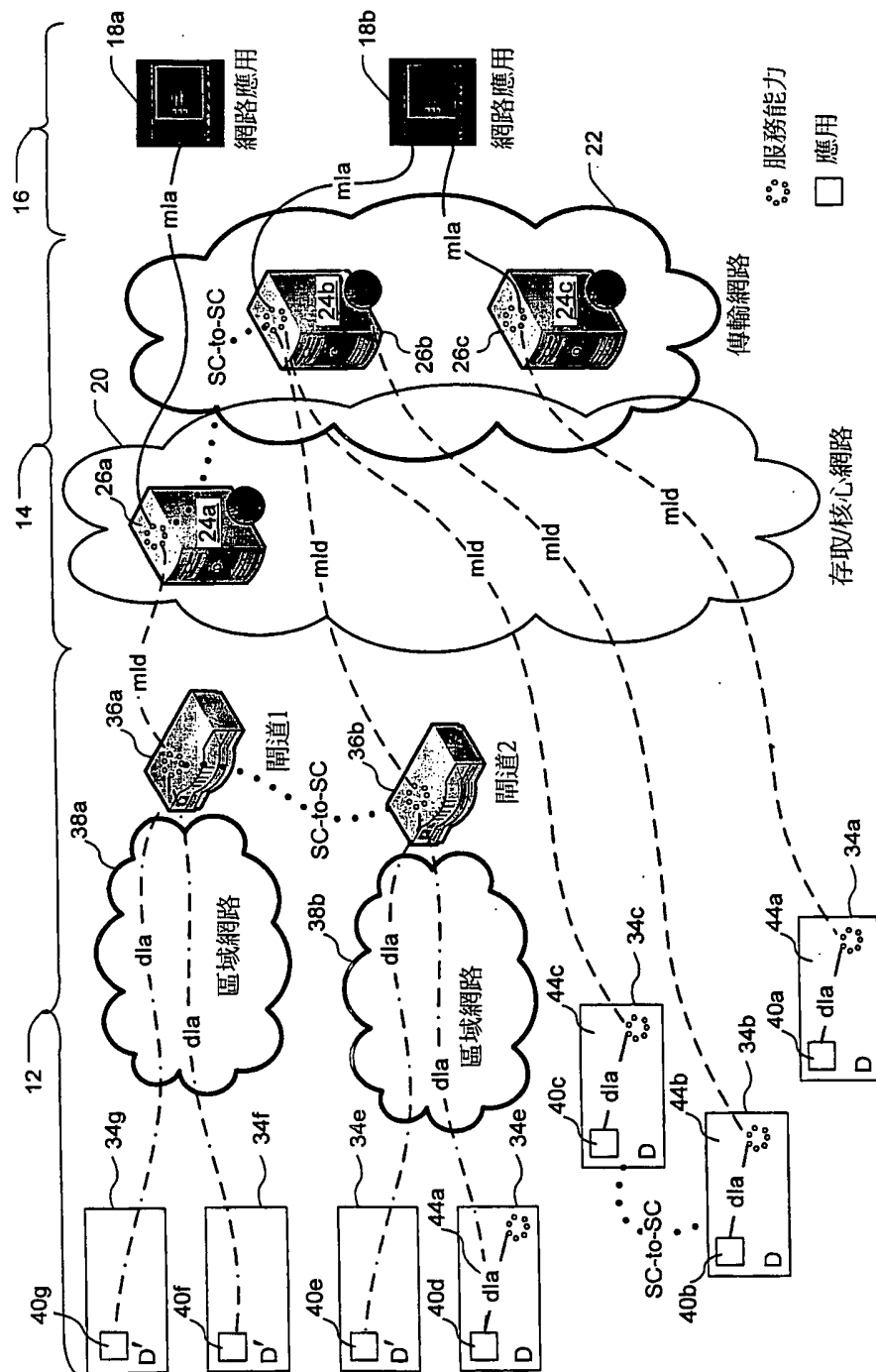
該處理器還適於向該第二 M2M 實體發送該命令，以引起對響應於一第二命令，在該第二 M2M 實體處所維護的該

從屬資源結構的該副本的該資源副本和該屬性副本中任一者的操縱。

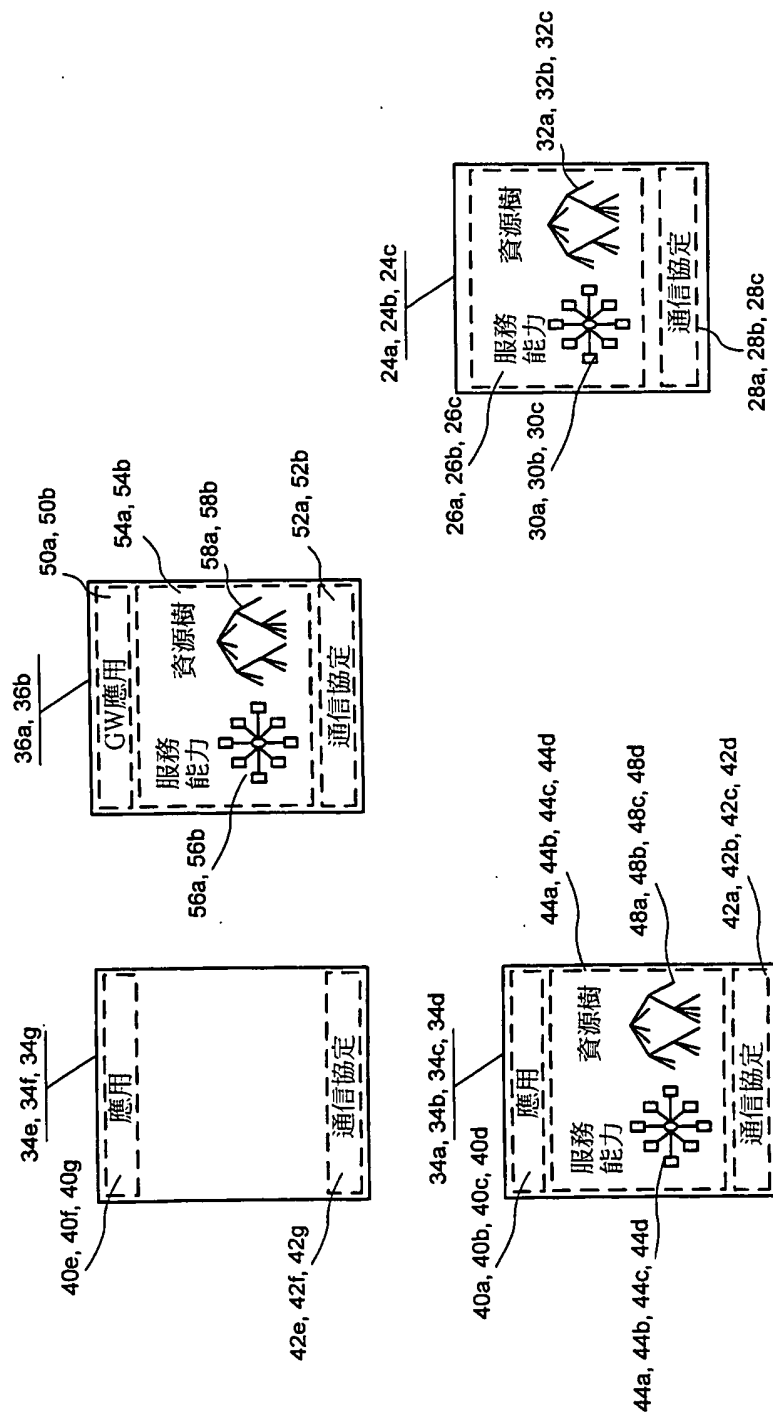
36．如申請專利範圍第 35 項所述的設備，其中該命令包括一非 RESTful 方法。

