第十七章

數位資源命名管理系統

Handle Systems of Digital Resources

陳昭珍
Chao-Chen Chen
國立臺灣師範大學圖書資訊研究所 教授
Professor, Graduate Institute of Library & Information Science, National Taiwan Normal University

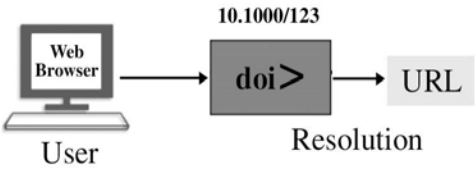
葉建華
Hian-Hua Yeh
國立臺灣大學資訊工程學系 博士後研究
Post-Doctoral Researcher, Department of Information Engineering, National Taiwan University

壹、前言

網路上的資源若要有效的檢索、管理與利用，必須有好的蒐尋機制，而對數位資源而言，此蒐尋機制的基礎應包括：詮釋資料(metadata)、數位資源識別碼(identifiers，或稱 handles)、及命名管理系統(handle system)。有了詮釋資料後，資源才可能被檢索；有數位資源唯一的識別碼，數位物件的版權才得以控制；而若數位資源移位時，則需透過彈性的命名管理系統，才可找到該數位物件。由於數位資源會改變存放的位置，因此在詮釋資料中不會記錄該物件的絕對位置，乃記錄其數位資源識別碼，所以必需有一數位命名管理系統，將數位資源識別碼（handle）解析爲 URL (Uniform Resource Locator)。簡單的數位資源命名管理系統主要的功能乃在將數位資源唯一的識別碼，解析爲其對映的一個 URL，如圖一所示。而複雜的數位資源命名管理系統則可將一個數位資源識別碼對映到數個

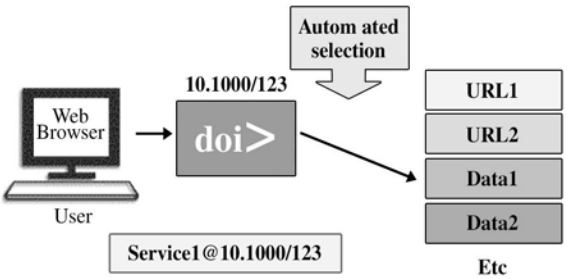
URL，如圖二。

此外，數位資源命名管理系統有一套完整的語法與架構。本文之目的主要在探討此命名管理系統之意義、功能、命名空間、架構、服務與安全機制。



圖一　簡單的數位資源命名管理系統

資料來源：“ DOI Handbook”, http://www.doi.org/handbook_2000/resolution.html



圖二　複雜的數位資源命名管理系統

資料來源：“ DOI Handbook”, http://www.doi.org/handbook_2000/resolution.html

貳、數位資源命名管理系統簡介

一、數位資源命名管理系統定義範圍

所謂的「數位資源命名管理系統」，是一種用來在網路上提供全域名稱服務的分散式資訊系統。這樣的系統，除了希望能夠達到上述的功能之外，還必須兼具幾項特性：具有運作效率、具有擴充彈性、以及具有安全保密特性等等。在數位資源命名管理系統的設計中，主要包含了一個開放性的通訊協定、一個命名空間、以及該通訊協定的參考實作部分。而在這個開放性的通訊協定定義下，構成了訂定數位資源識別碼的分散式儲存與解析查詢機制，同時也完成了針對網路資源做

定位與存取的工作。識別碼所對應到的資料值可以因應實地的需要而改變其狀態，但是識別碼本身卻不需因此而更動；因此在資源狀態變動的情形下（如儲存位置改變），識別碼本身的特性可以維持持續不變。在此同時，每個識別碼都會有其所屬的管理者，同時識別碼的管理工作也可以在分散式的環境下運作無誤。再者，識別碼的運作也具有安全機制，可以在一個授權允許的管理程式中進行。

數位資源命名管理系統提供了一個聯合的機制，允許現存的區域性命名空間在取得唯一命名授權（naming authority）的前提下，加入其全域的命名空間。在這樣的做法下，區域性的命名以及其關聯到的資料值便不會因加入數位資源命名管理系統而有所改變。任何對該區域性命名空間所發出的識別請求，都會透過特定的數位資源命名管理系統通訊協定服務介面獲得適當的處理。由於現存的區域命名空間都要取得唯一命名授權才能夠加入全域的命名空間，因此也間接地確保了原網路資源的區域命名也具有唯一的特性。

目前已經有許多已知的服務提供網際網路資源命名的服務，其中較有名的便是最常見的網域名稱服務(Domain Name System， 簡稱 DNS) [2,3]。DNS 服務主要的任務，就是要提供名稱資源對網路 IP 的對應服務，因此適用在各種不同的網路主機、網域、通訊協定、網路群組、以及各種機構 [3]。目前由於網際網路成長迅速，因此對於 DNS 