37．如申請專利範圍第 35 項所述的設備，其中該命令包括一 RESTful 方法。

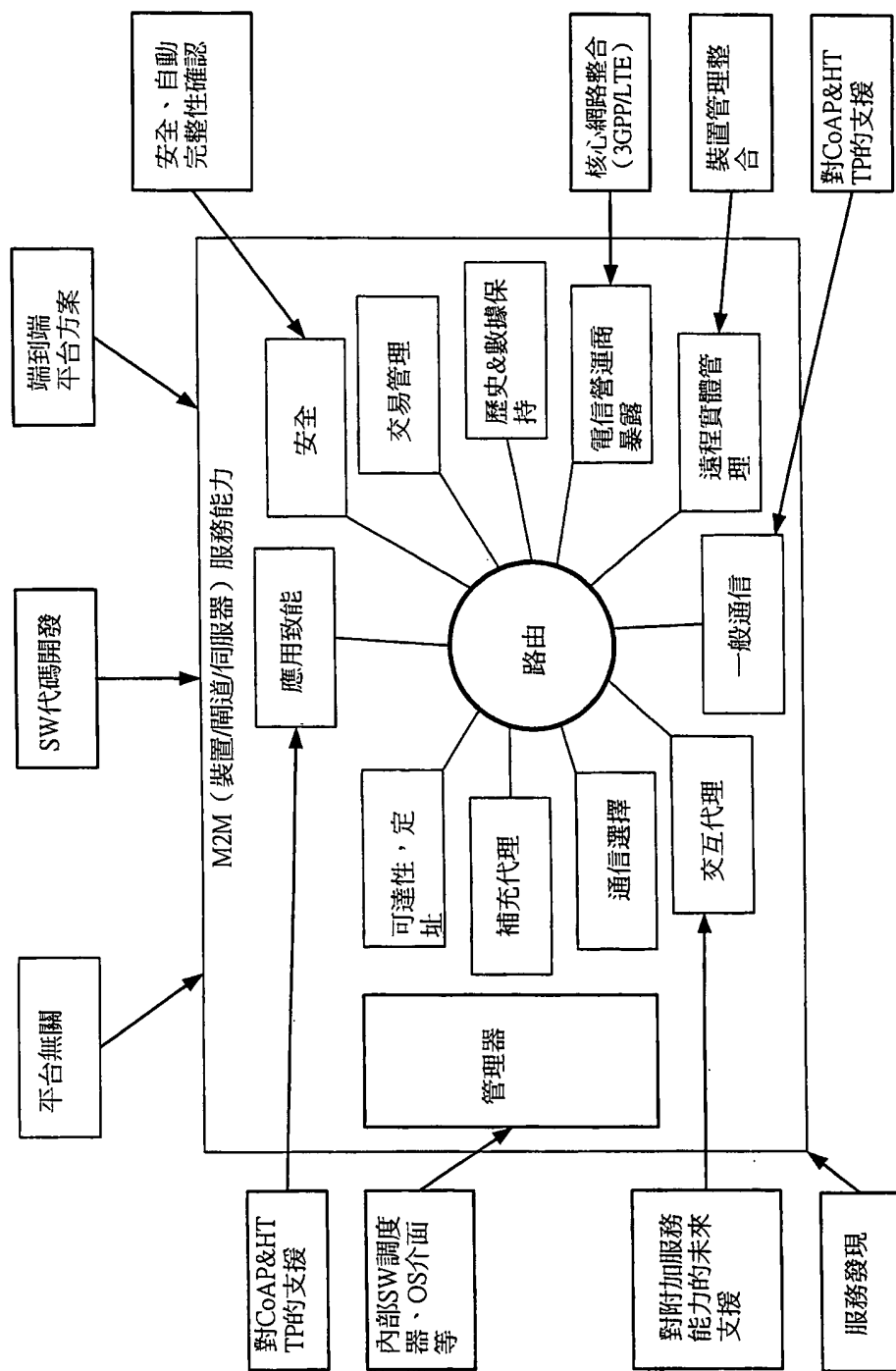
八、圖式：



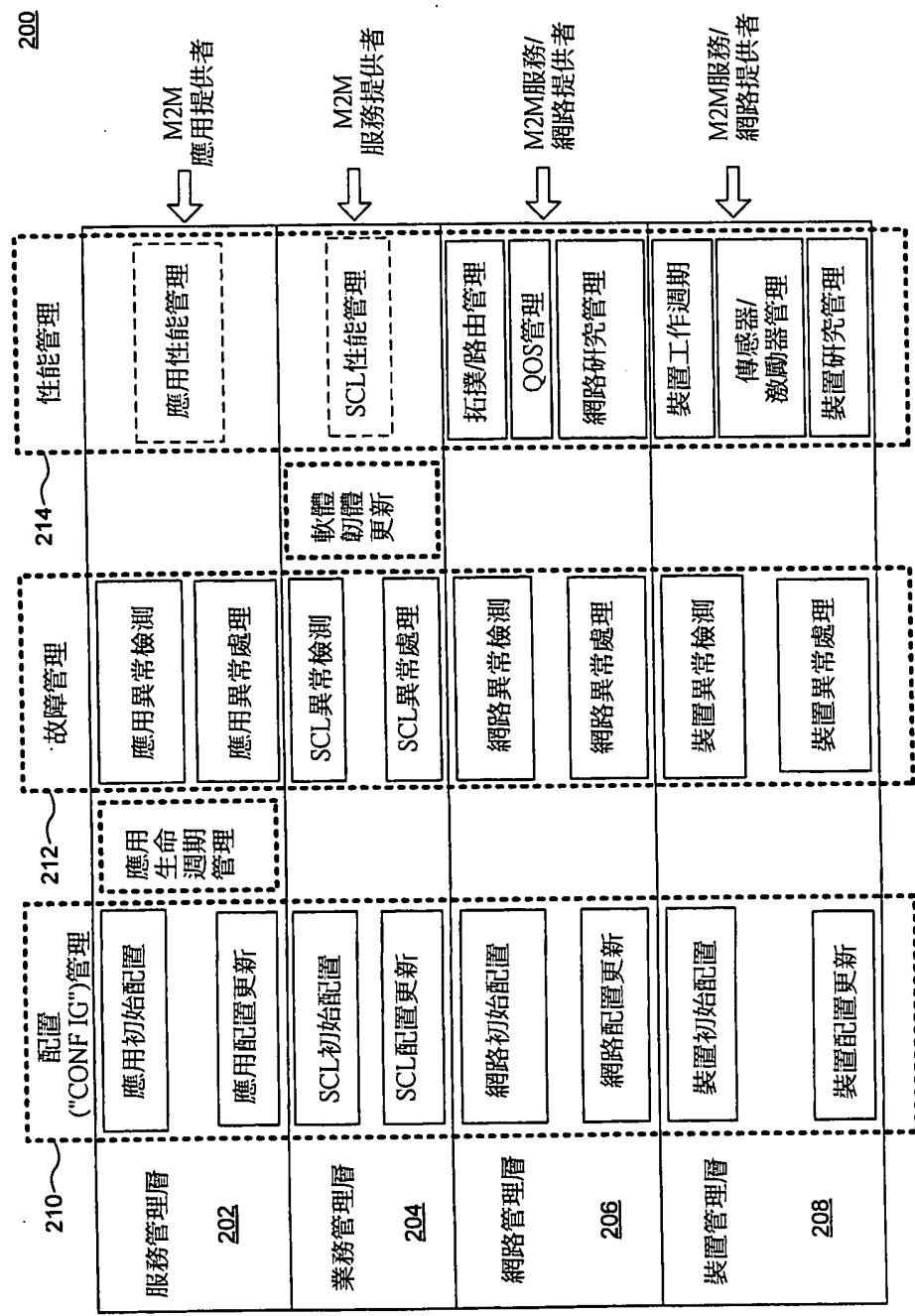
第1A圖



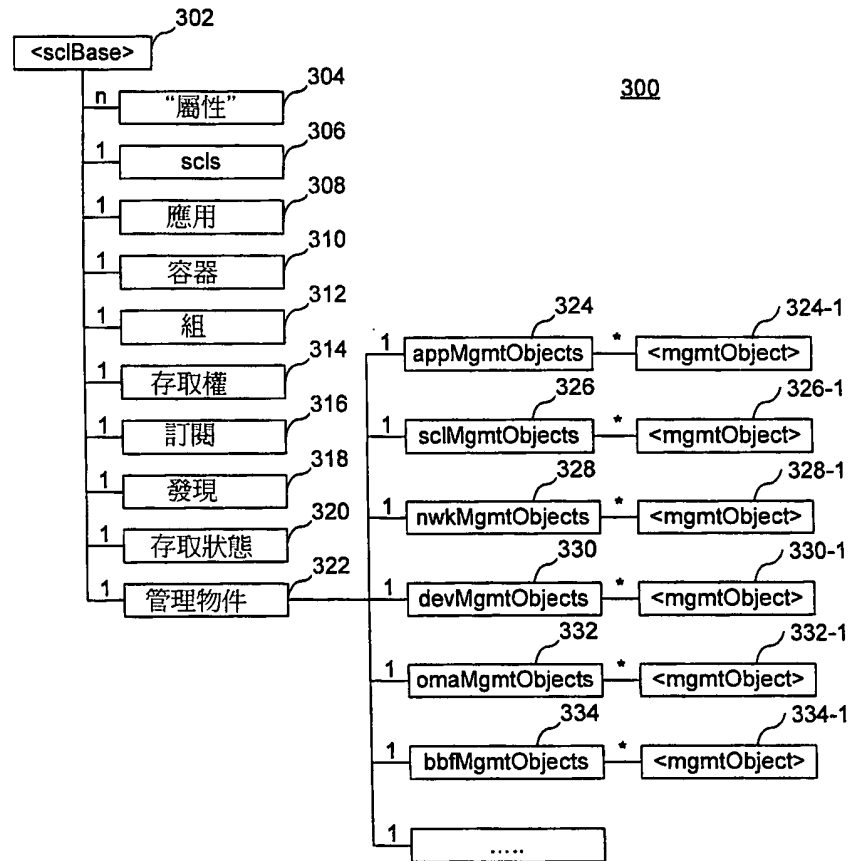
第1B圖



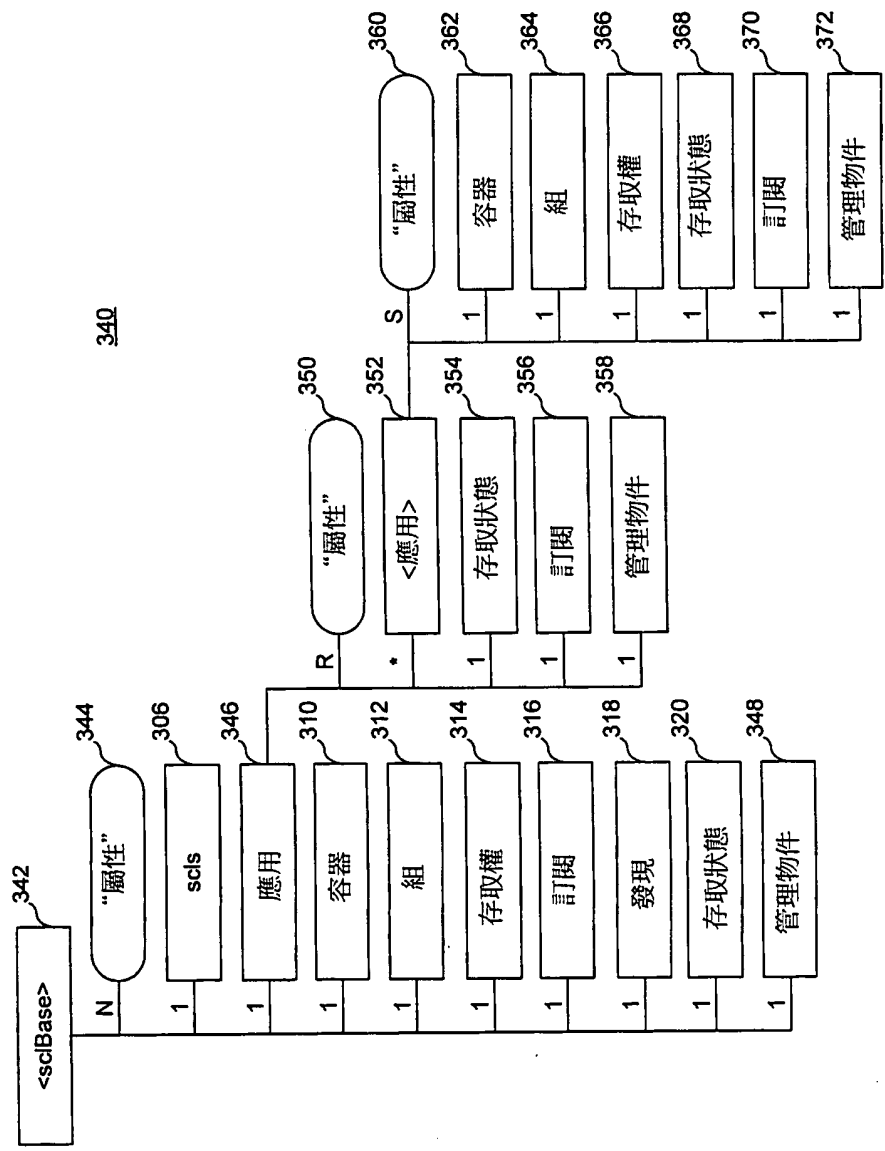
第1C圖



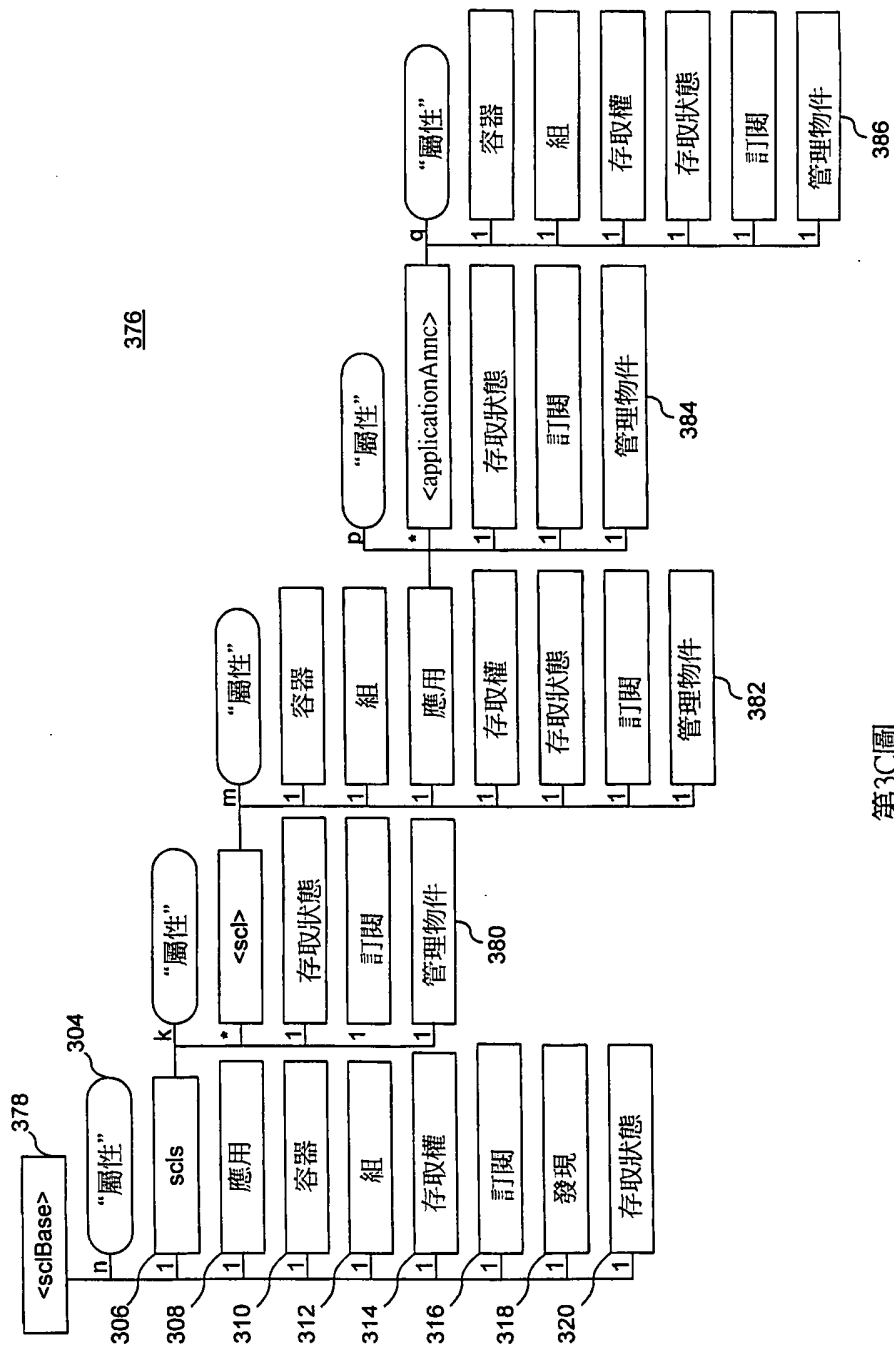
第2圖



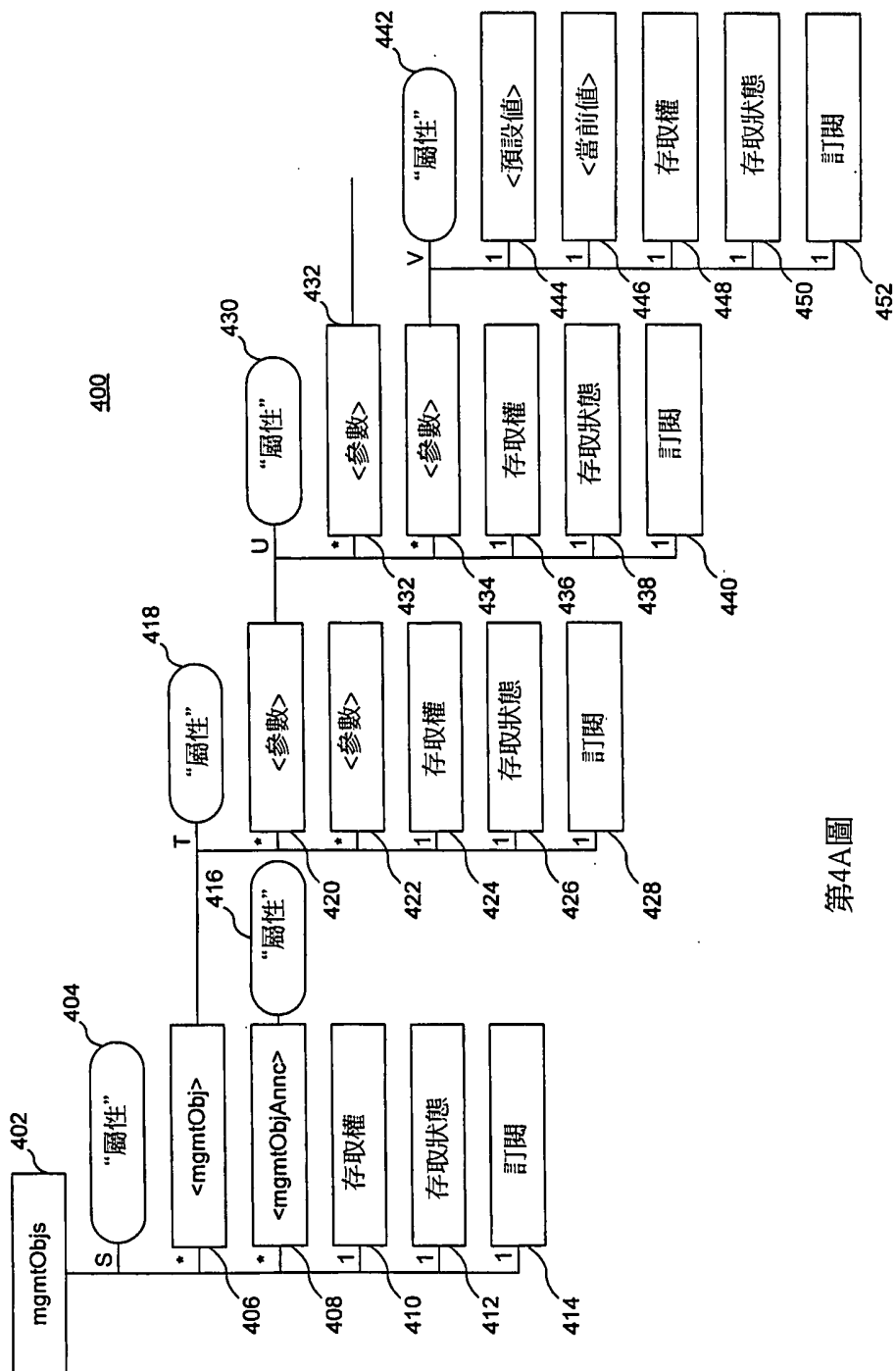
第3A圖



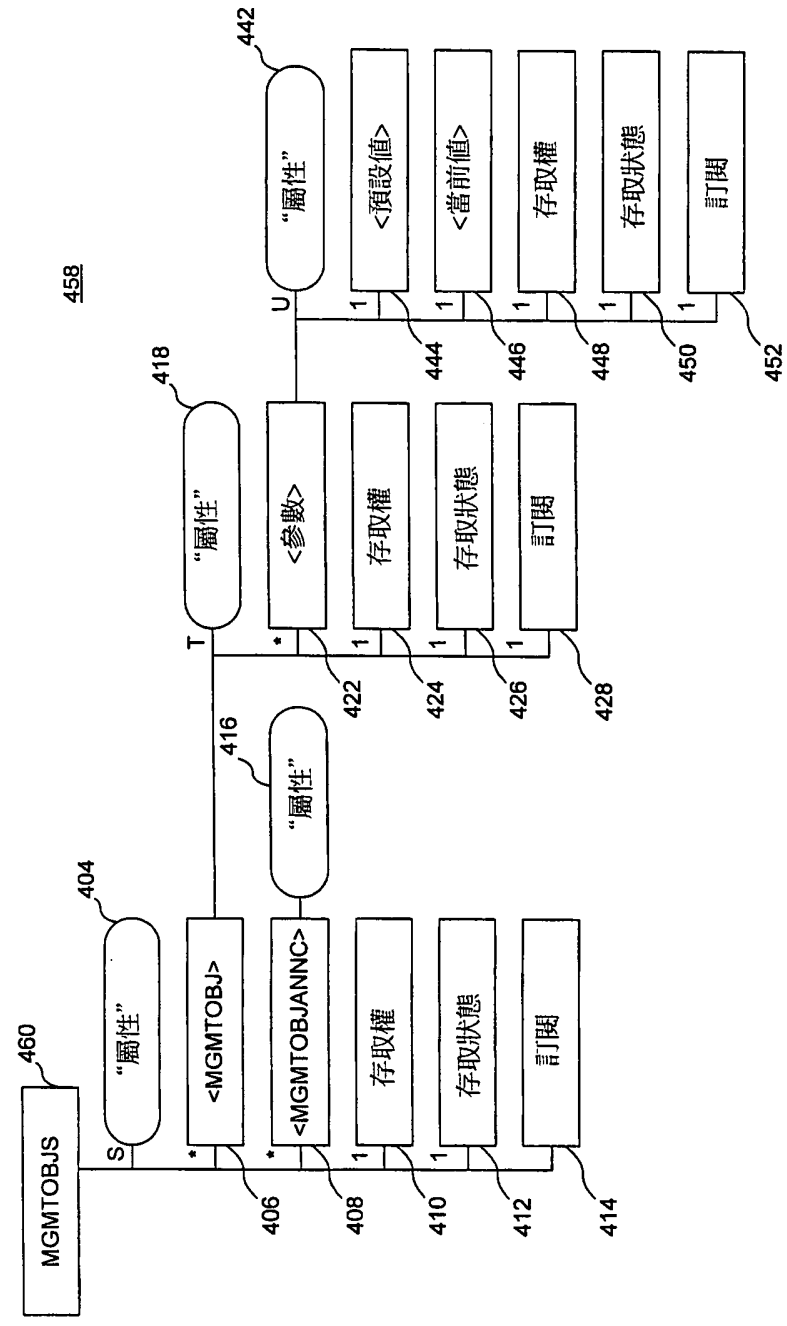
第3B圖



第3C圖

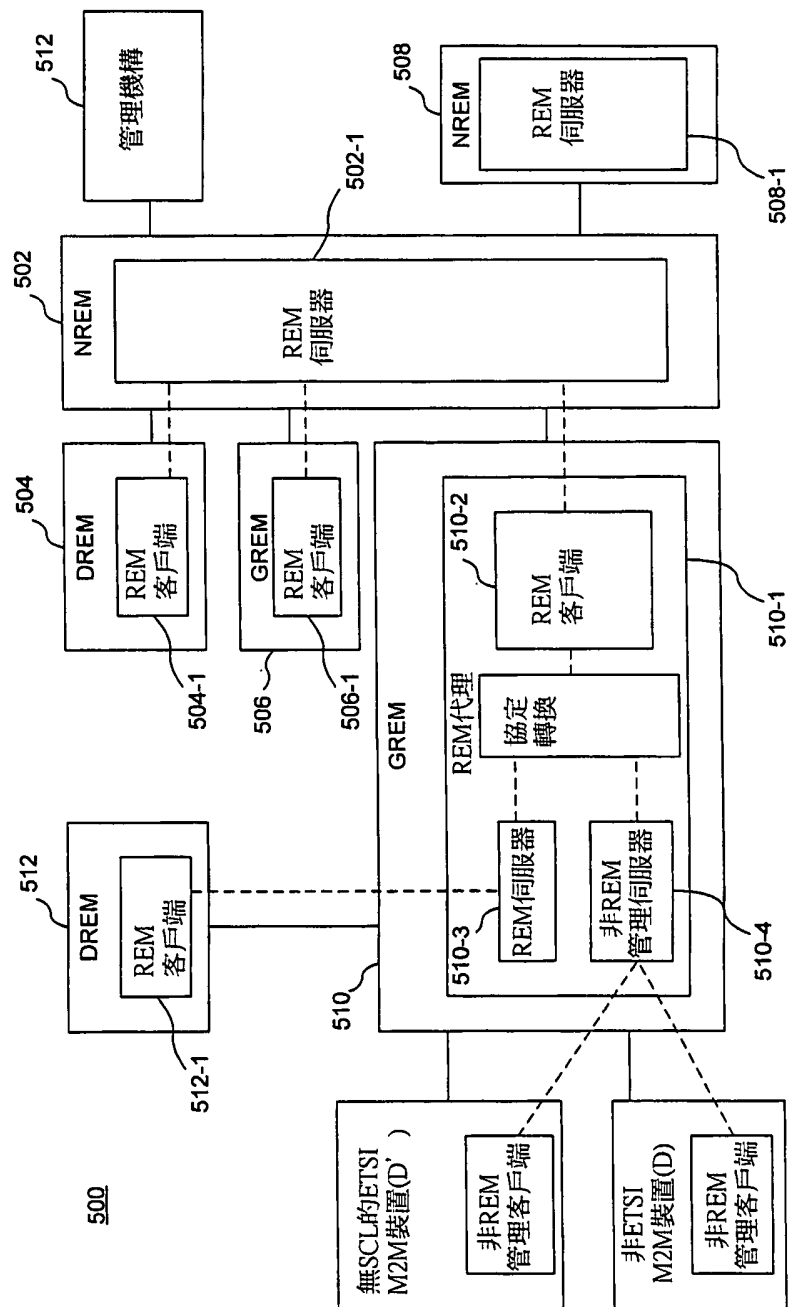


第4A圖

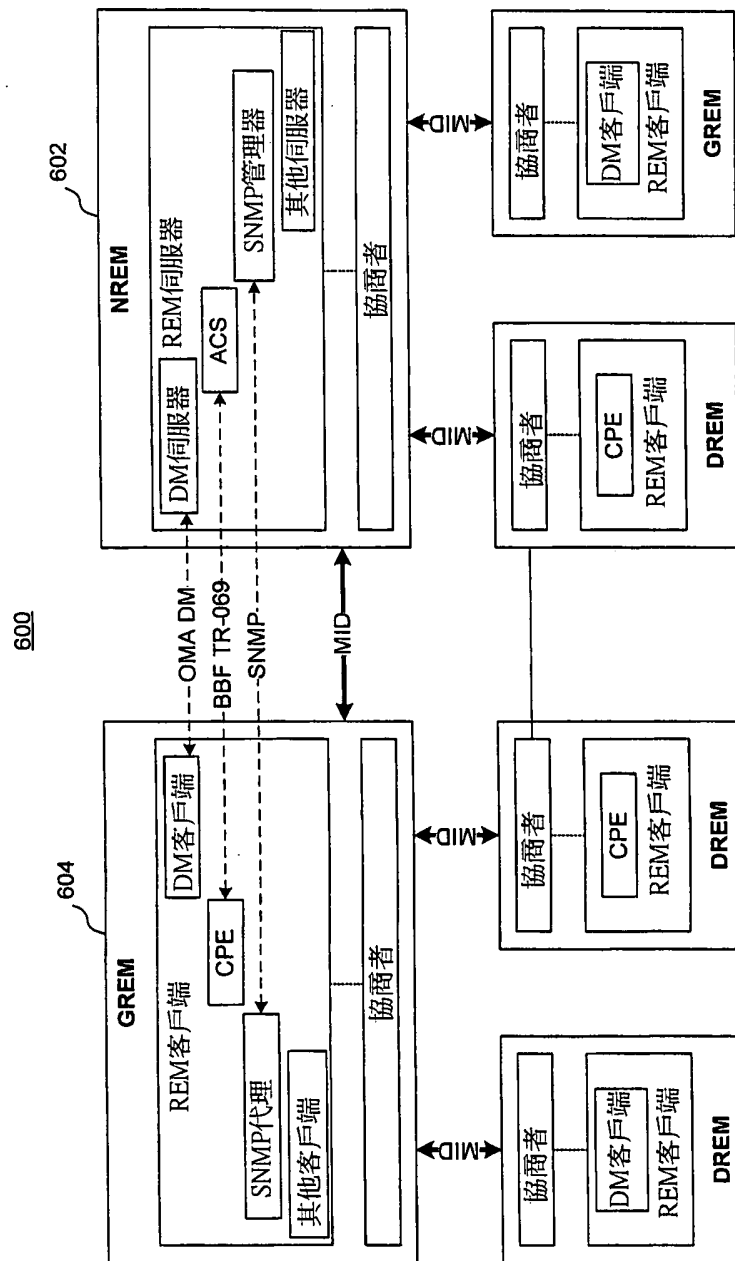


第4B圖

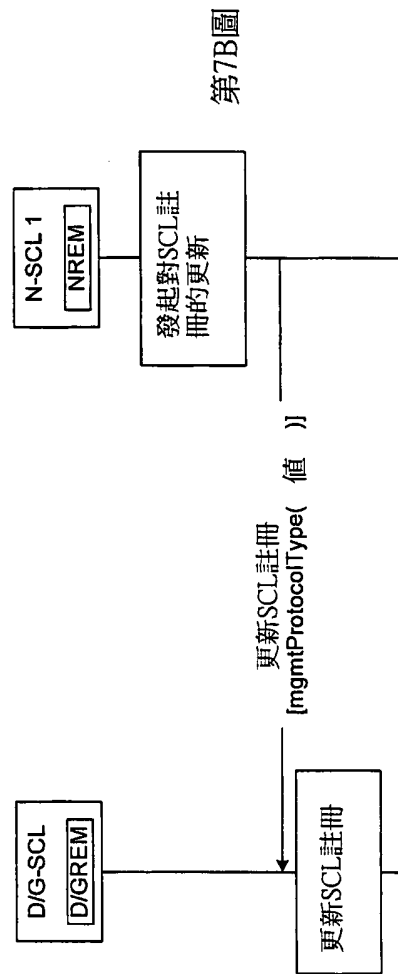
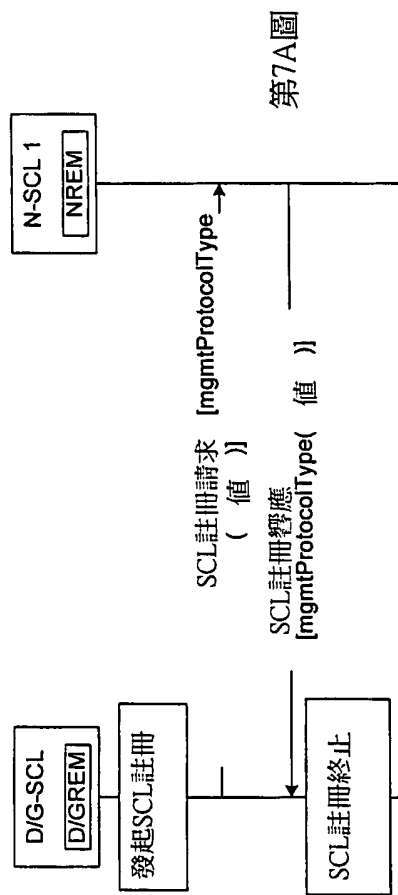


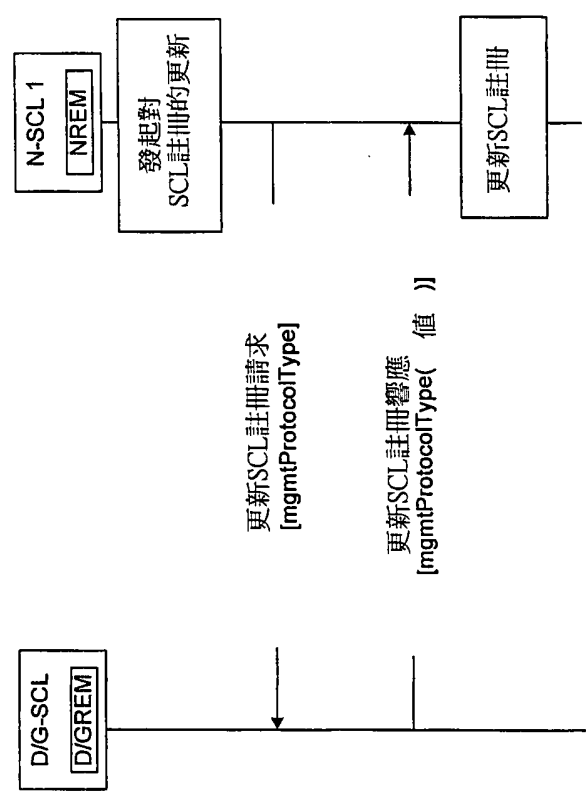


第5圖

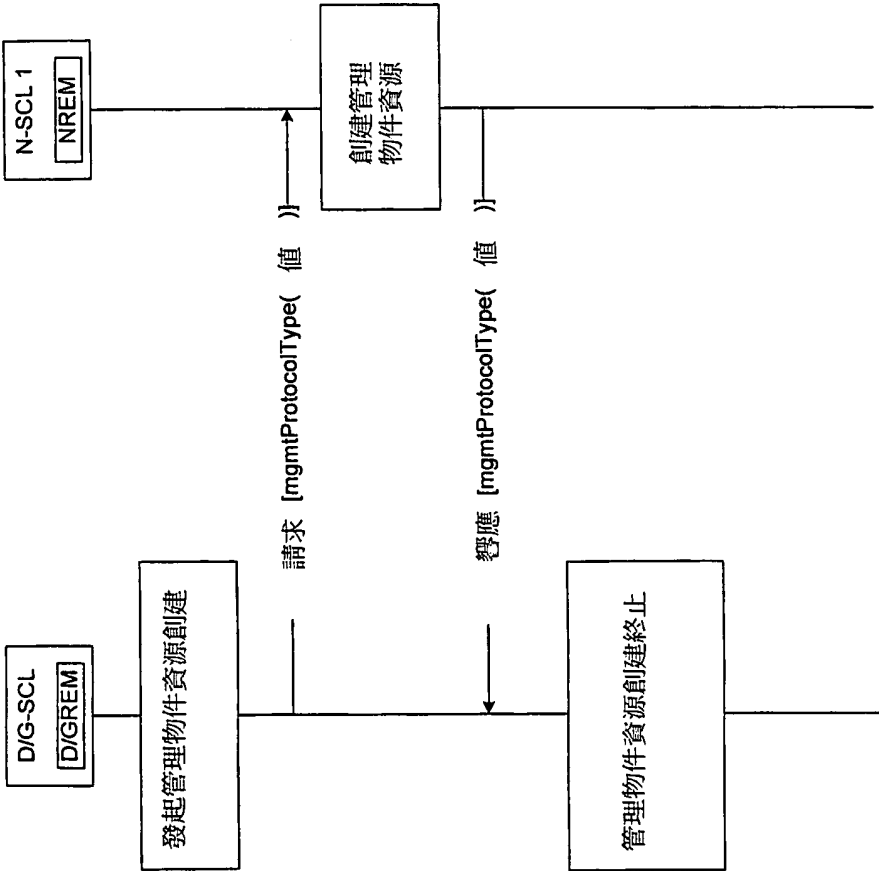


第6圖

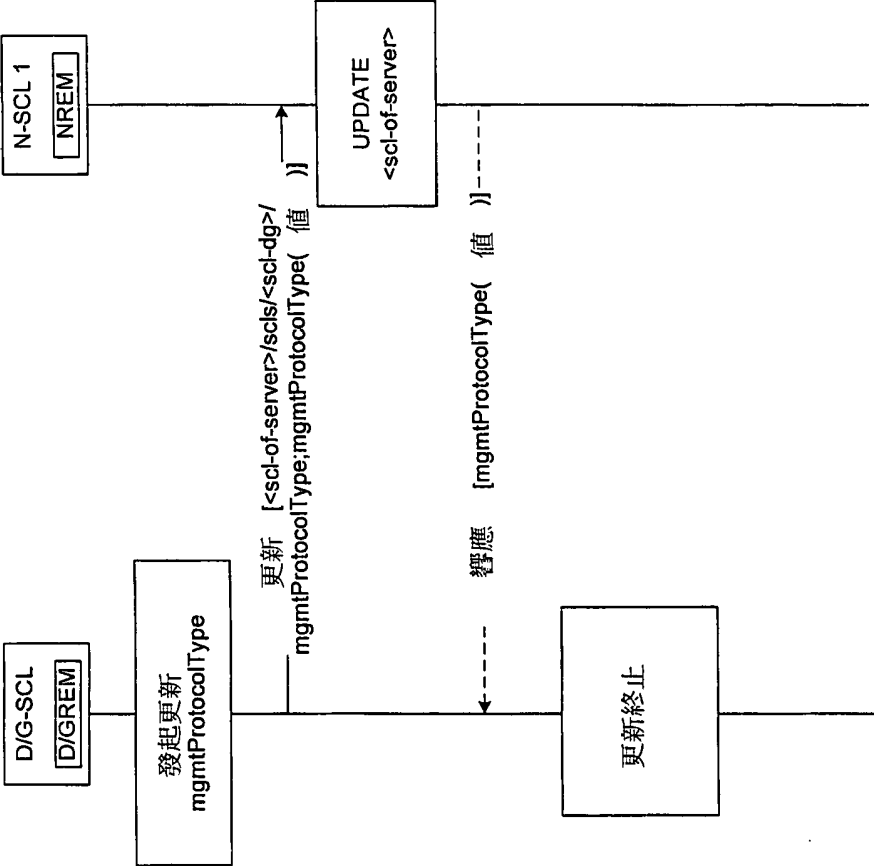




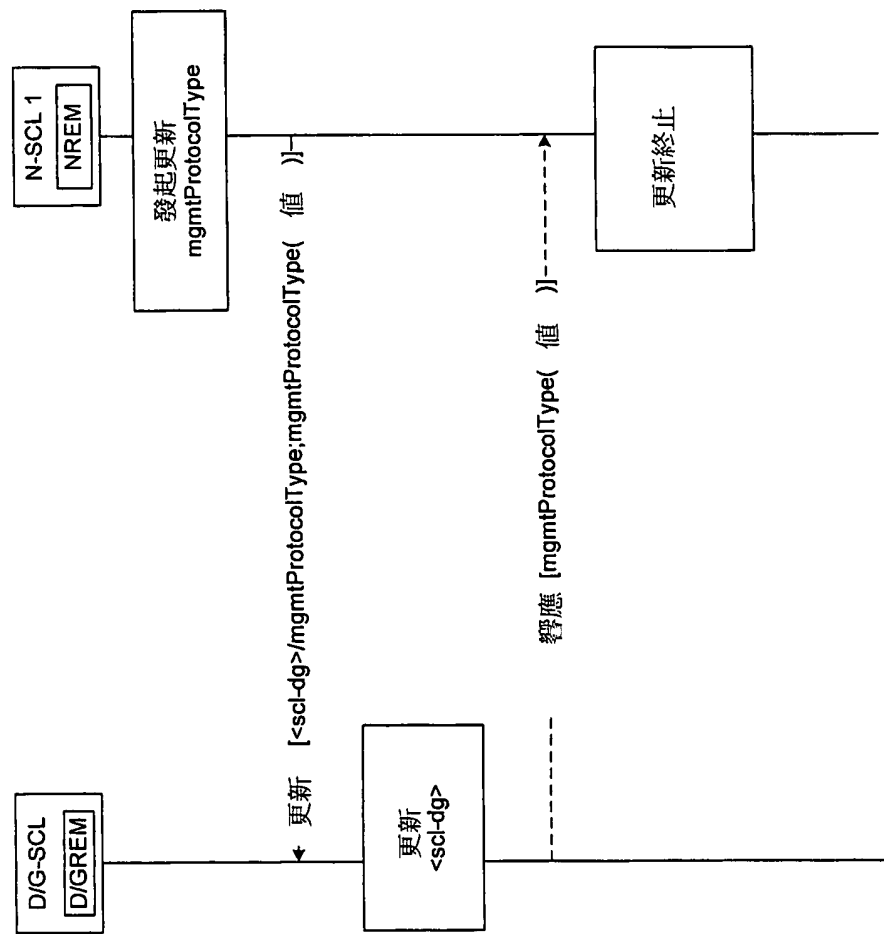
第7C圖



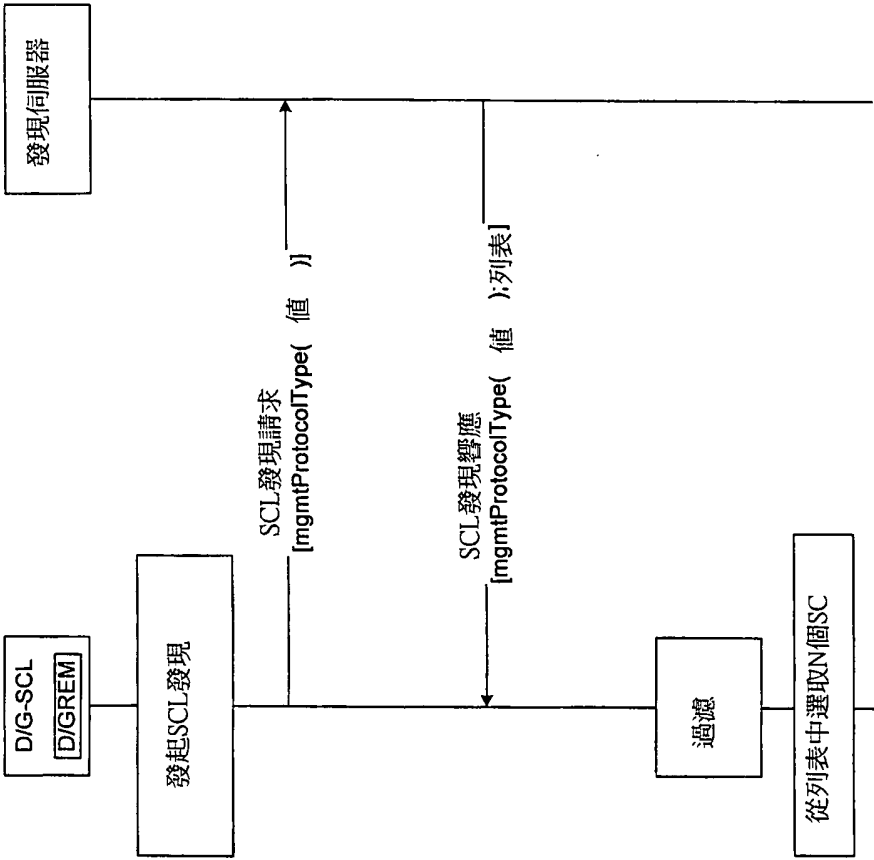
第8圖



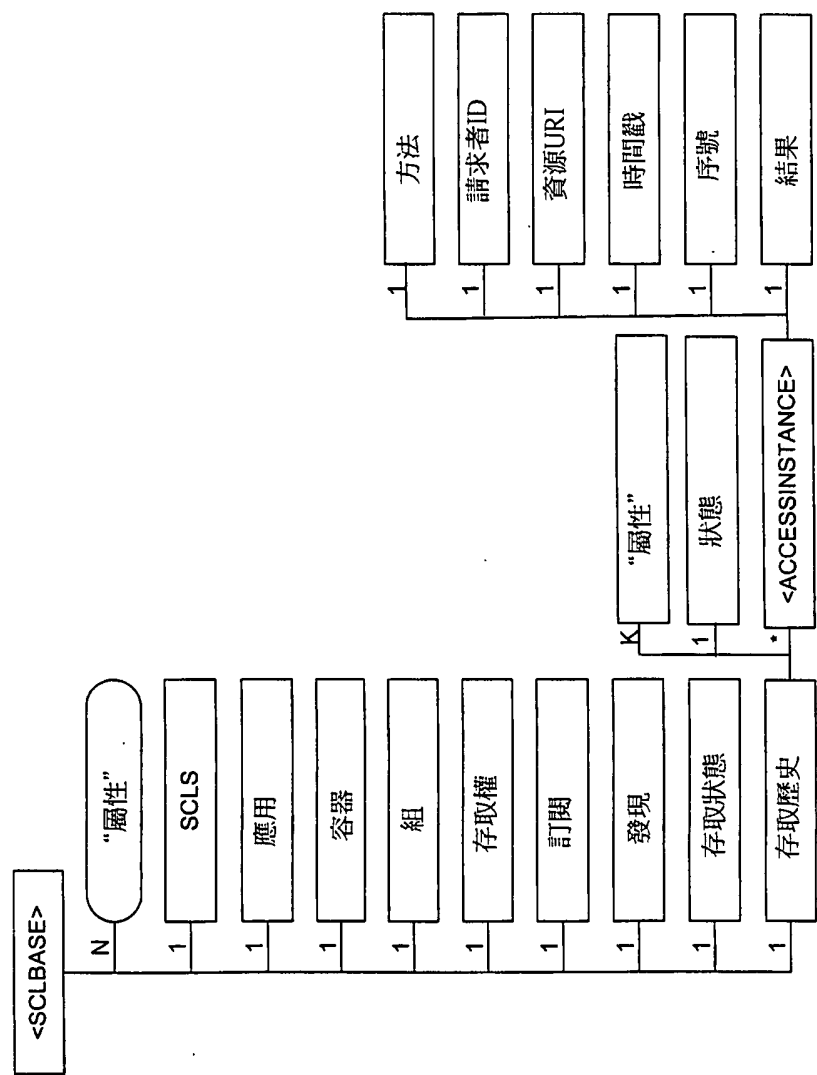
第9圖



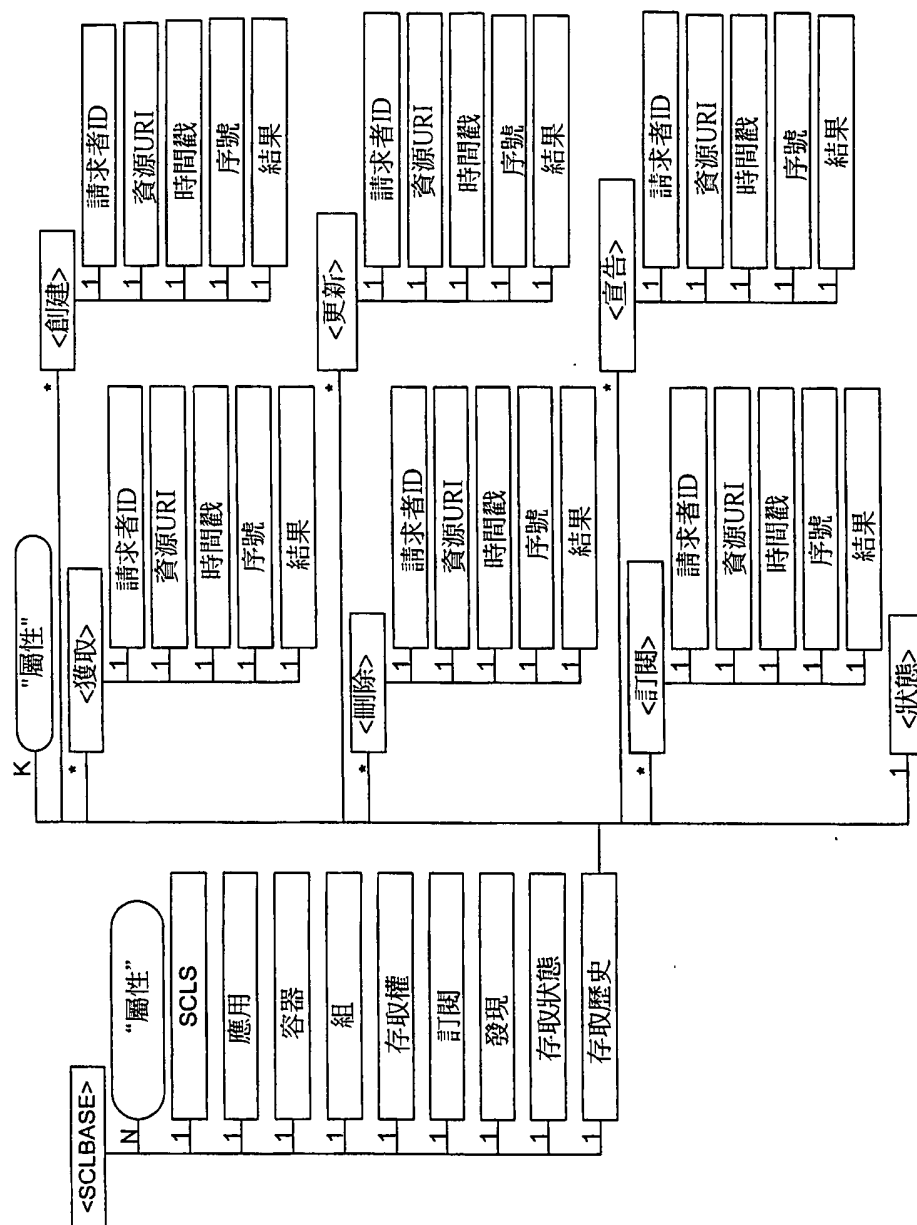
第10圖



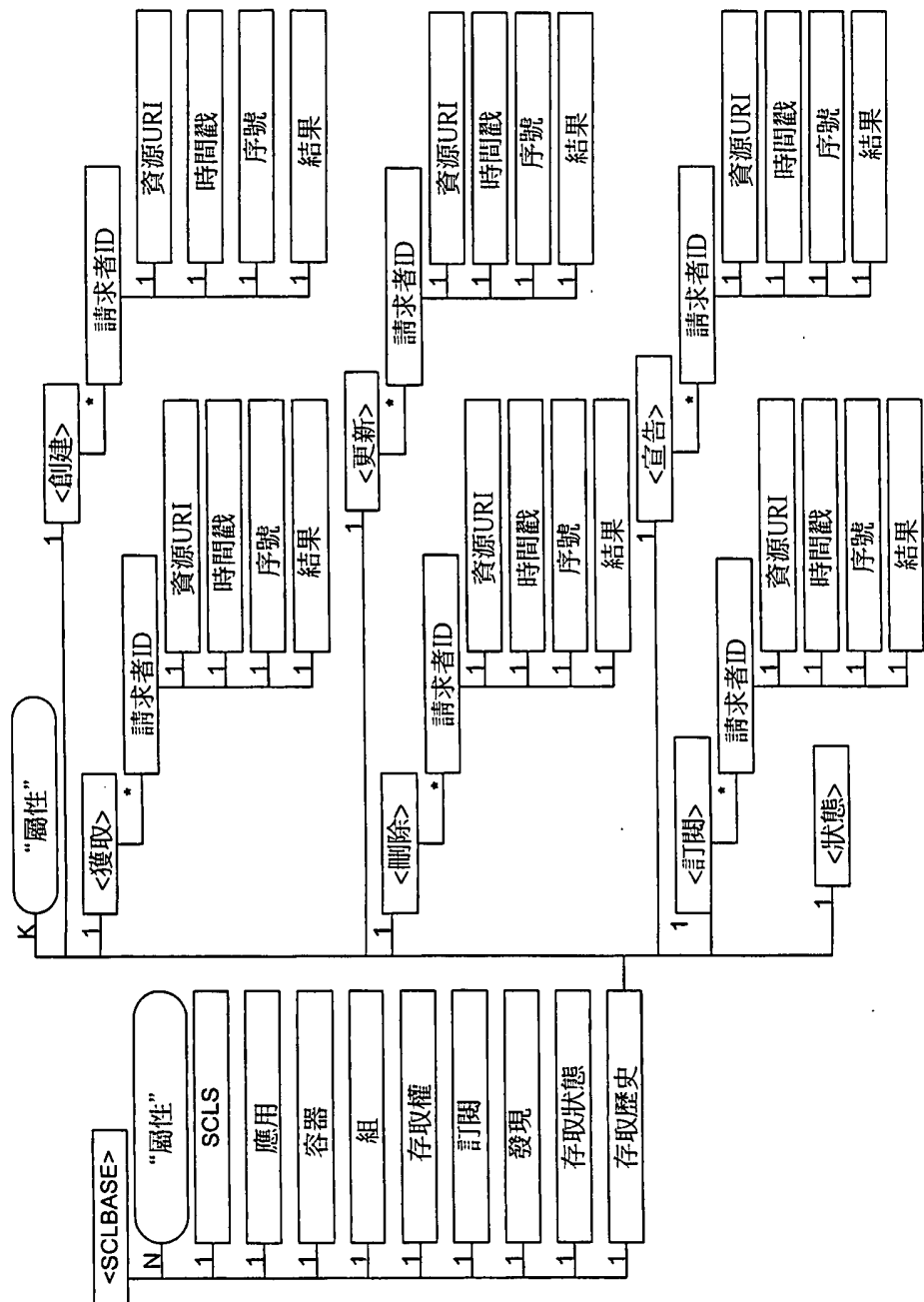
第11圖



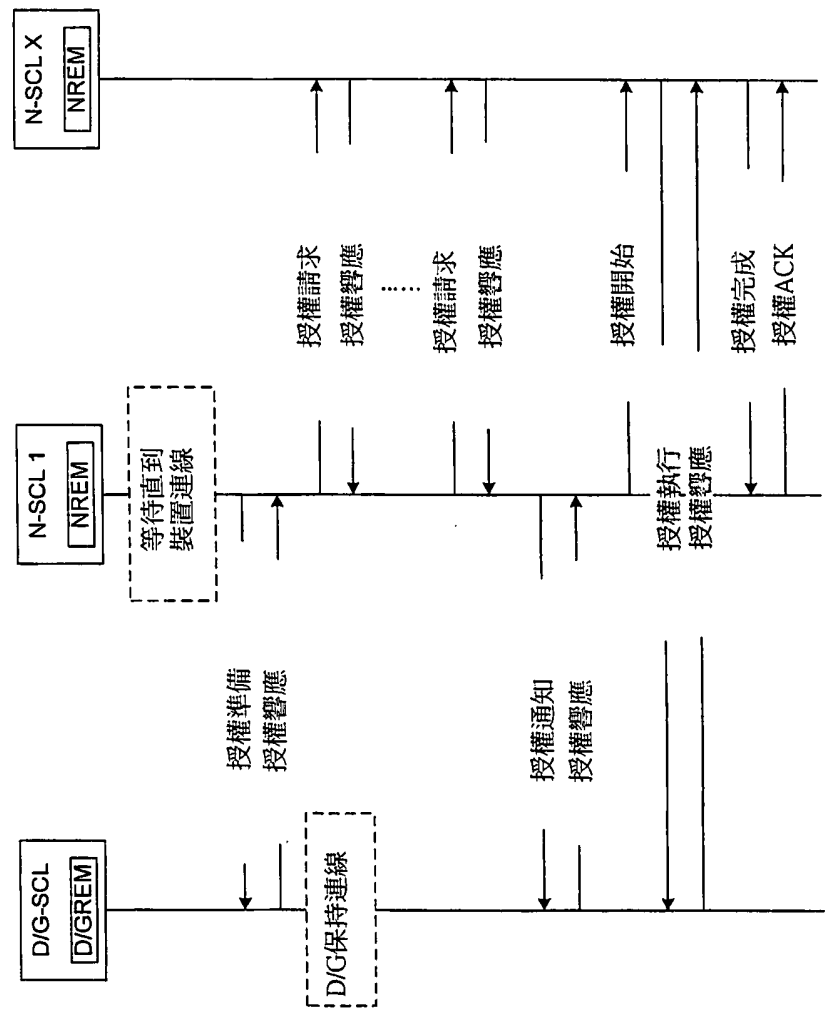
第12圖



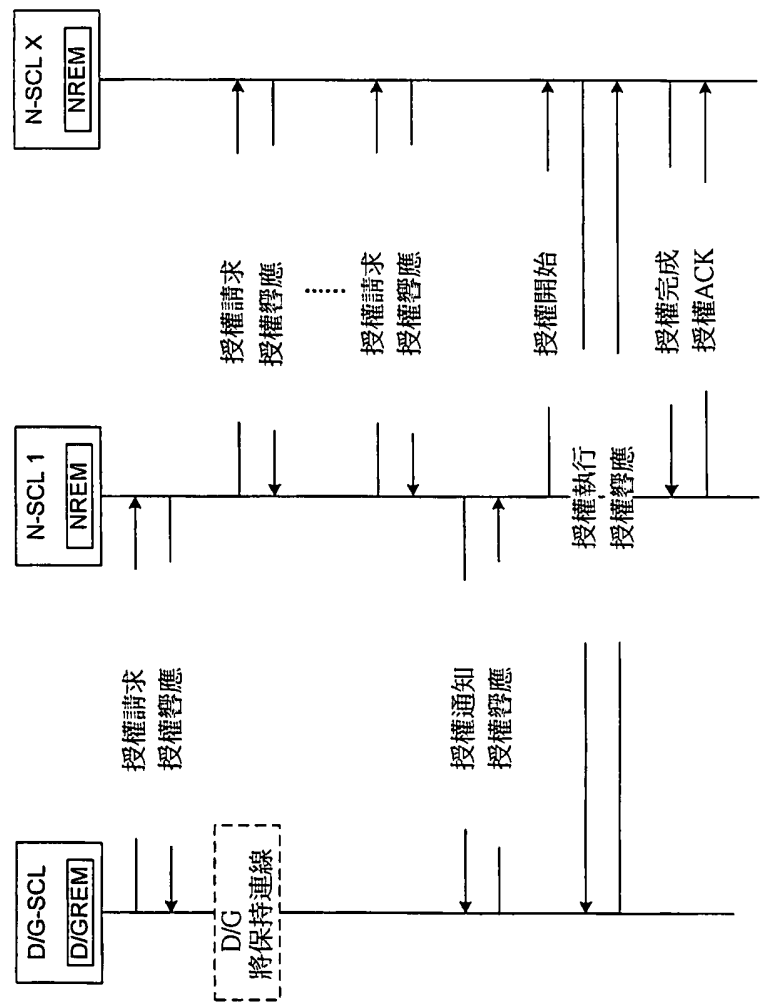
第13圖



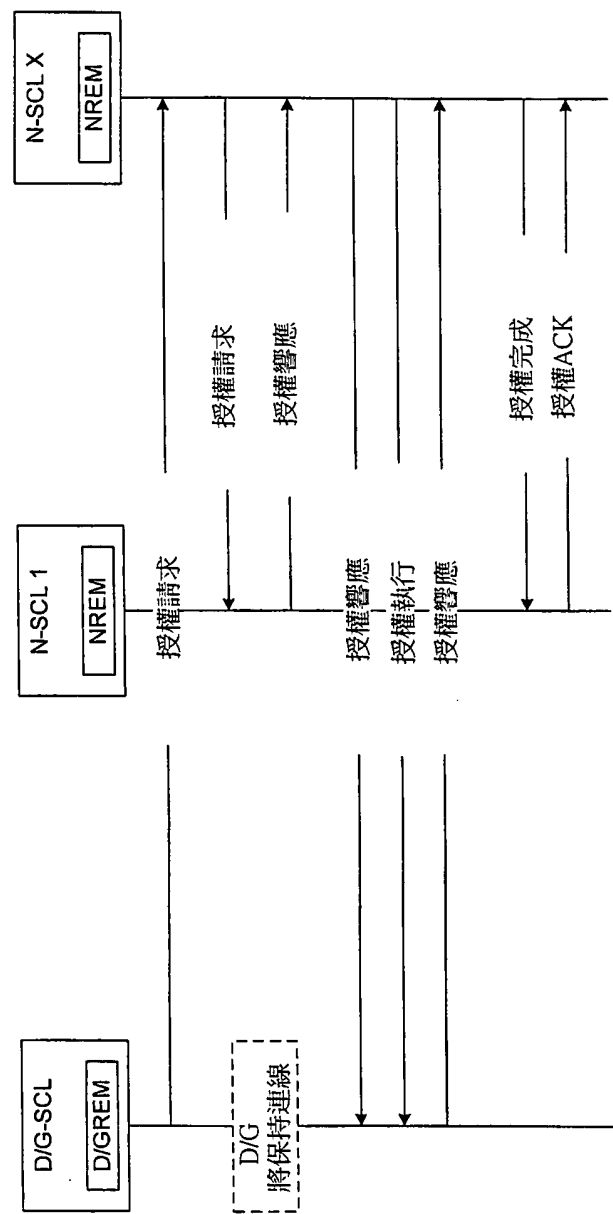
第14圖



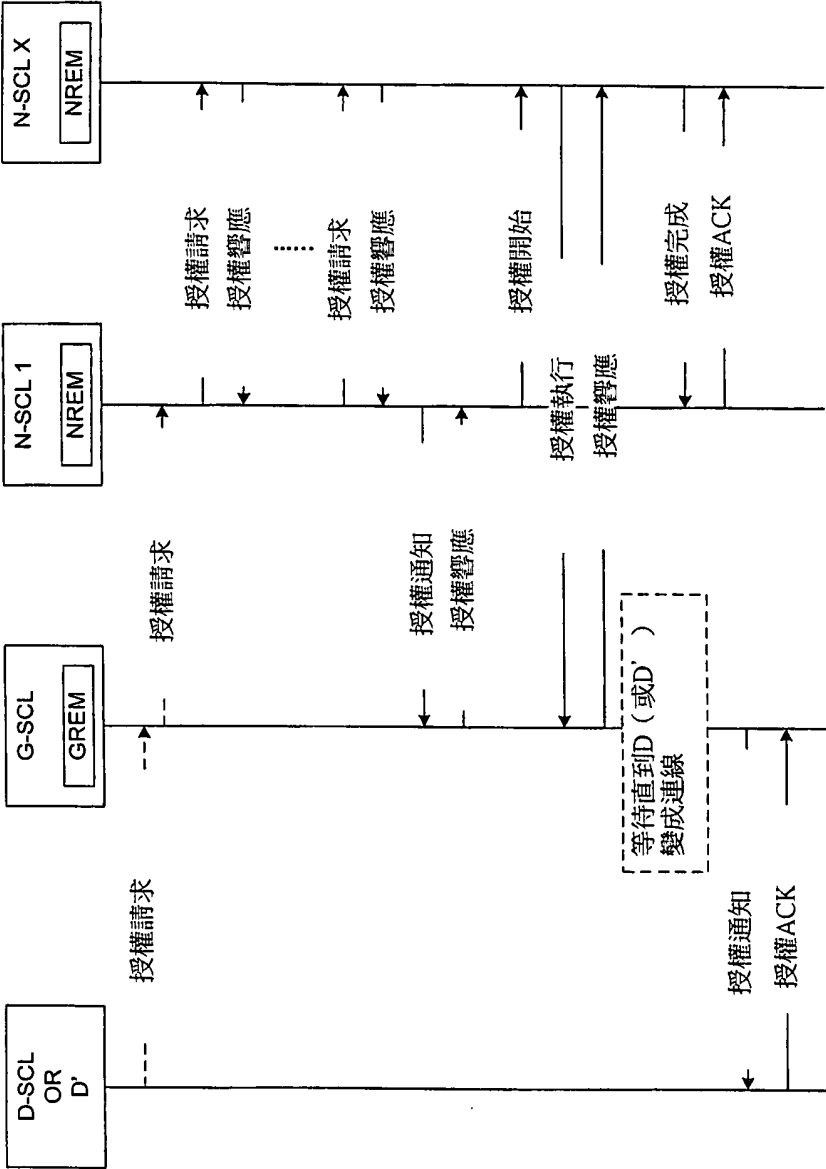
第15圖



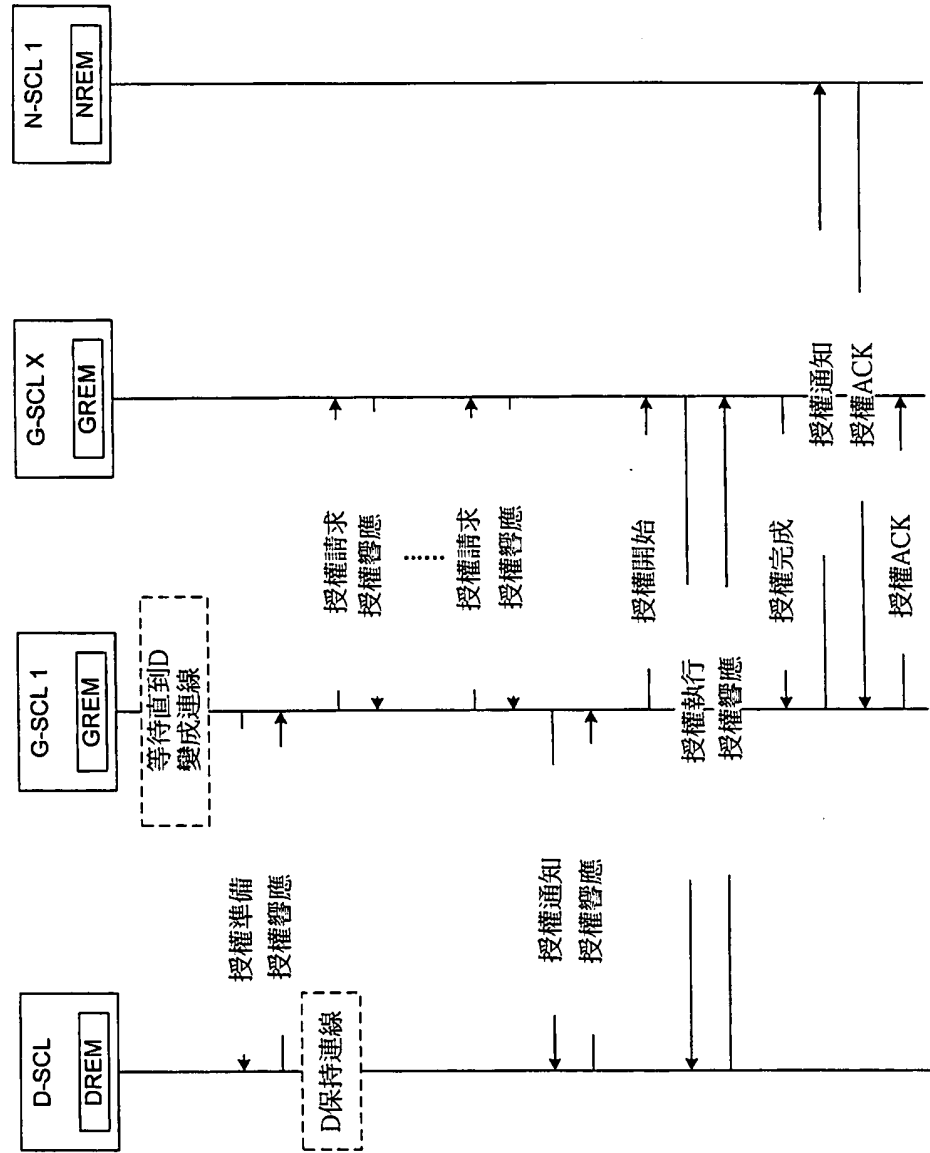
第16圖



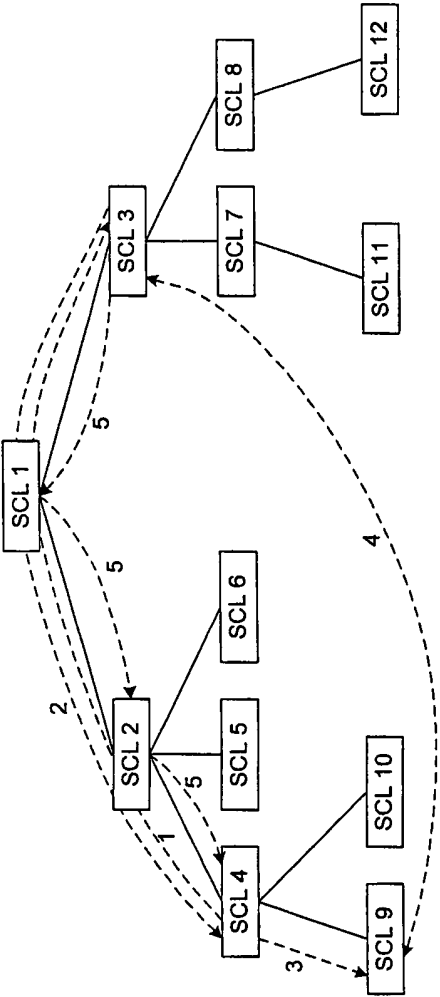
第17圖



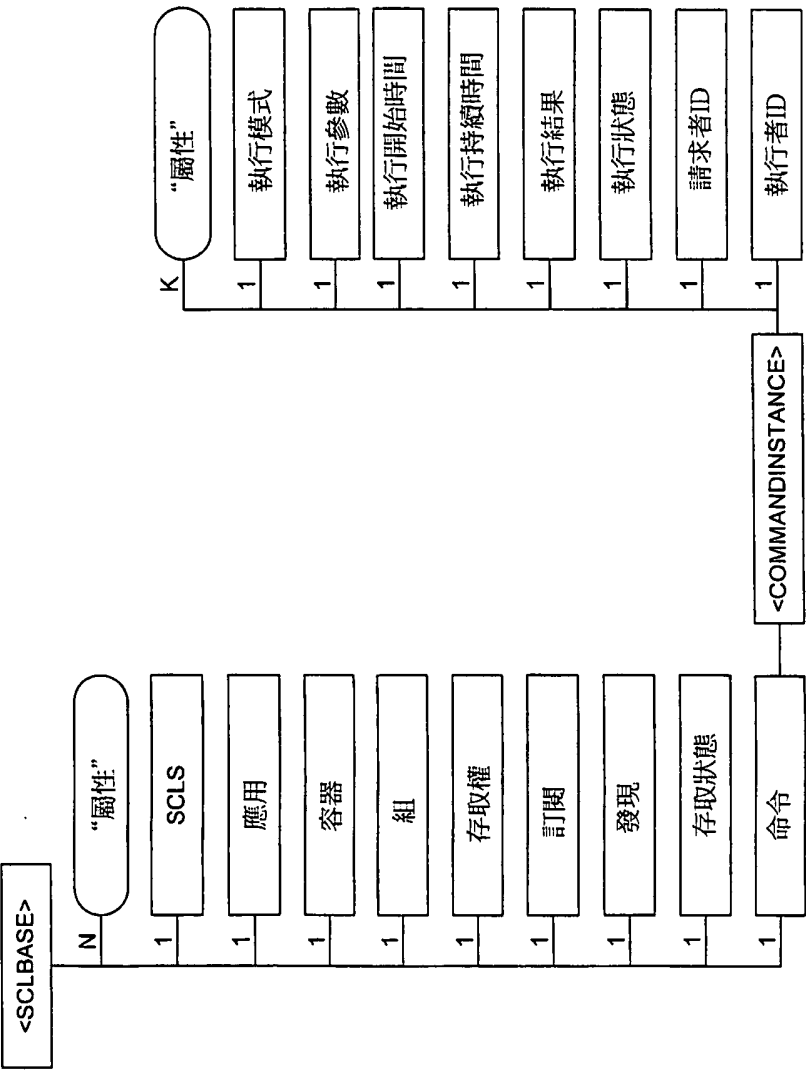
第19圖



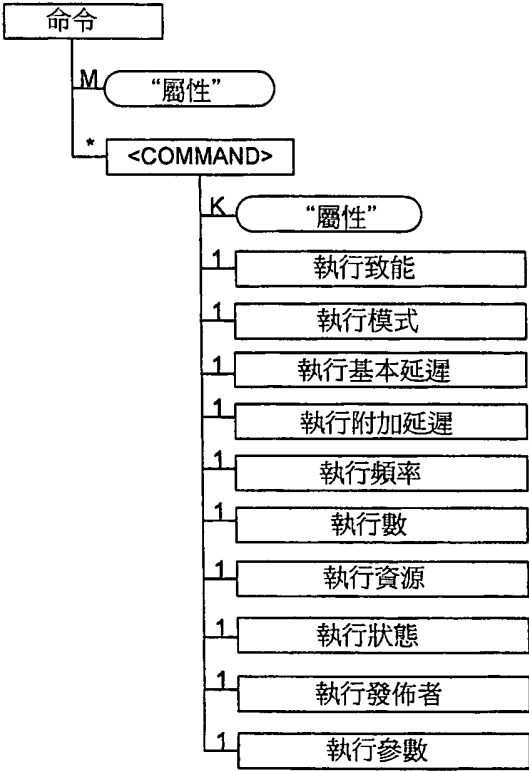
第20圖



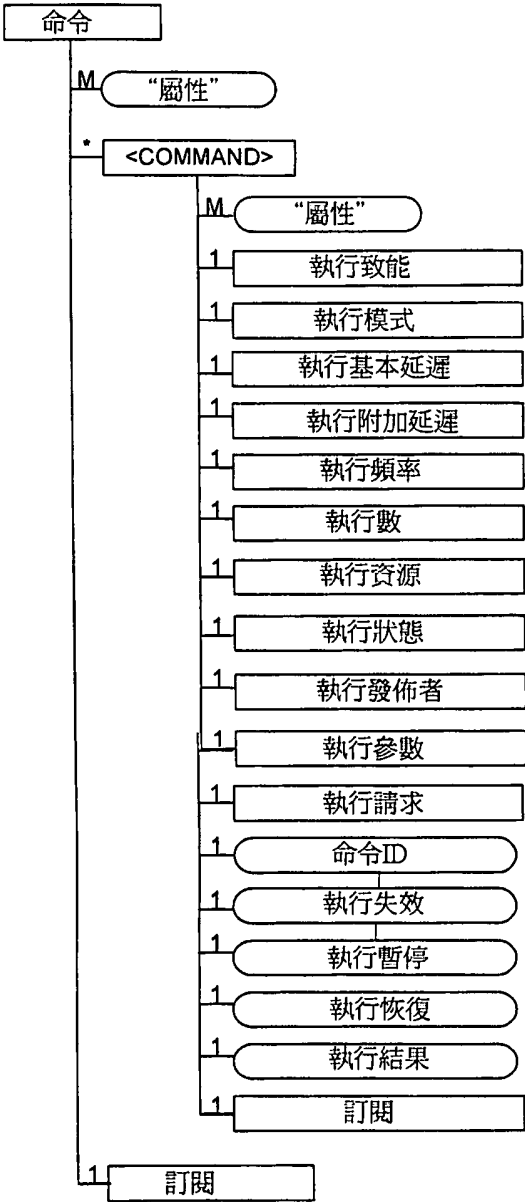
第21圖



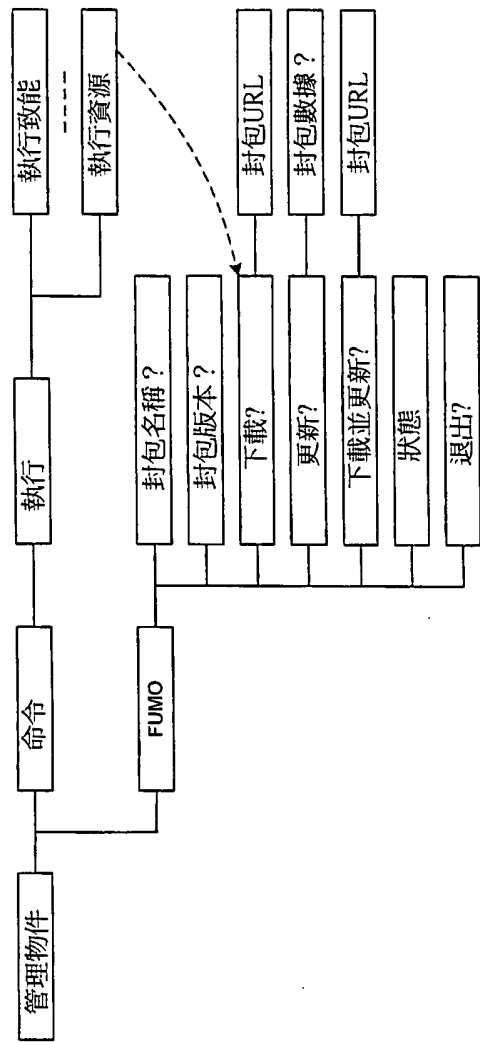
第22A圖



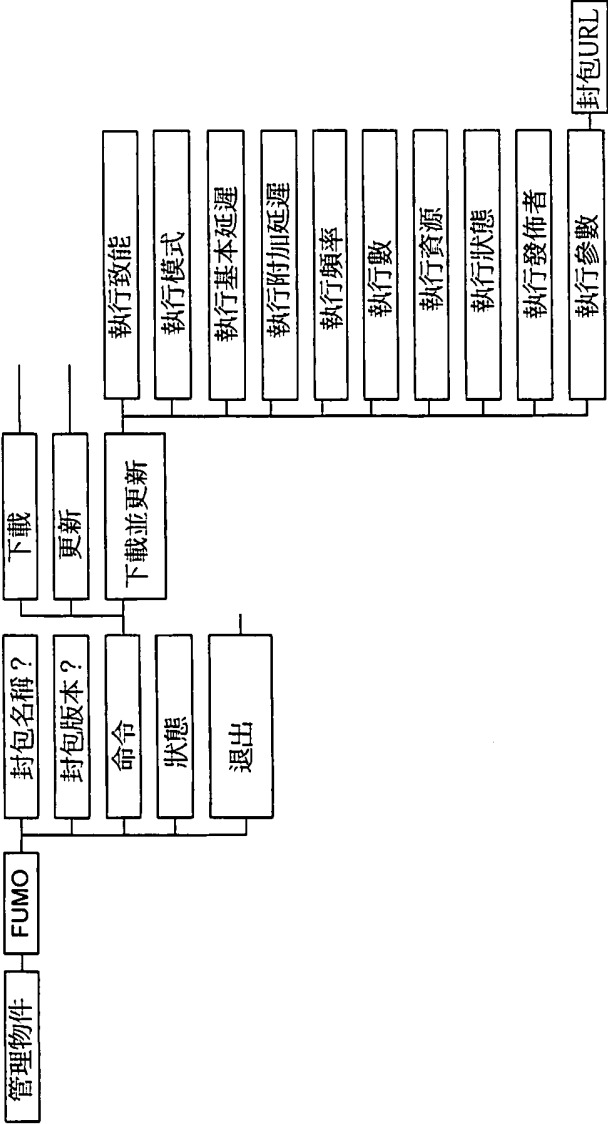
第22B圖



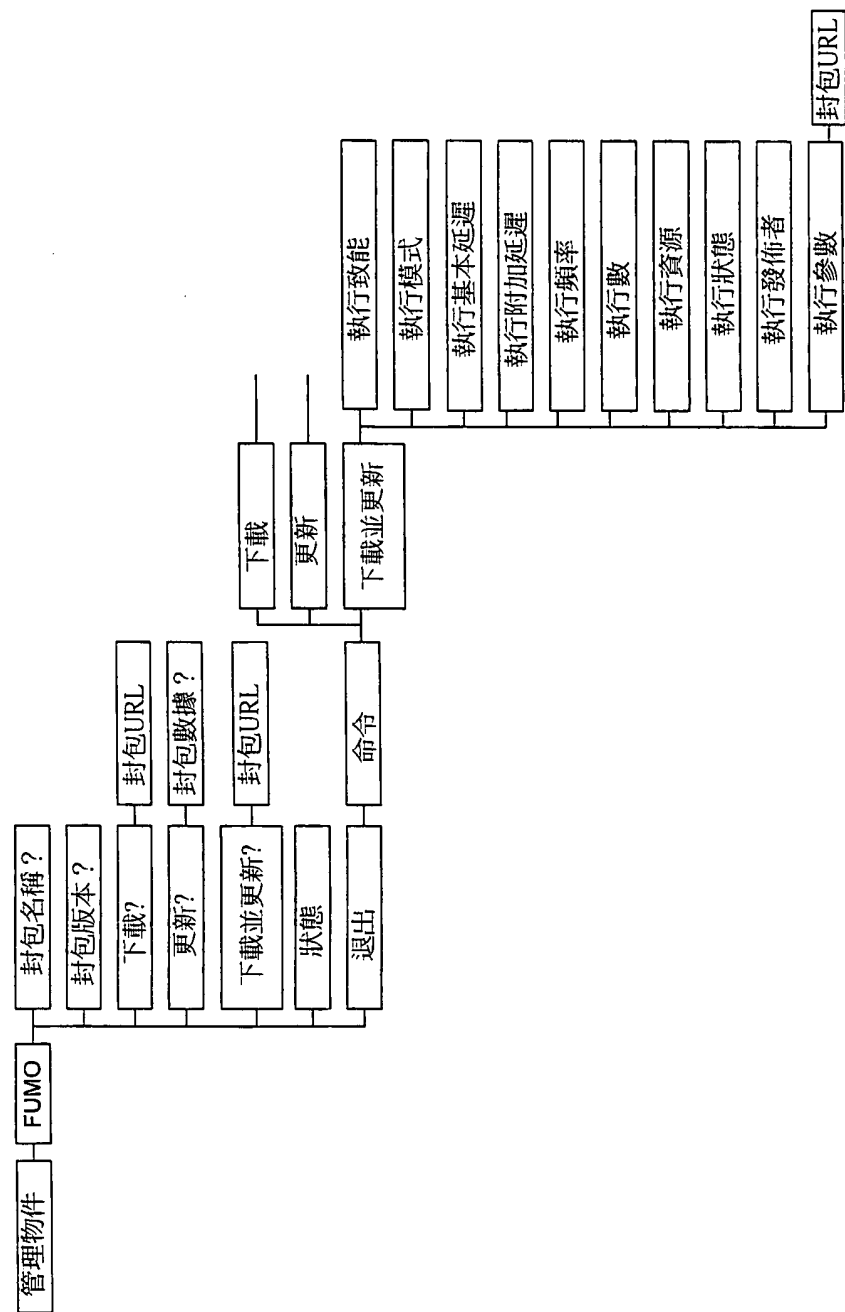
第22C圖



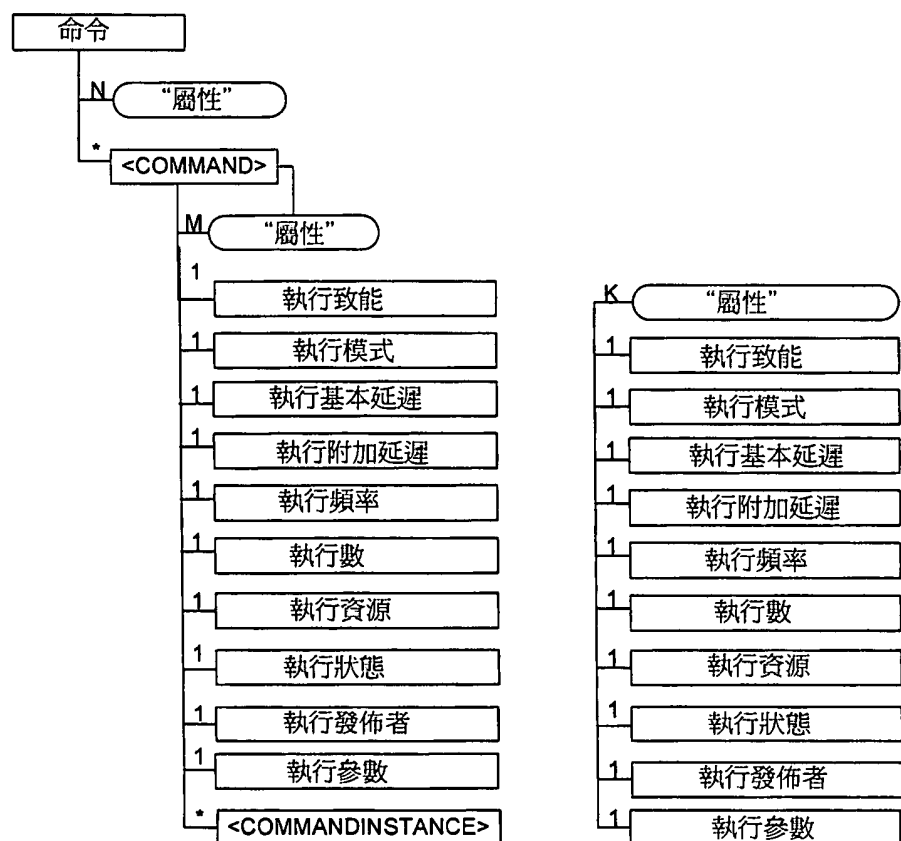
第23圖



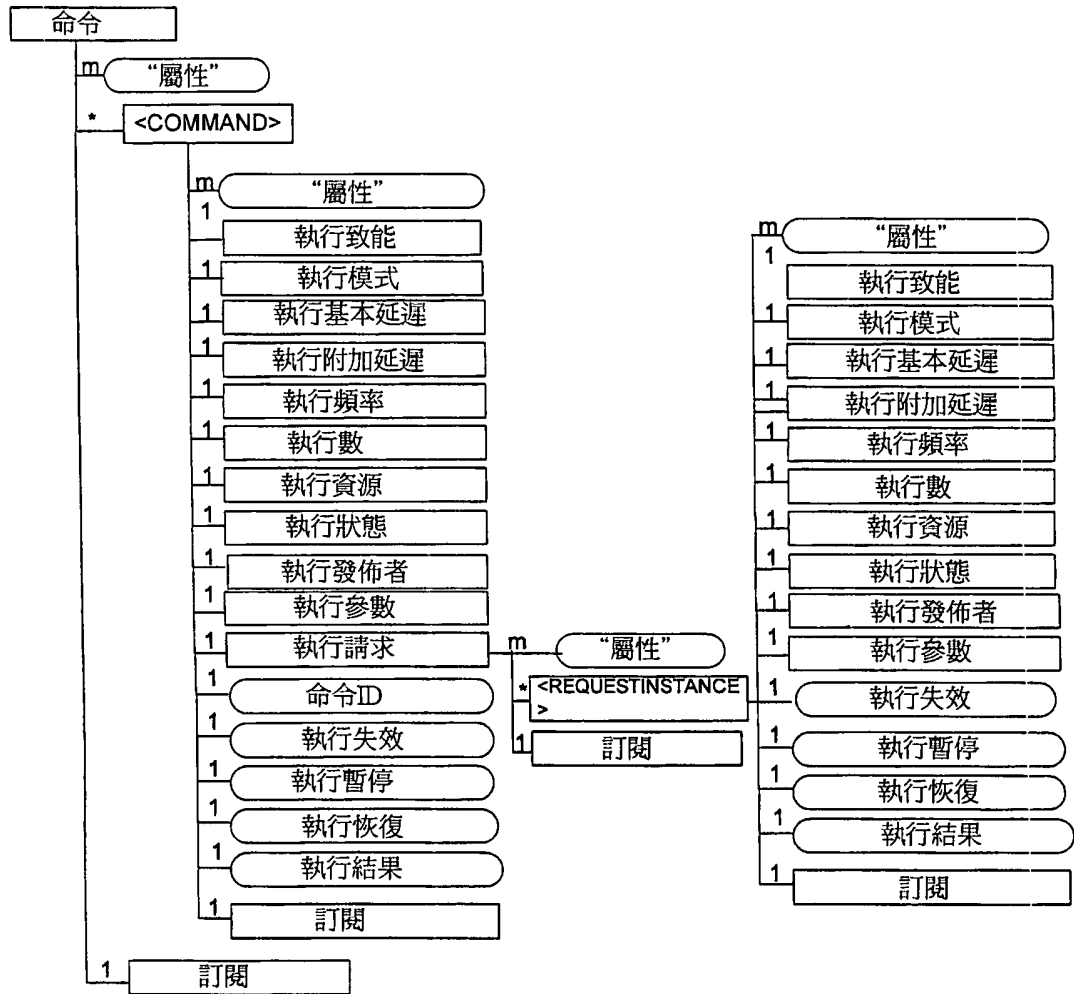
第24A圖



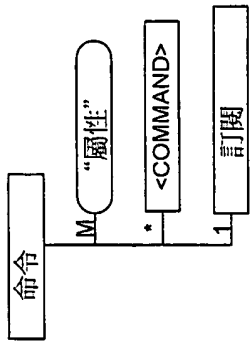
第24B圖



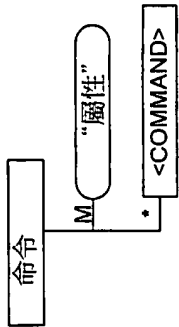
第25A圖



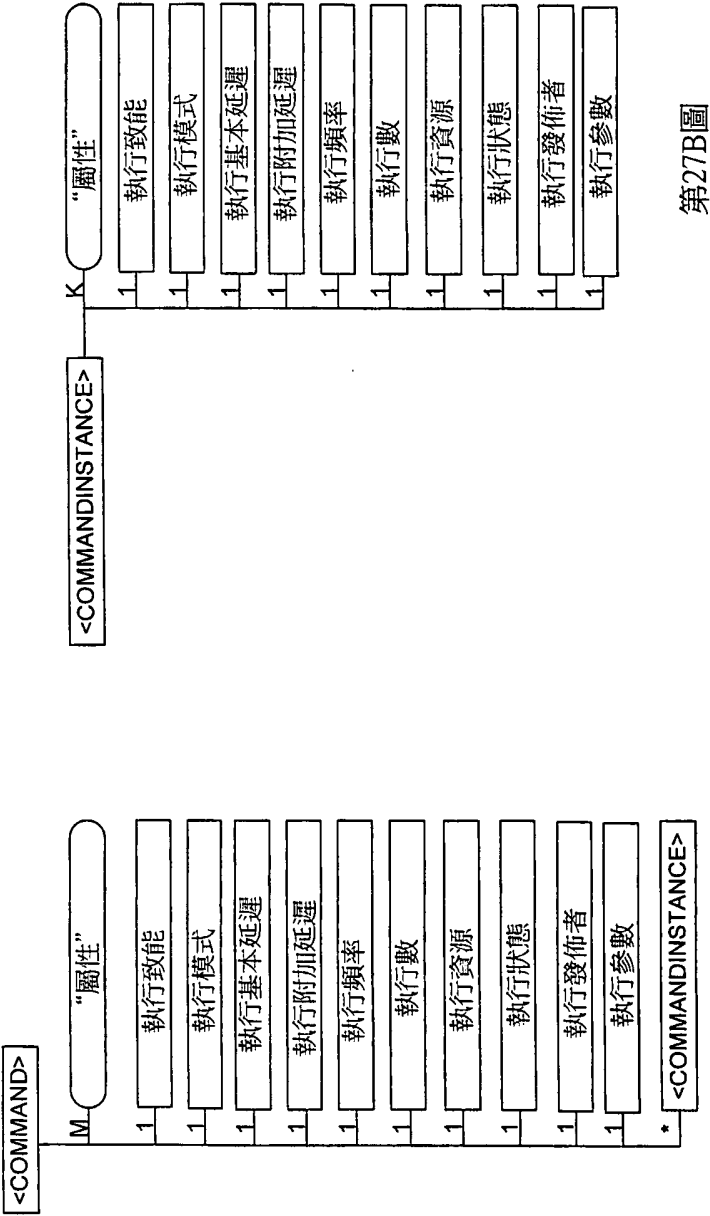
第25B圖



第26B圖

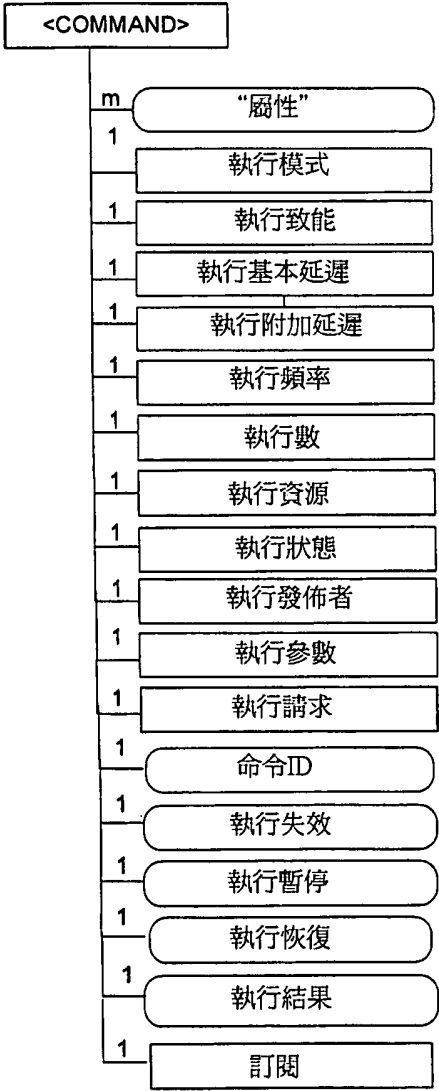


第26A圖

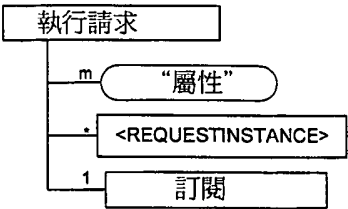


第27A圖

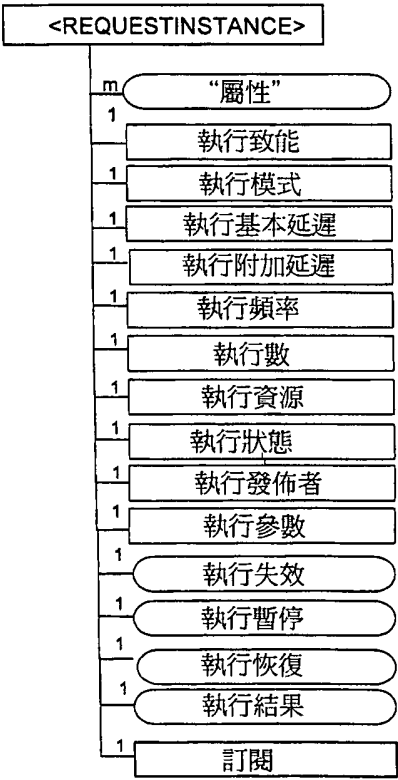
第27B圖



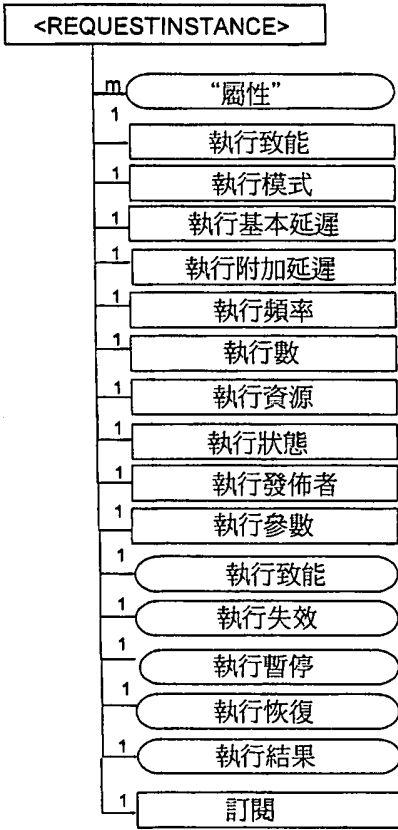
第28A圖



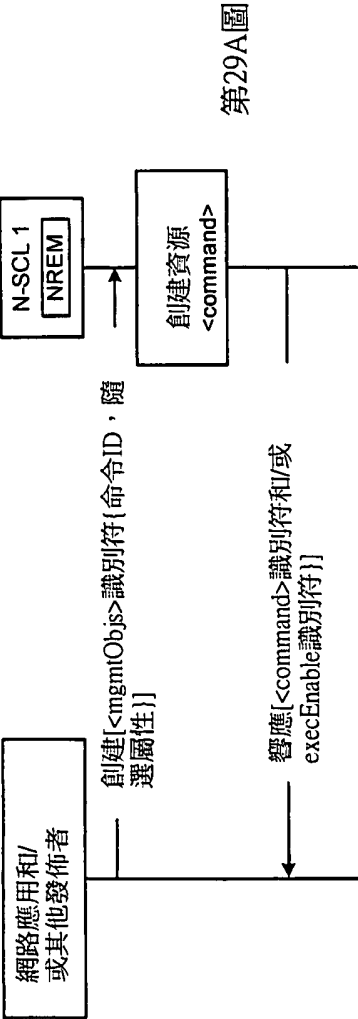
第28B圖



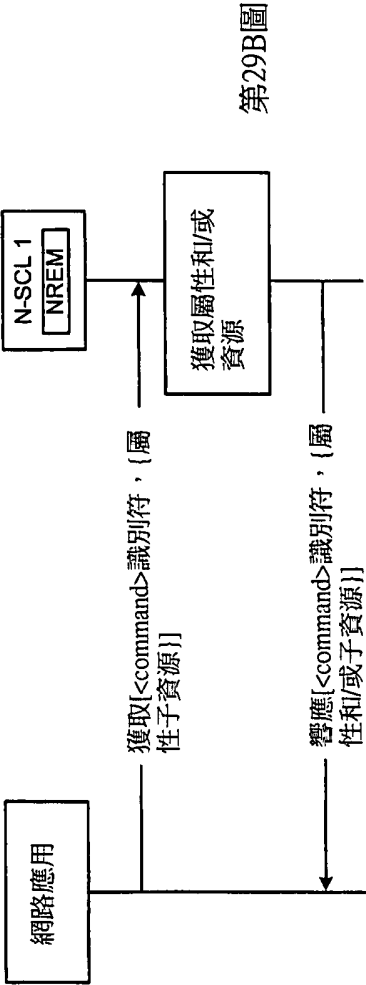
第28C圖



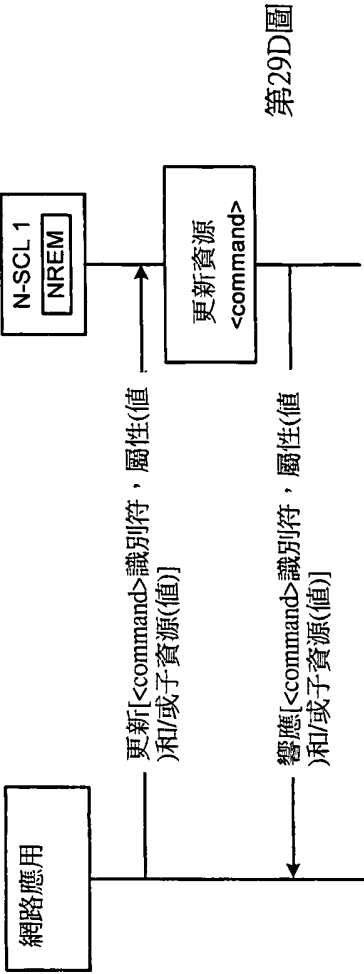
第28D圖



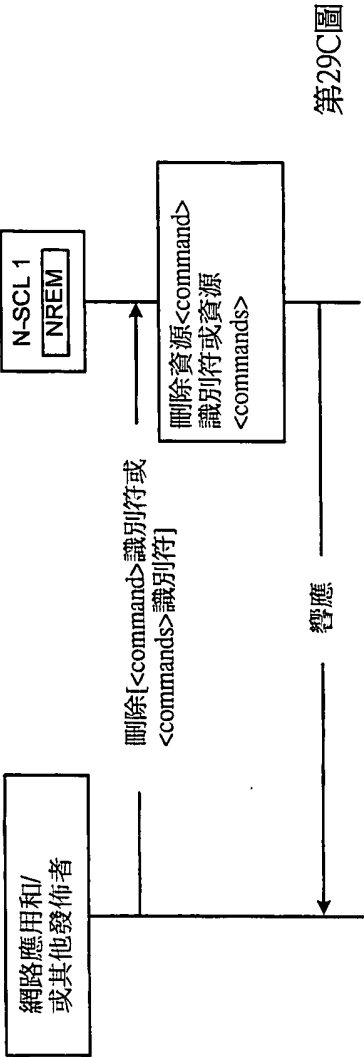
第29A圖



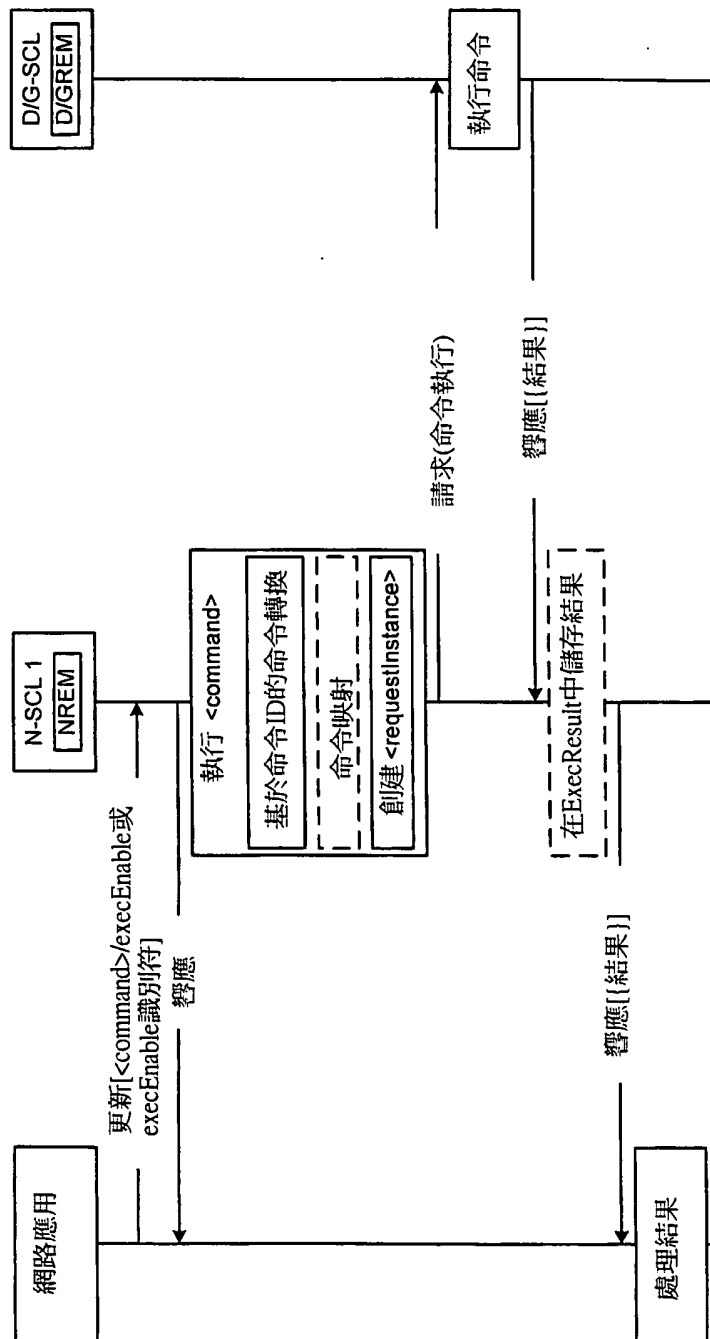
第29B圖



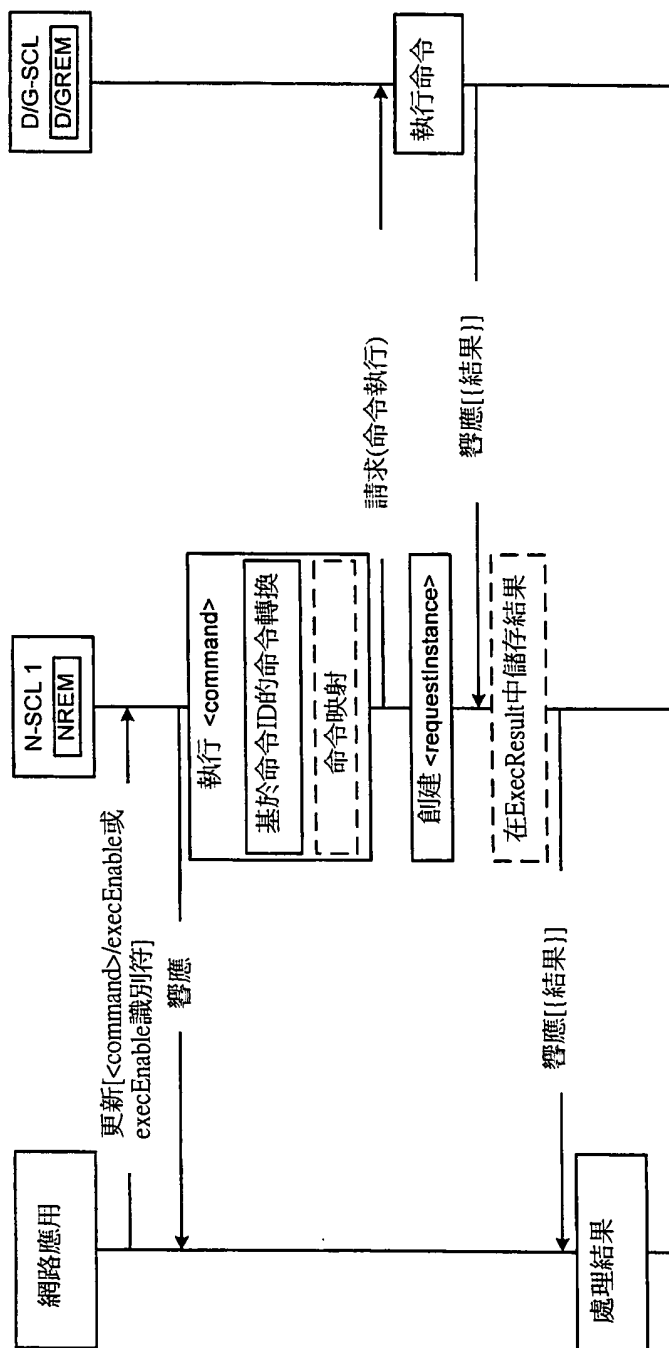
第29D圖



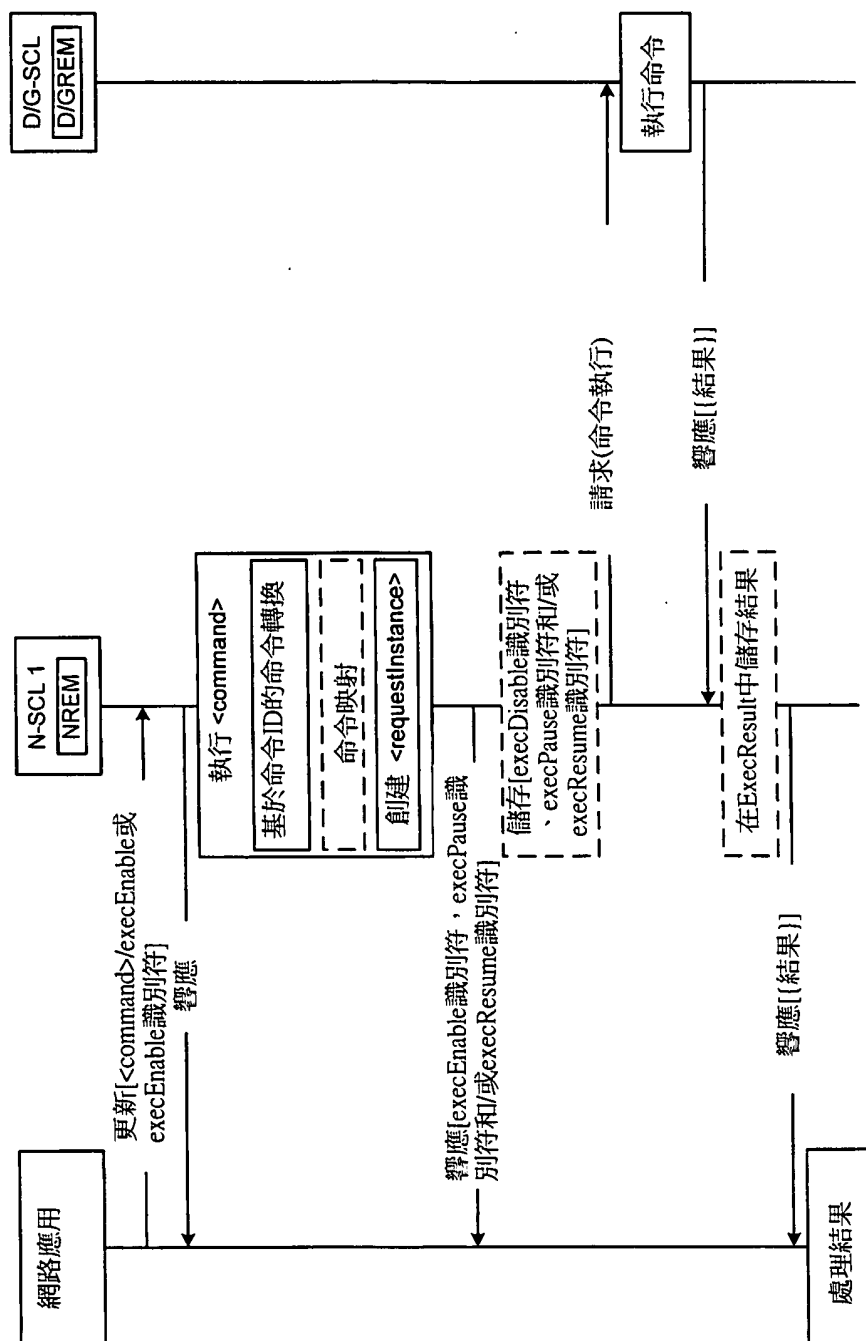
第29C圖



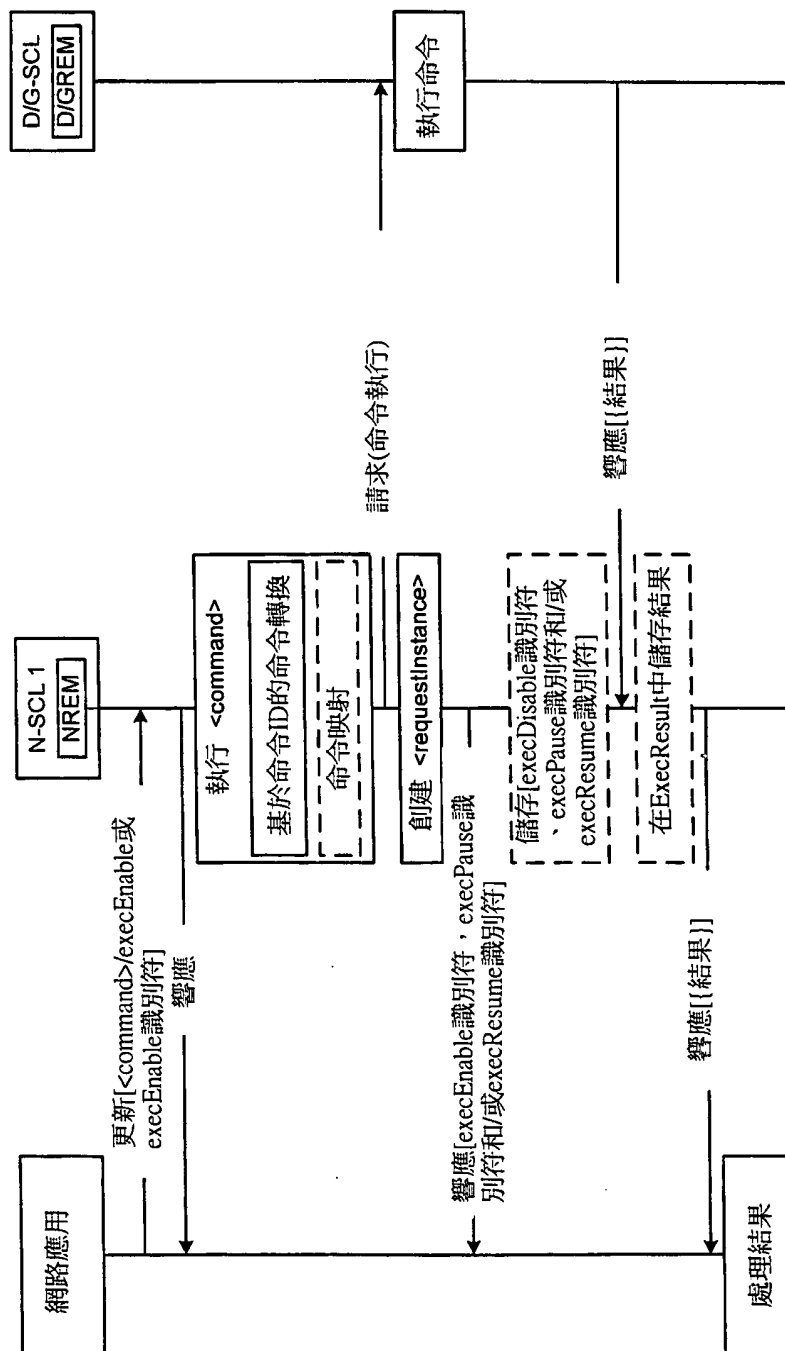
第29E圖



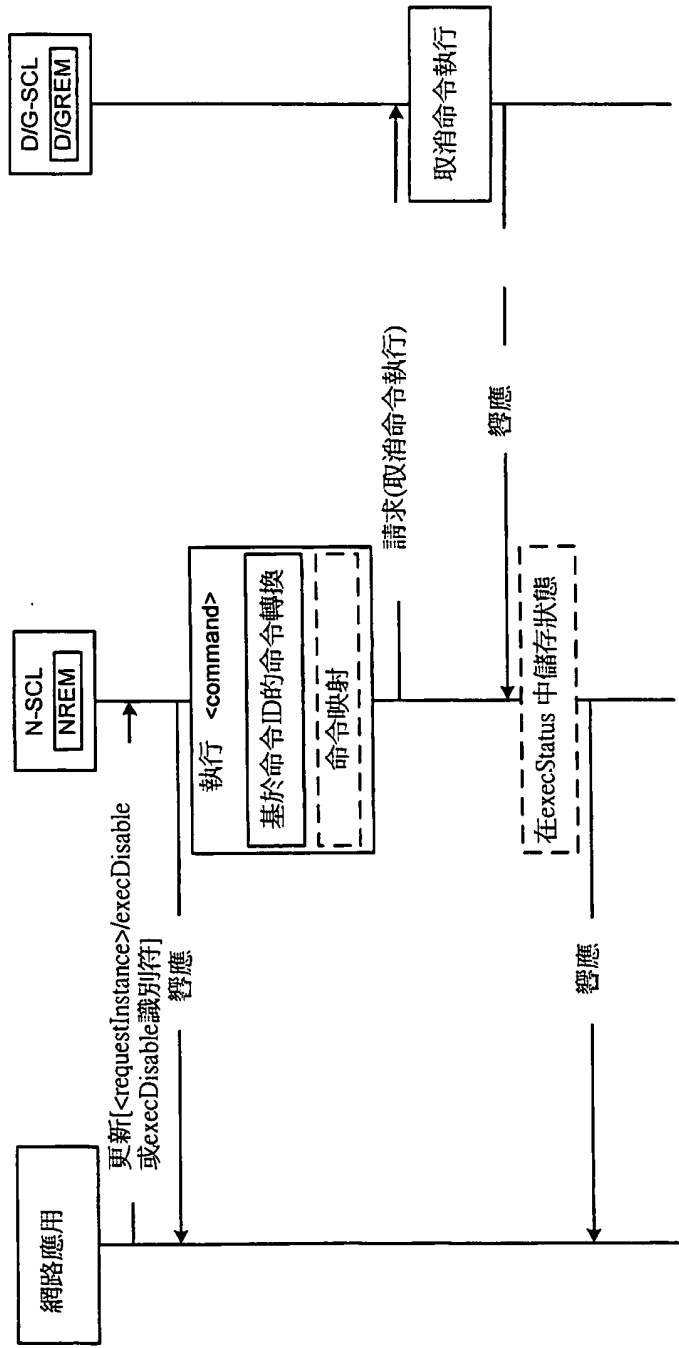
第29F圖



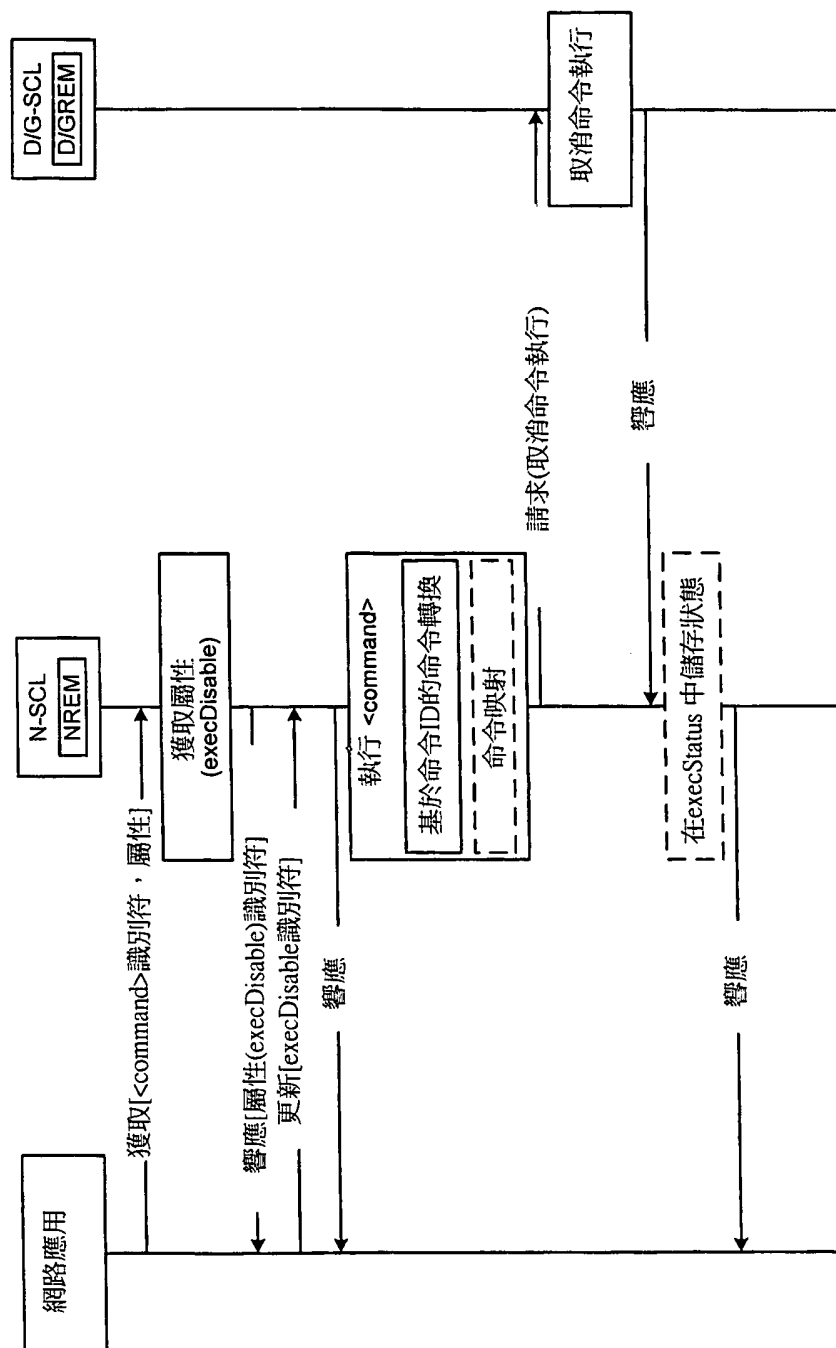
第29G圖



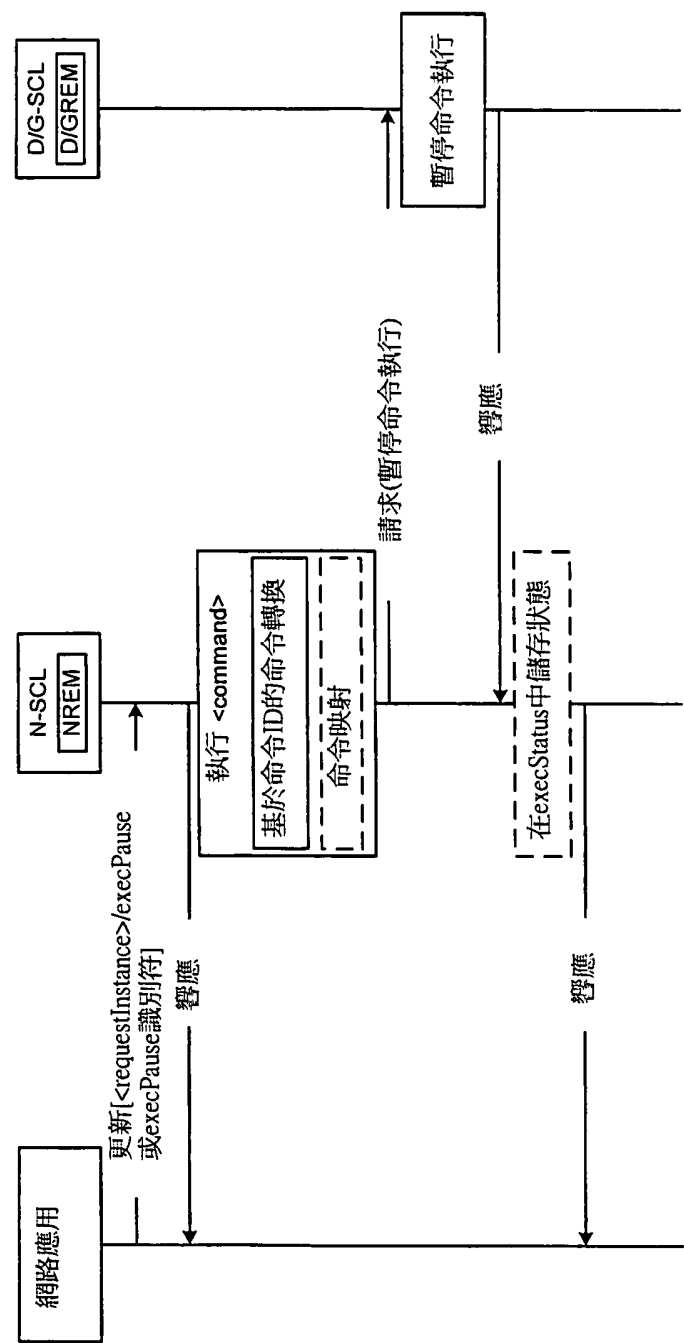
第29H圖



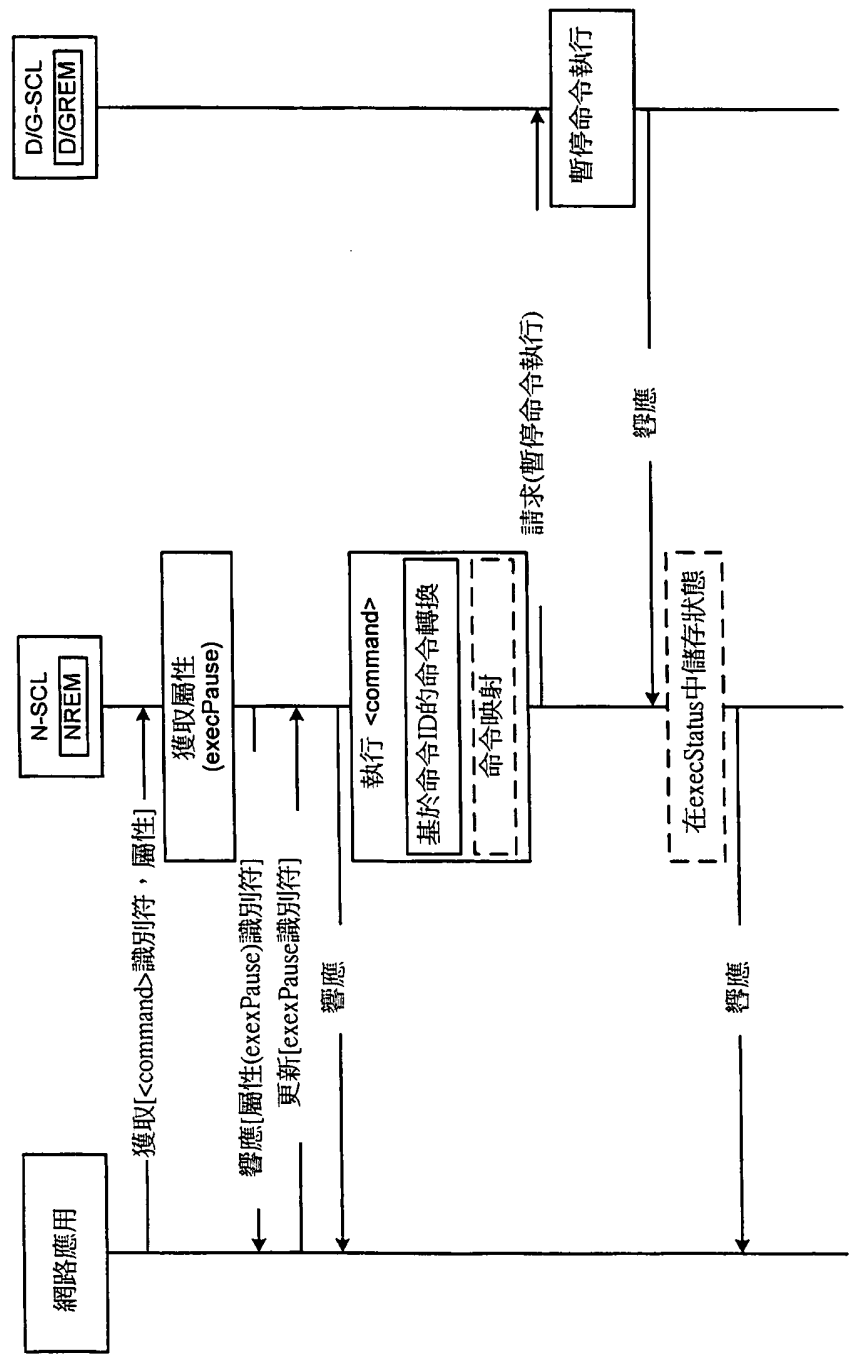
第29圖



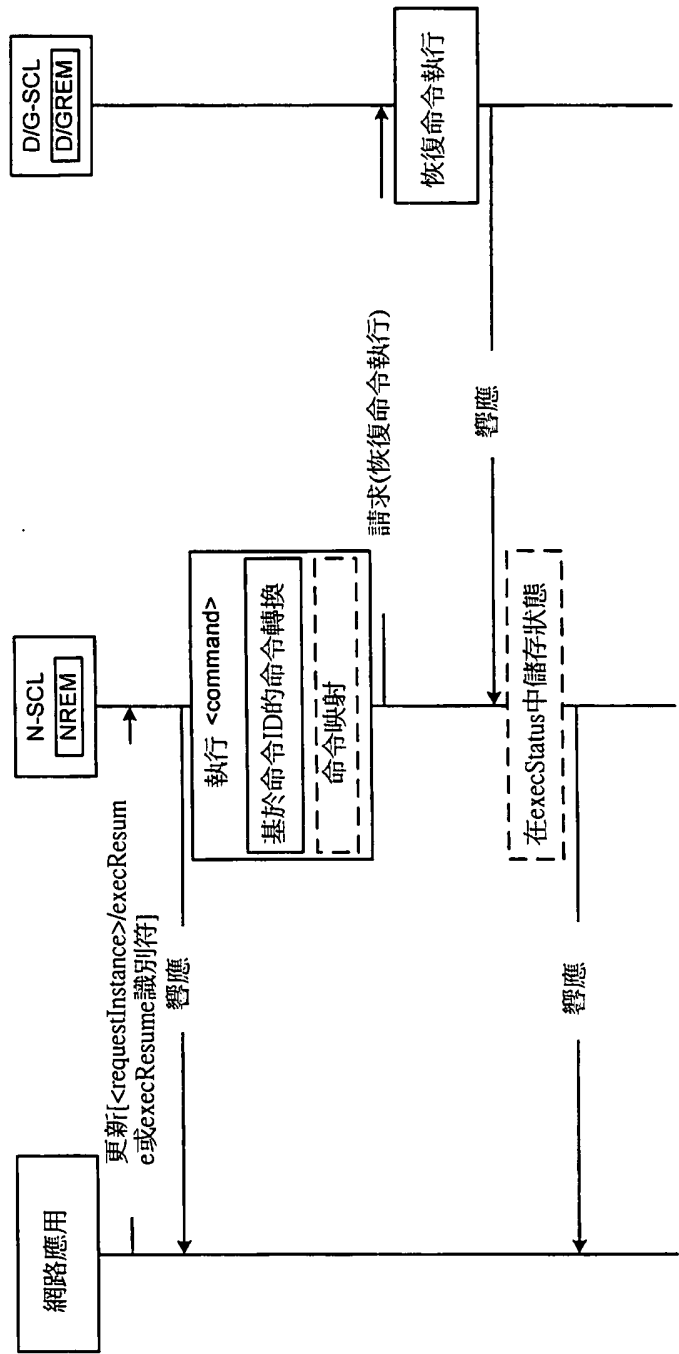
第29圖



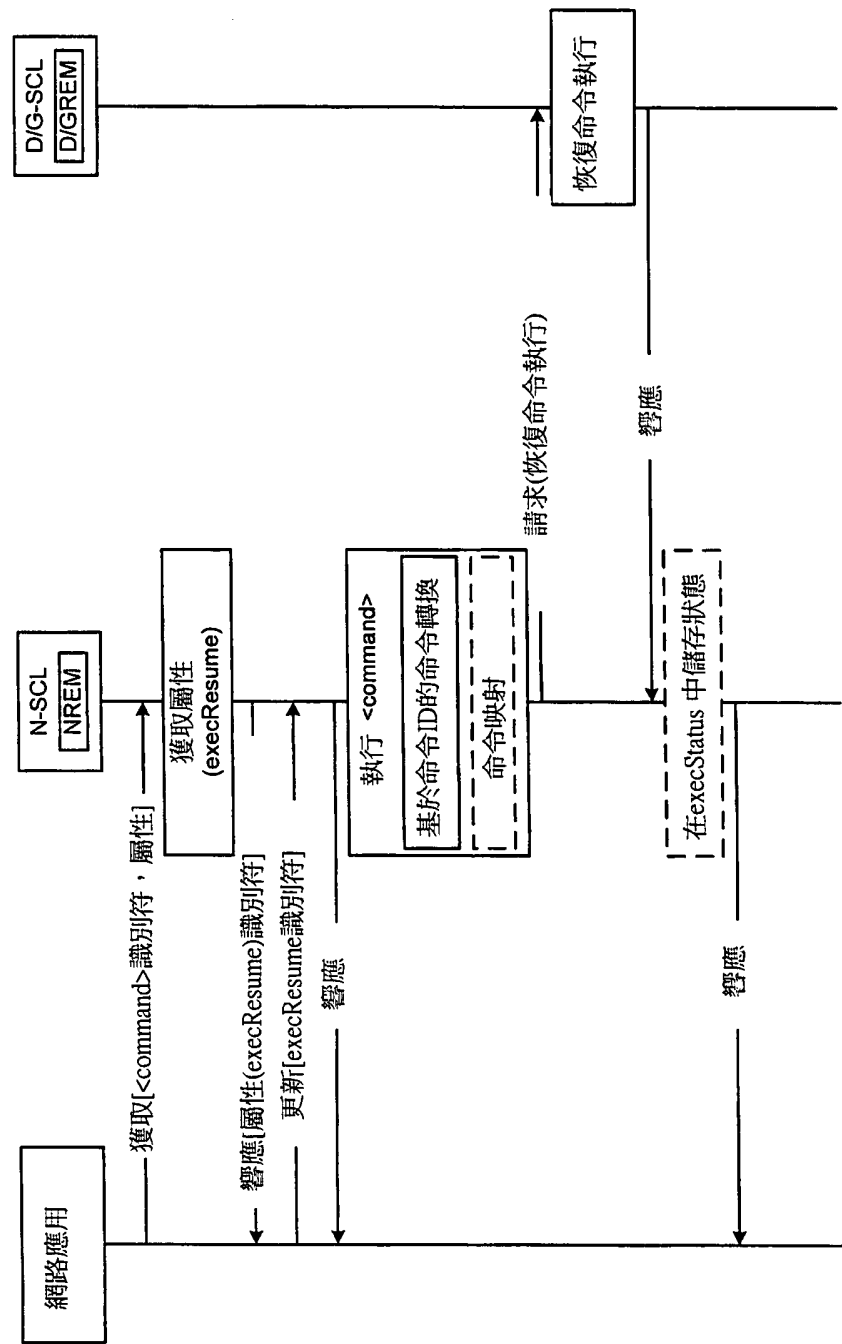
第29K圖



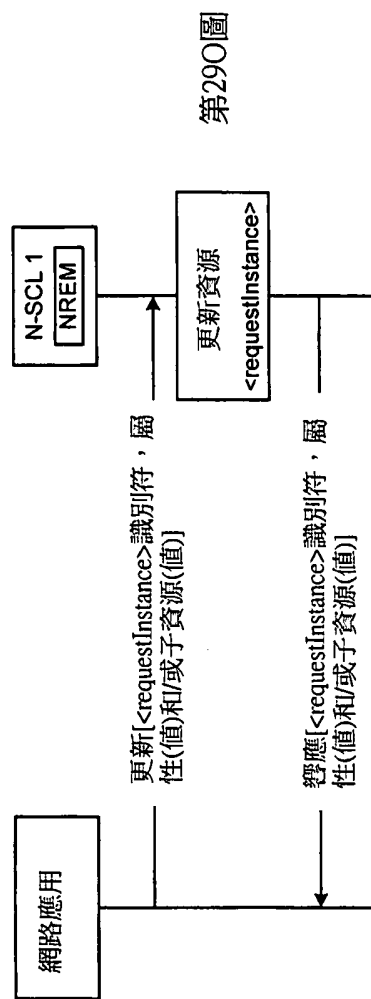
第29L圖



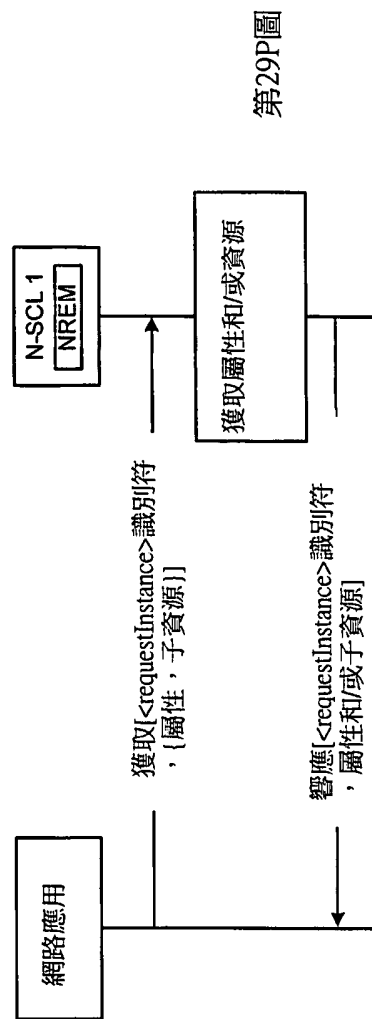
第29M圖



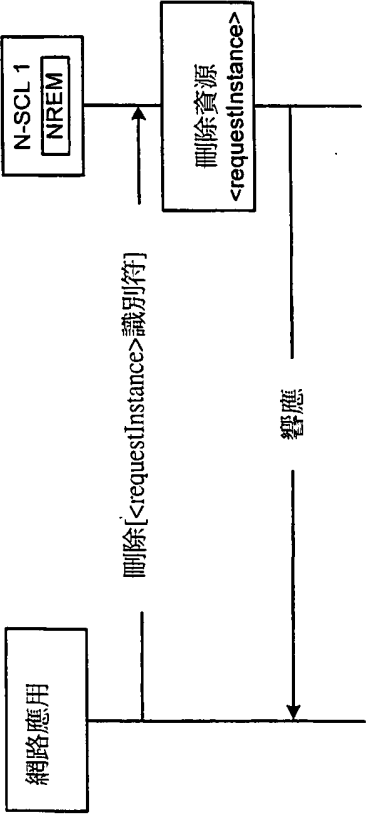
第29N圖



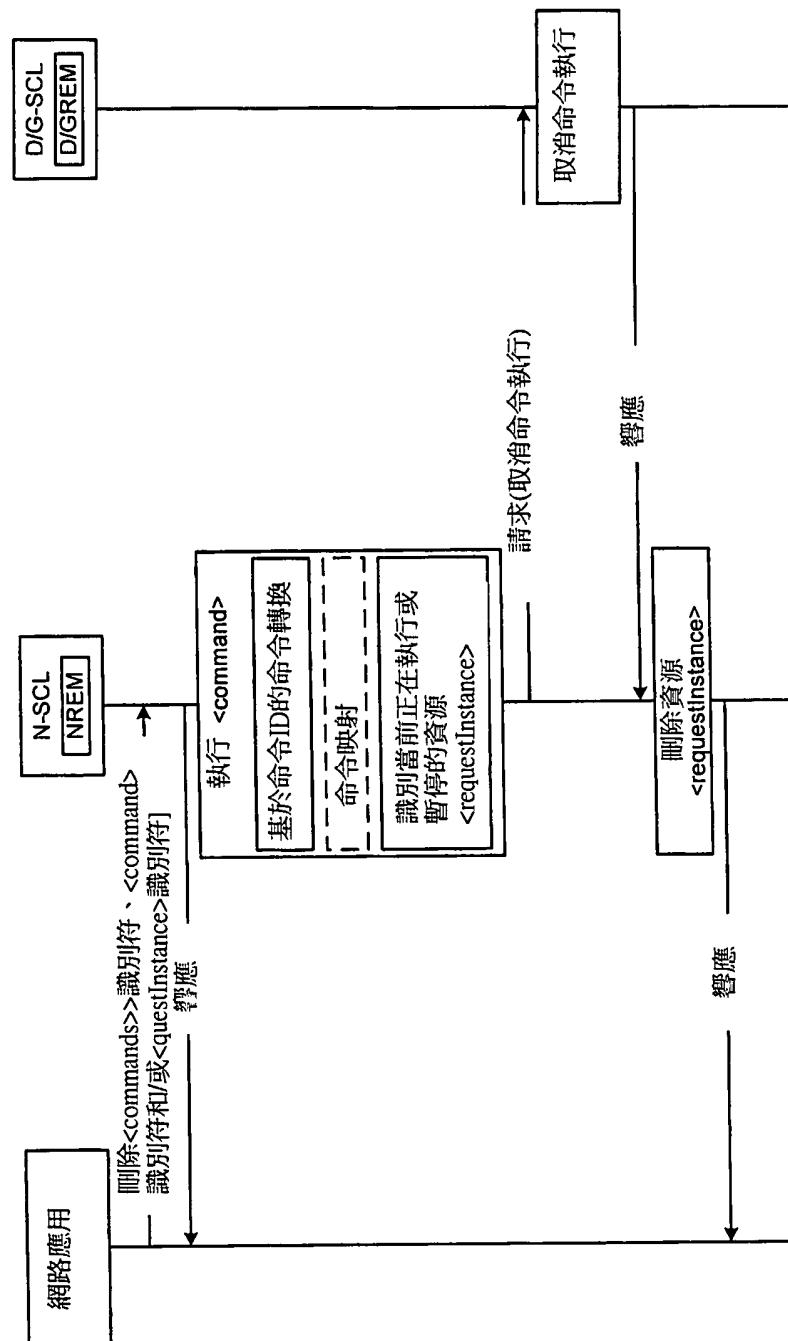
第290圖



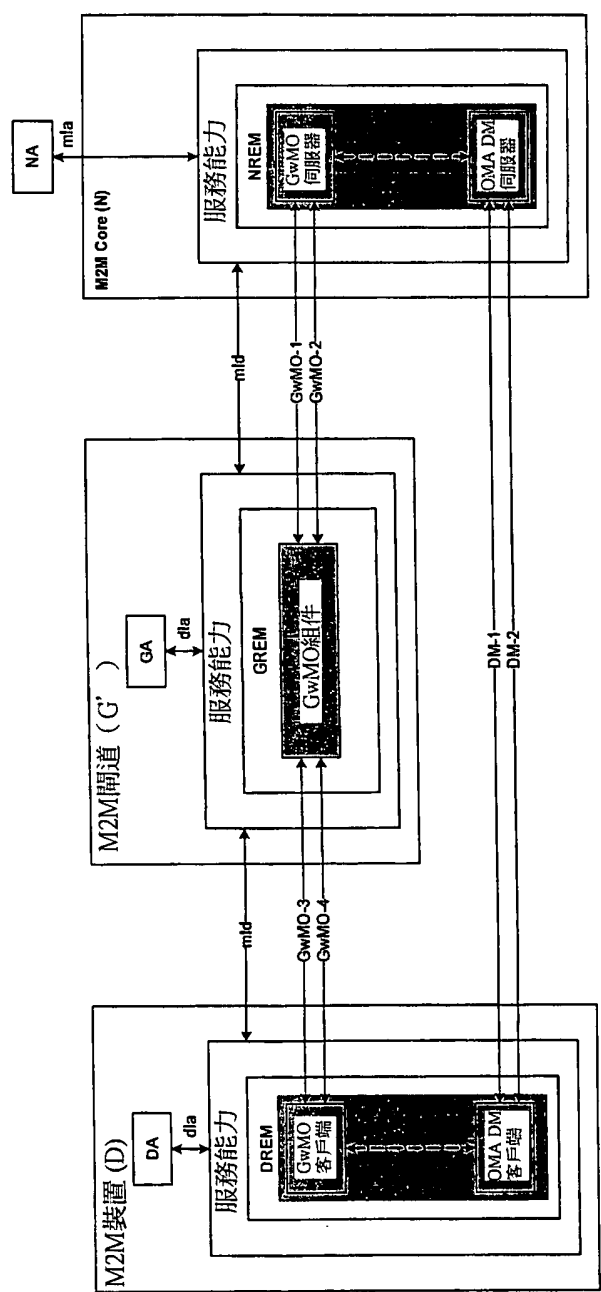
第29P圖



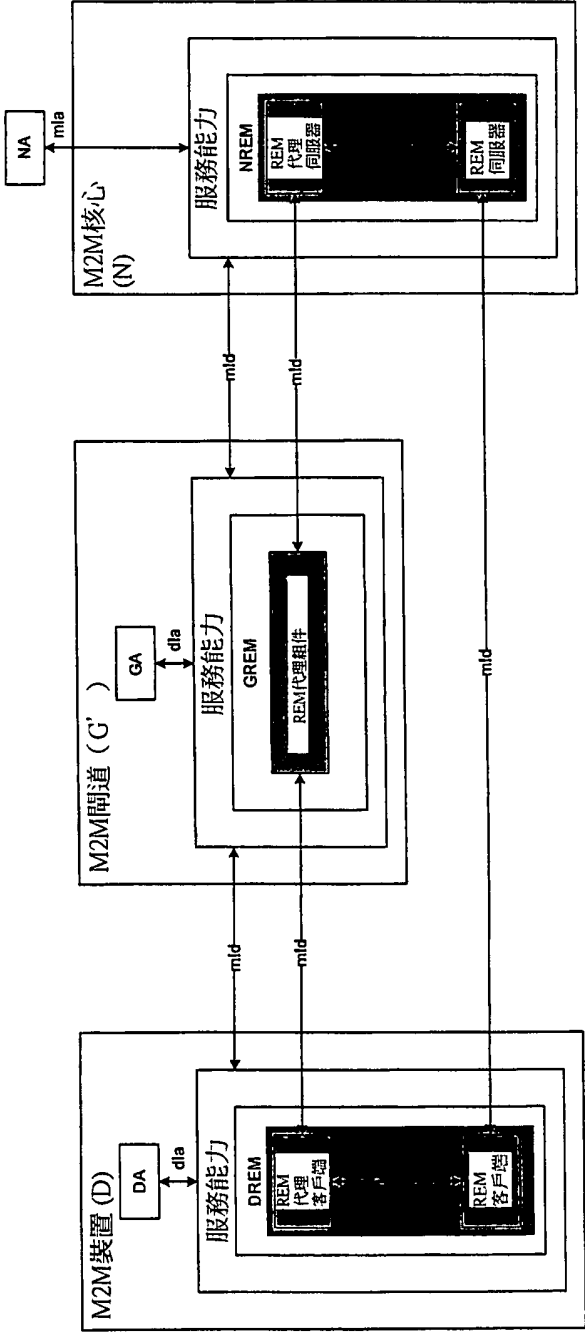
第29Q圖



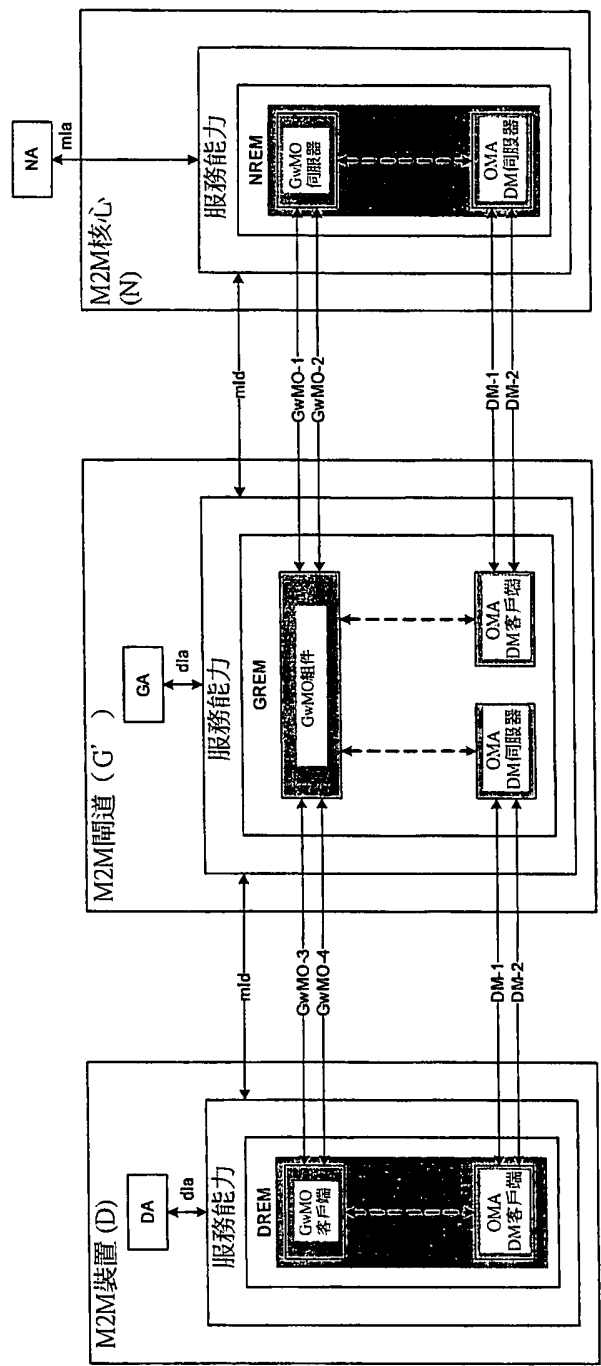
第29R圖



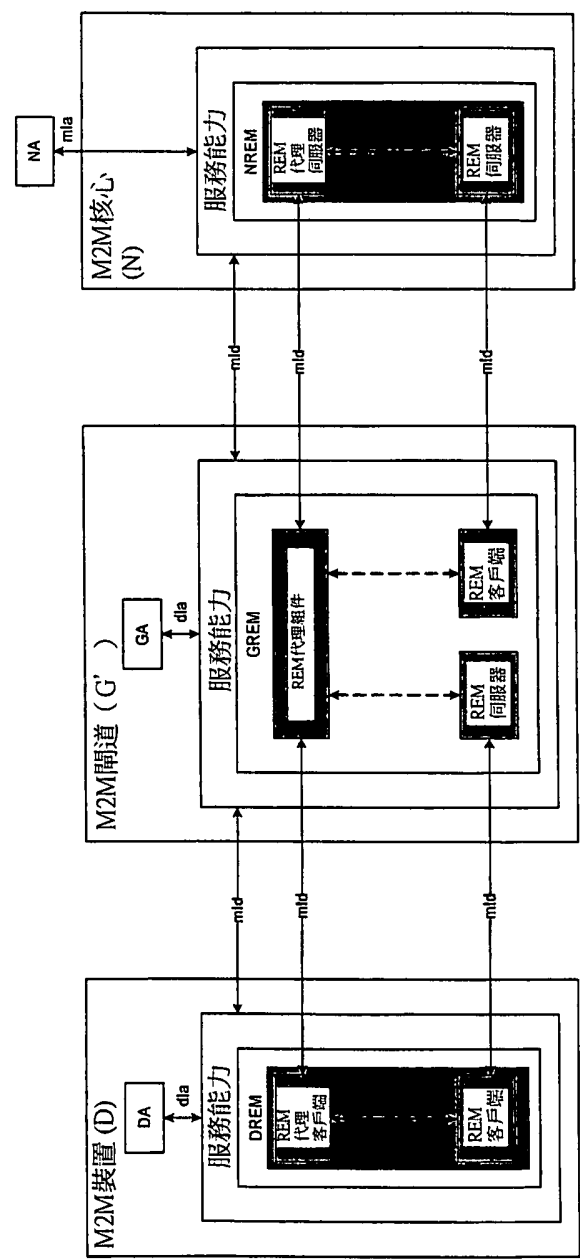
第30A圖



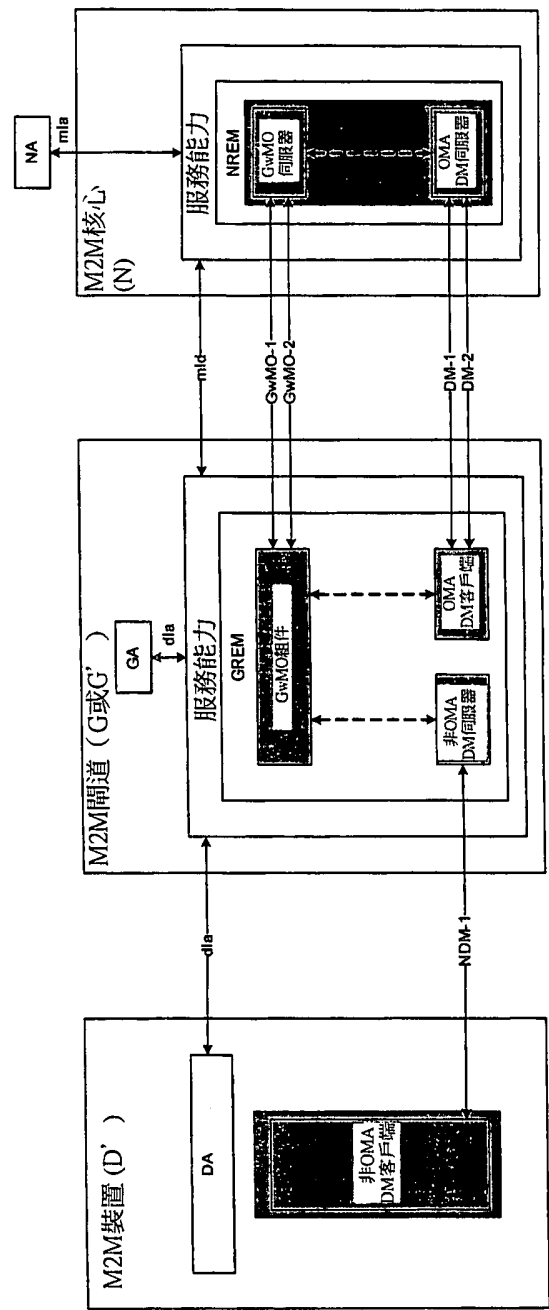
第30B圖



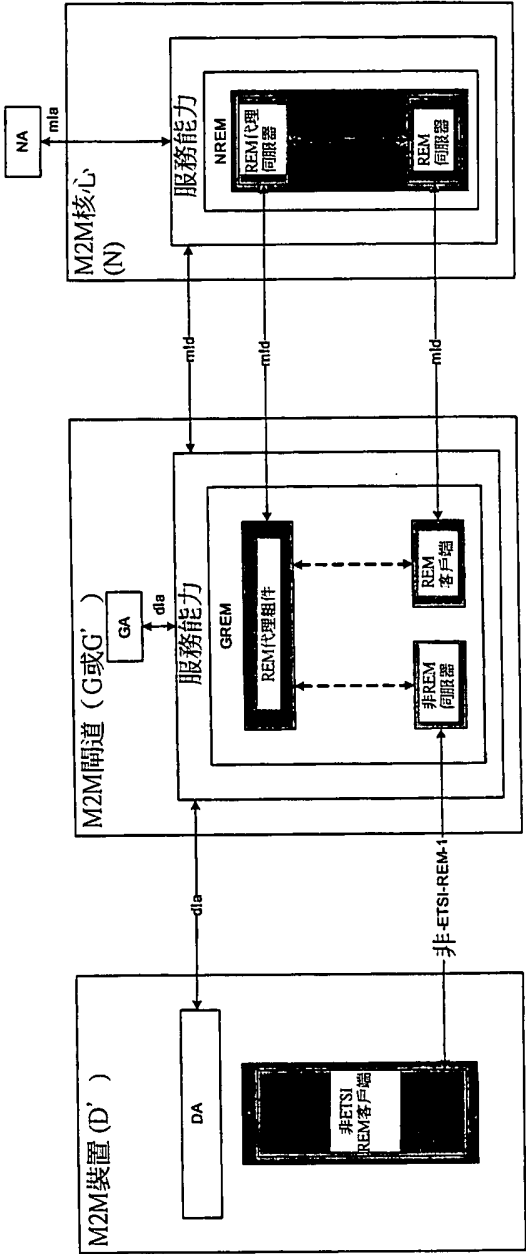
第31A圖



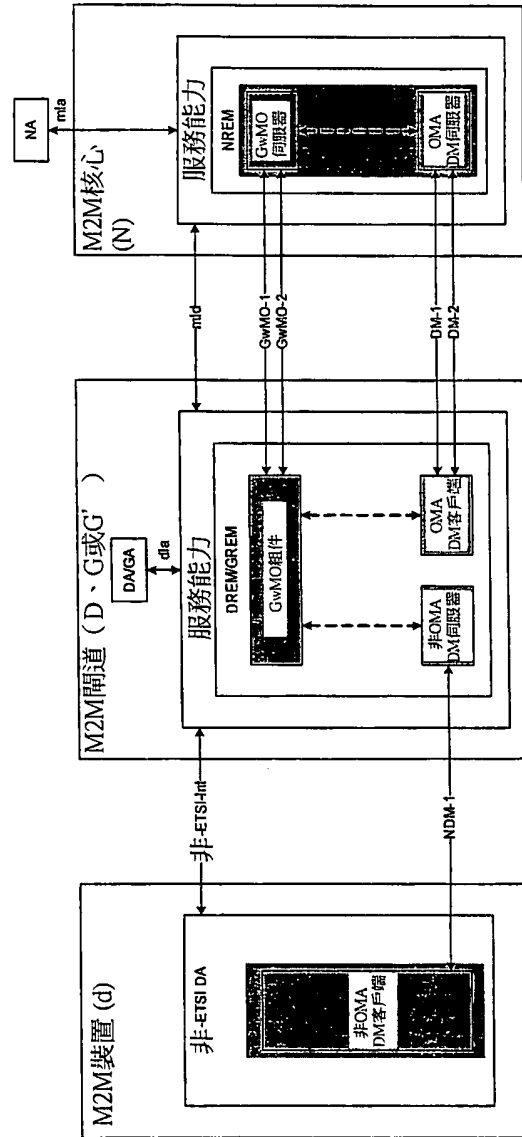
第31B圖



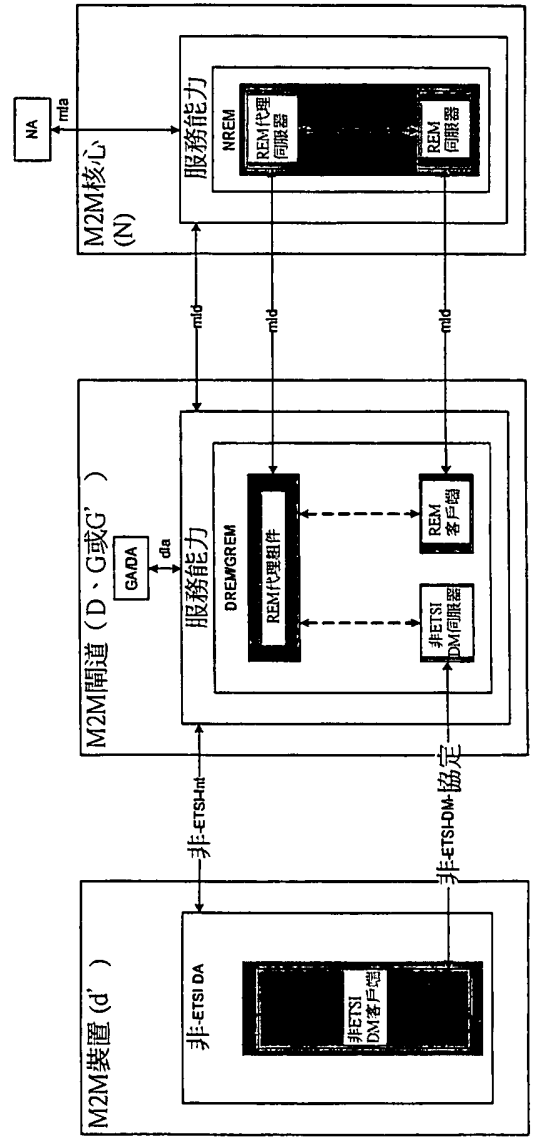
第32A圖



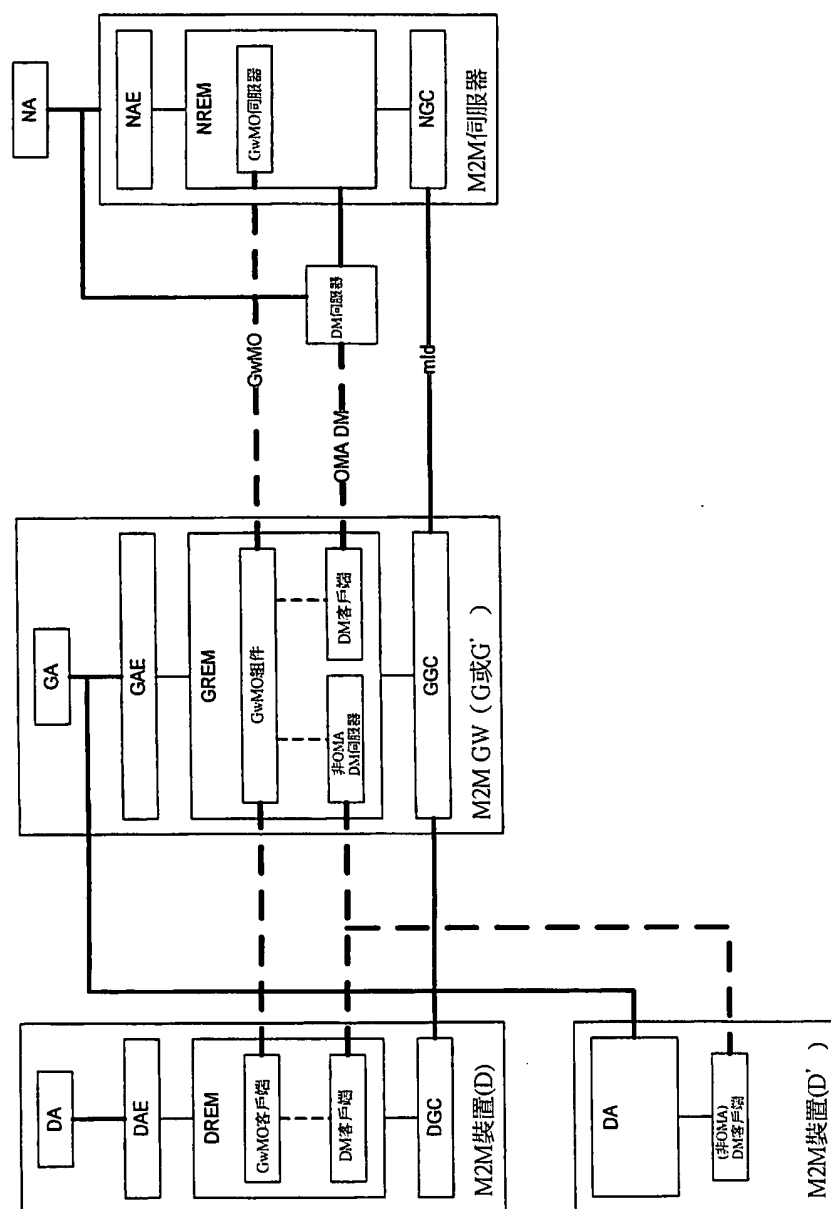
第32B圖

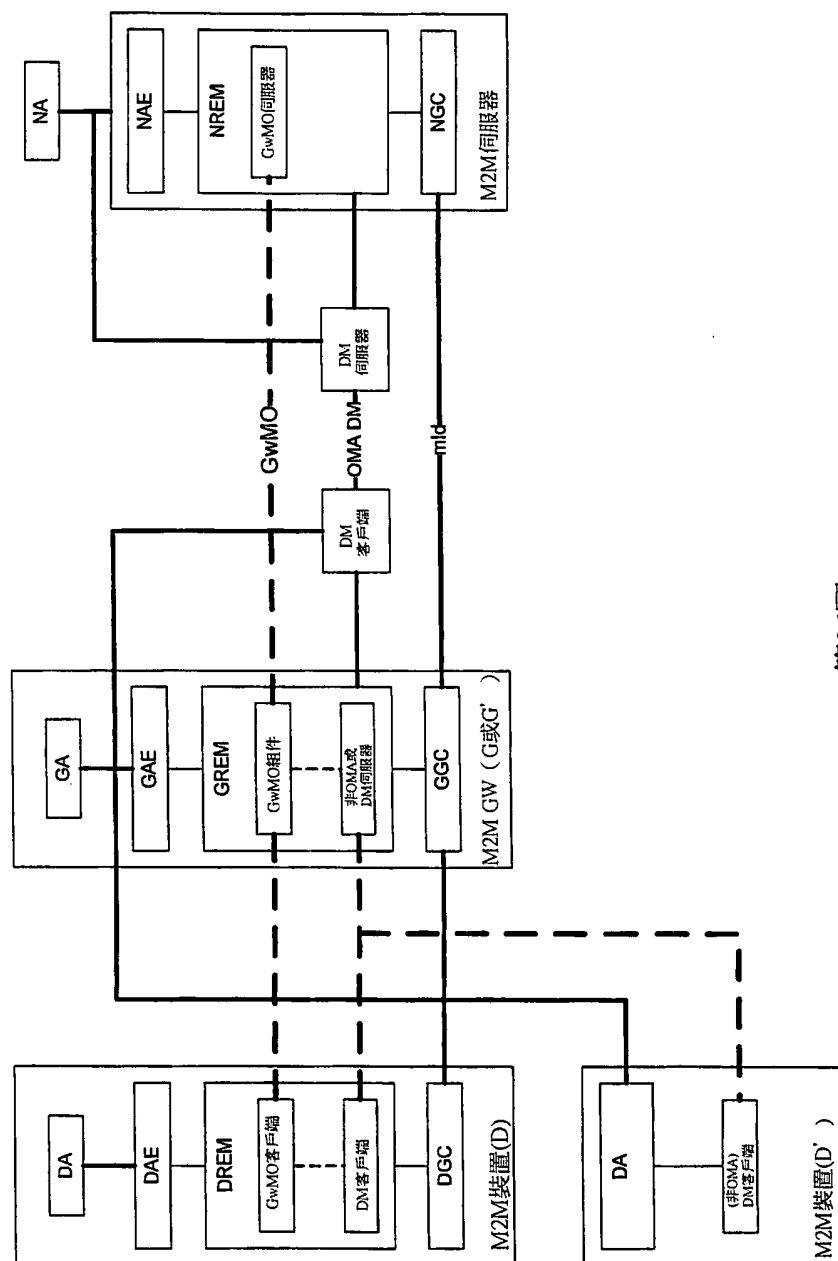


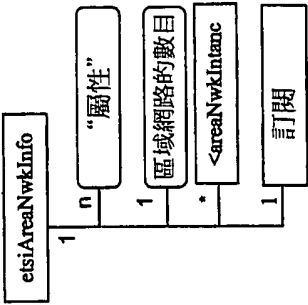
第33A圖



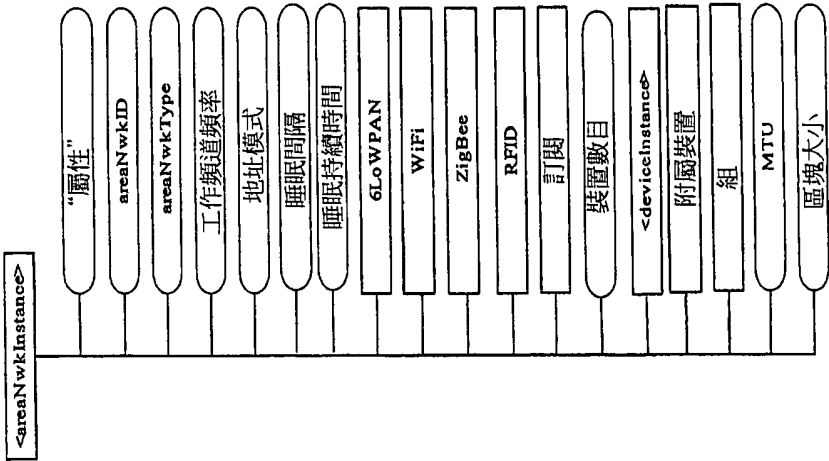
第33B圖



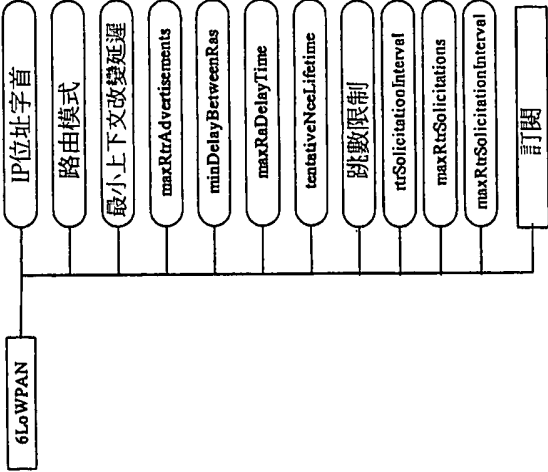




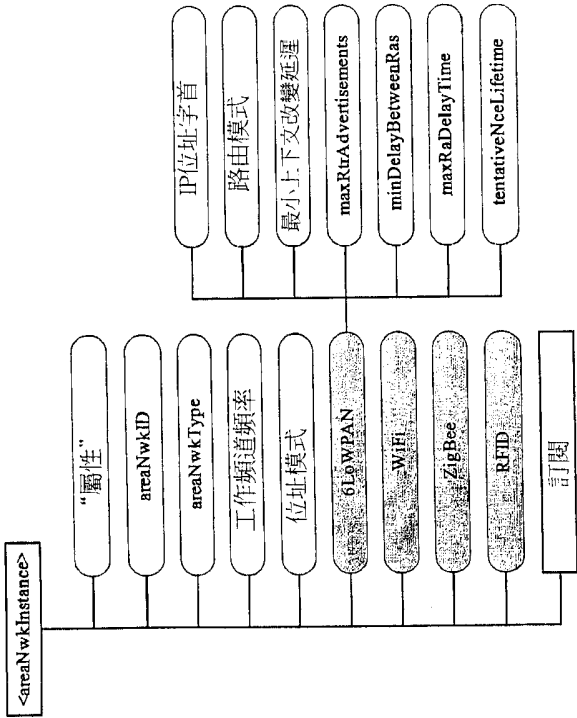
第37圖



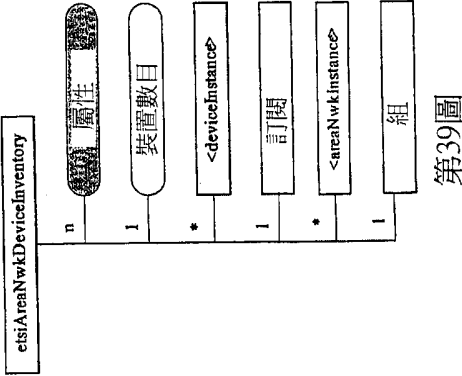
第38A圖



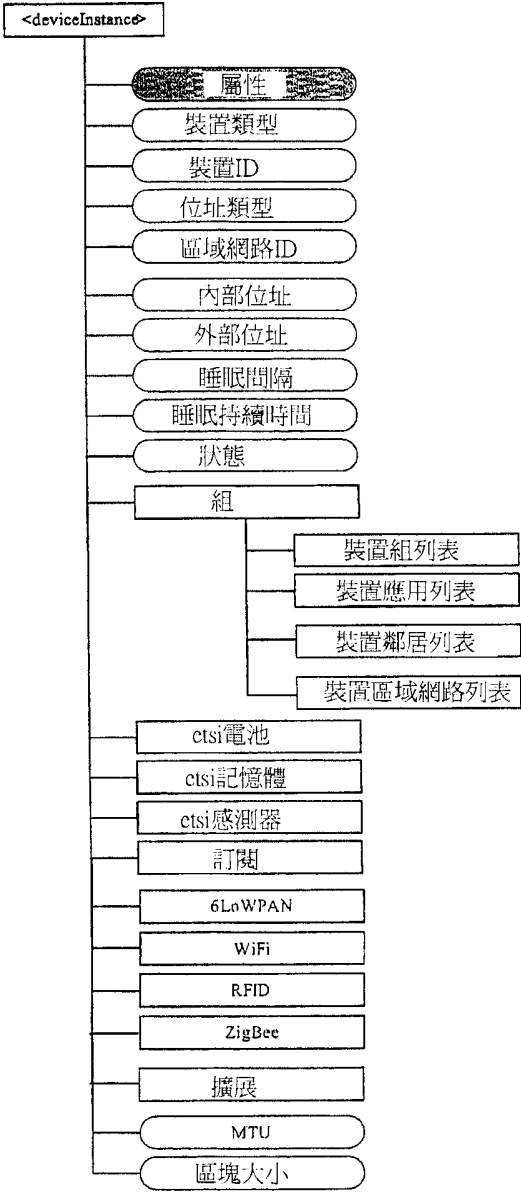
第38B圖



第38C圖



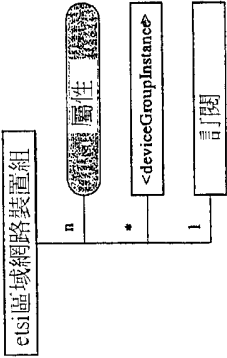
第39圖



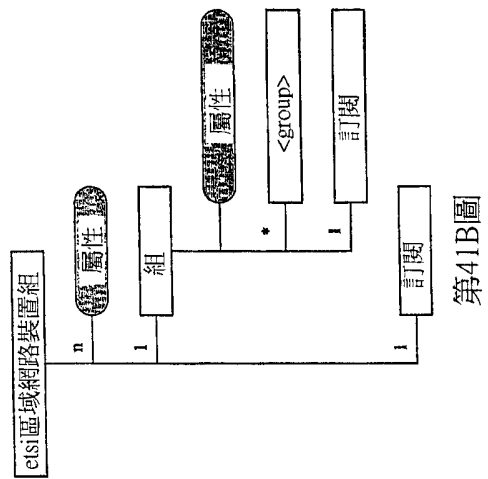
第40A圖

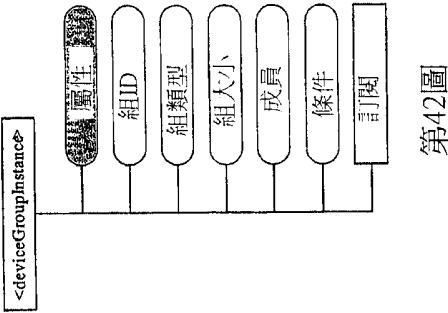


第40B圖

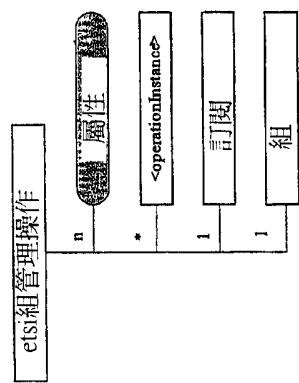


第41A圖

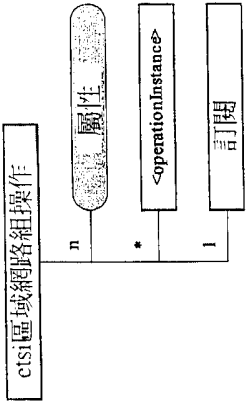




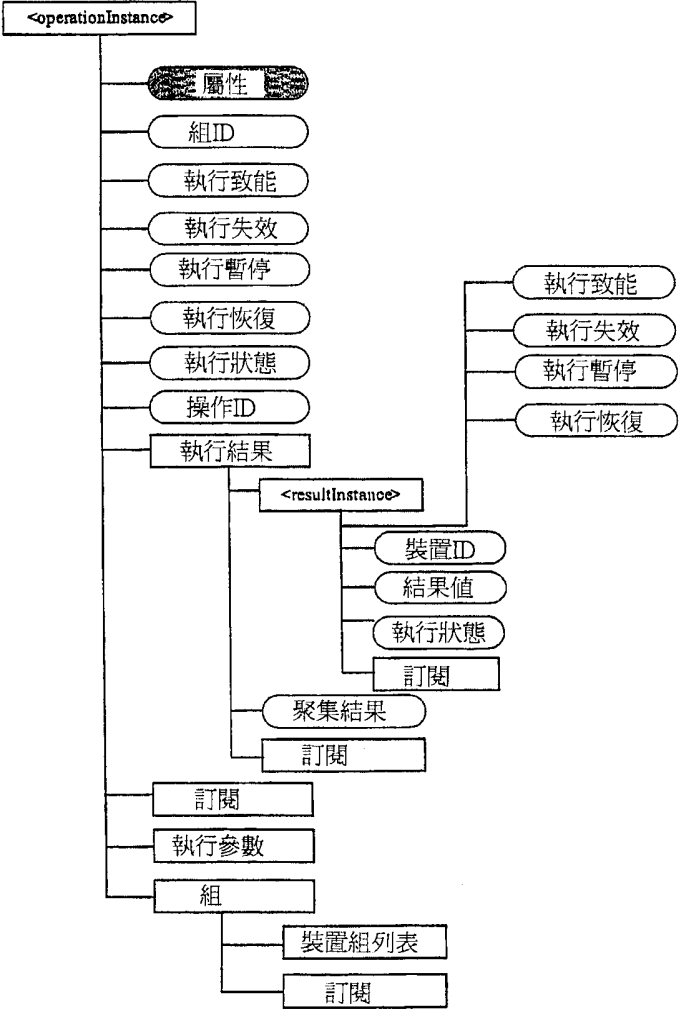
第42圖



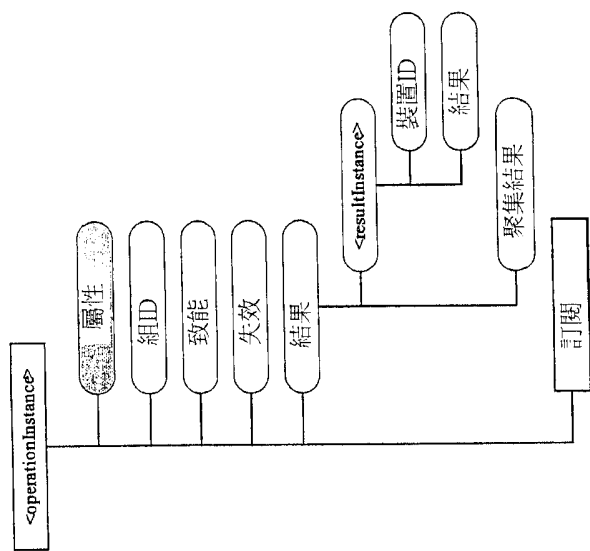
第43A圖



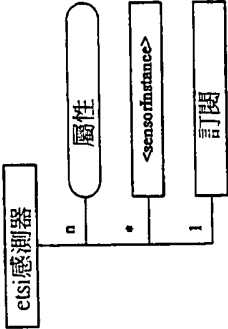
第43B圖



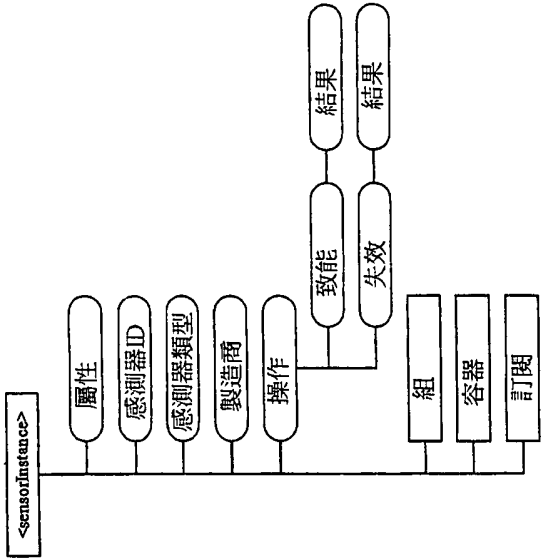
第44A圖



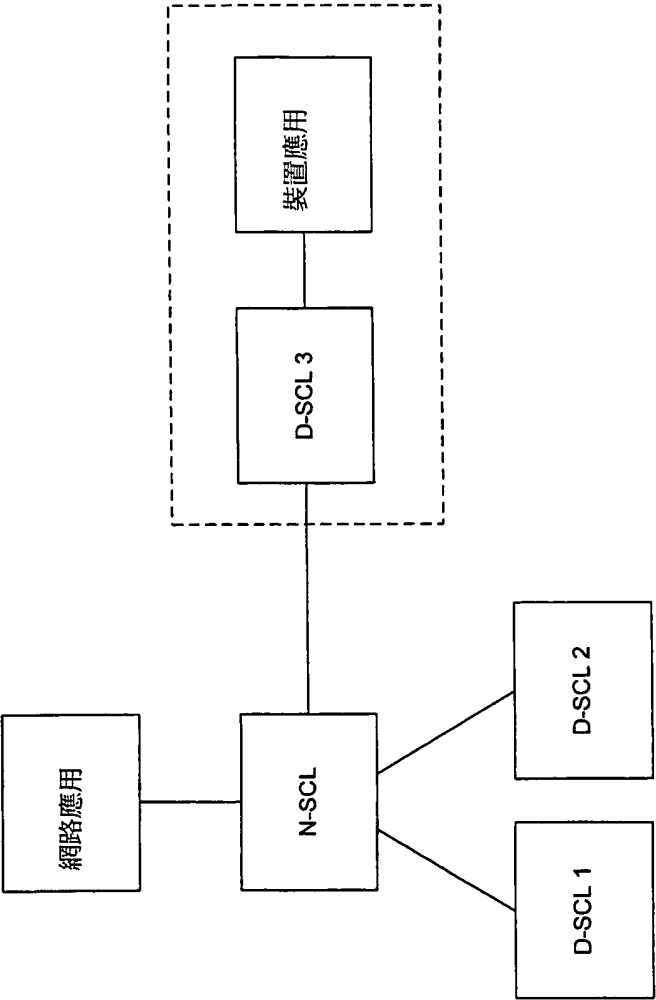
第44B圖



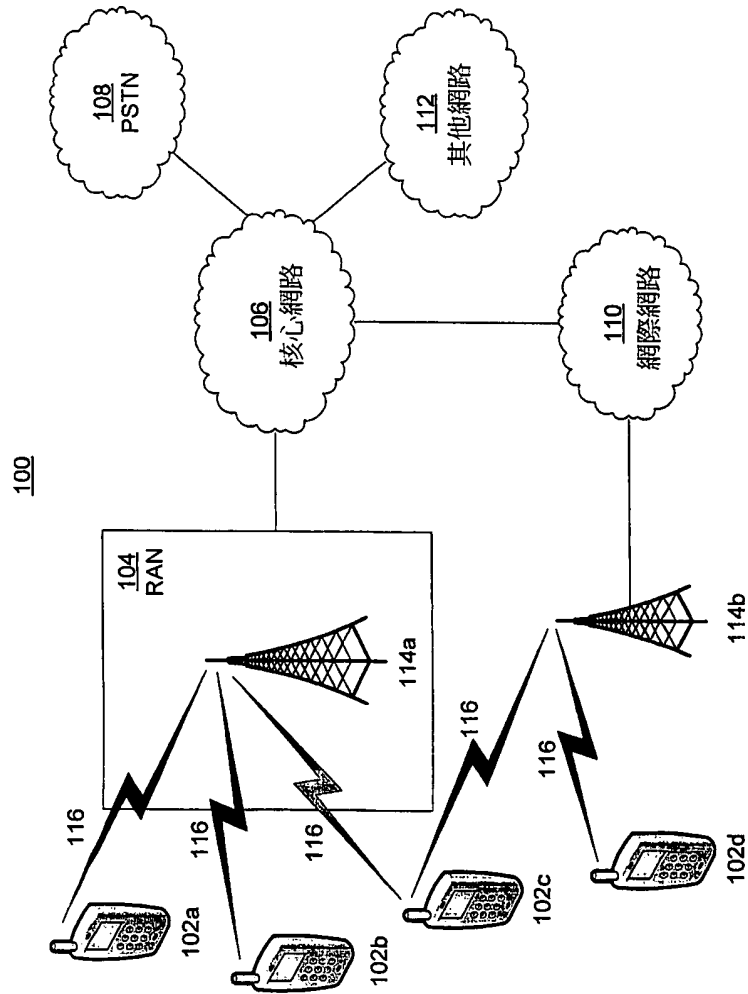
第45圖



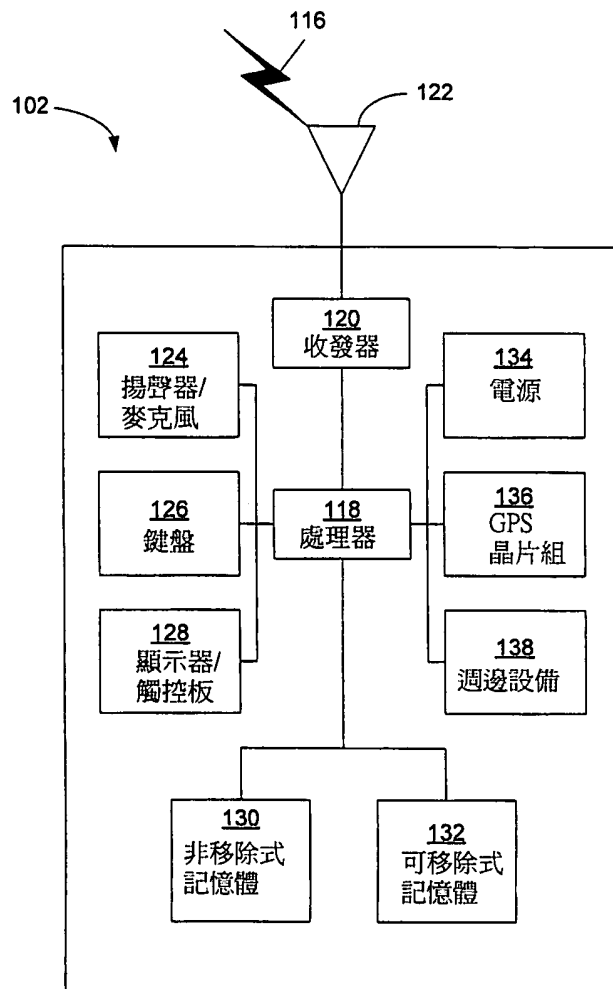
第46圖



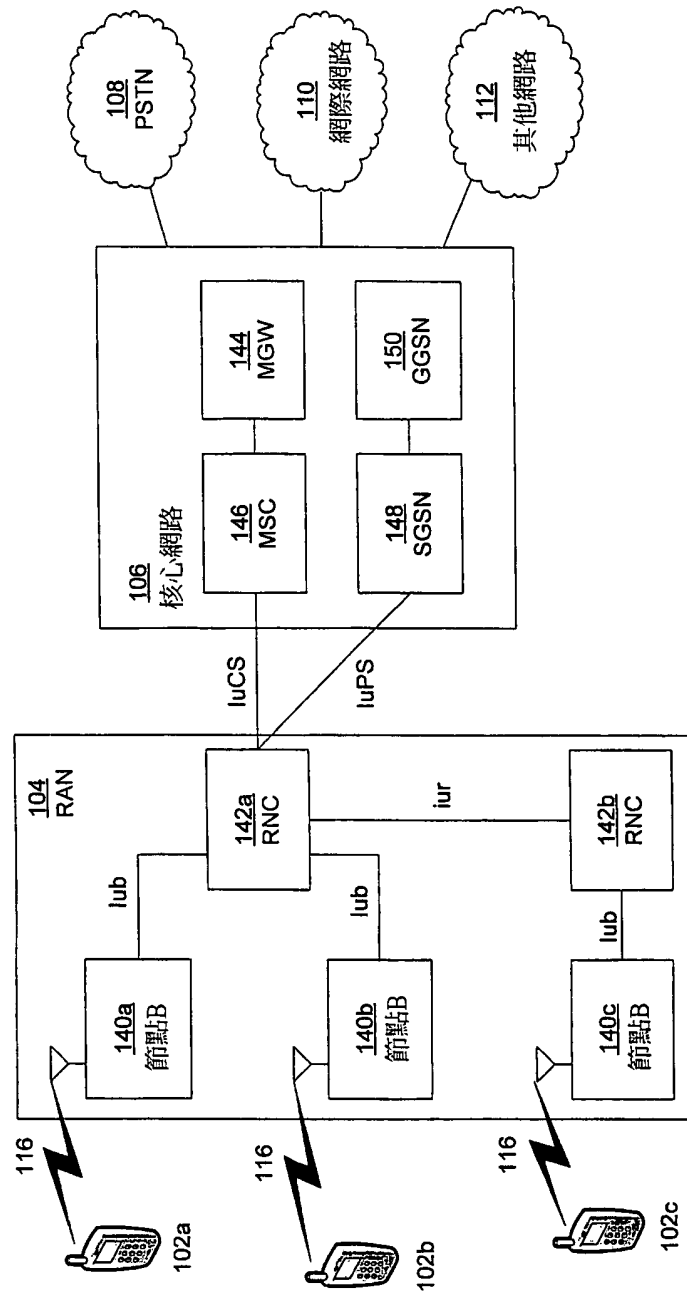
第47圖



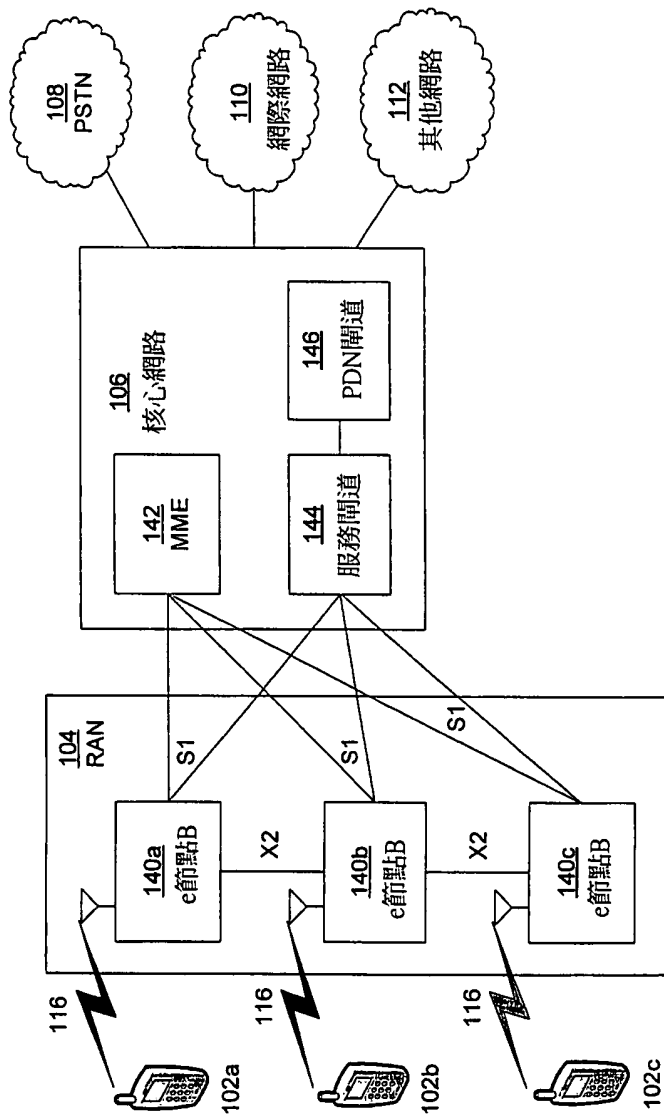
第48A圖



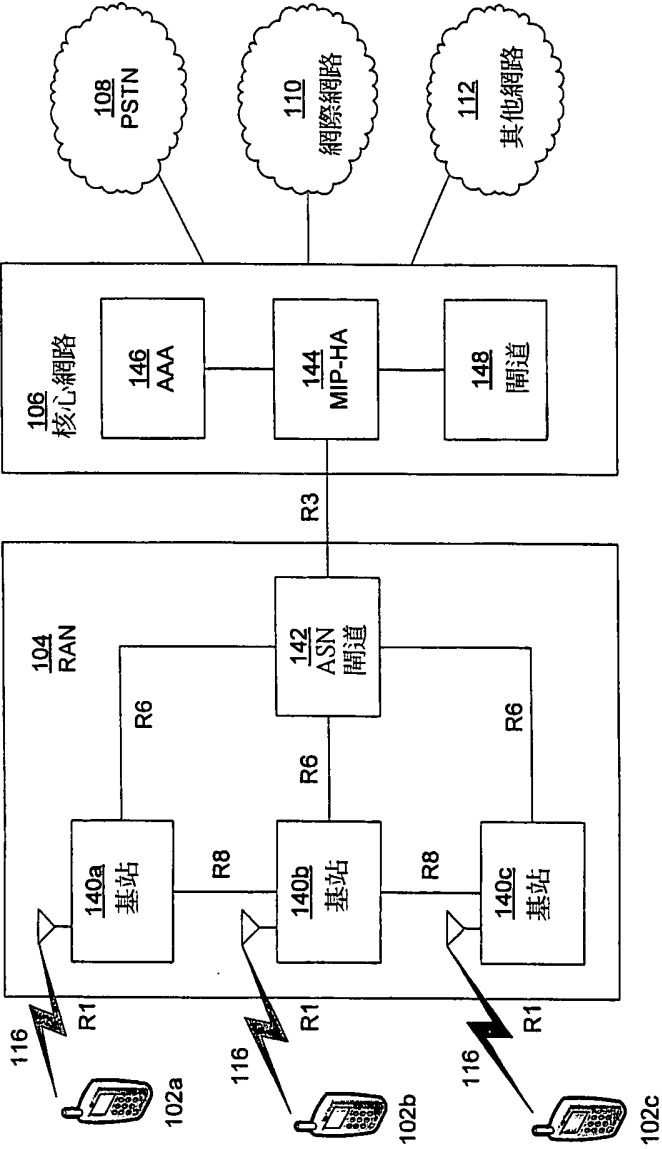
第48B圖



第48C圖



第48D圖



第48E圖

四、指定代表圖：

(一)本案指定代表圖為：第(1A)圖。

(二)本代表圖之元件符號簡單說明：

d1a、m1d、m1a：參考點

SC：服務能力

12：裝置域

14：網路域

16：網路應用域

18a、18b：M2M網路應用

20：存取和/或核心網路

22：傳輸網路

24b、24c：M2M伺服器

26c：網路服務能力層(“N-SCL”)

34a、34b、34c、34e、34f、34g：M2M裝置

36a、36b：M2M閘道

38a、38b：M2M區域網路

40a、40b、40c、40d、40f、40g：M2M應用(“DA”)

44a、44b、44c：SCL(“D-SCL”)

五、本案若有化學式時，請揭示最能顯示發明特徵的化學式：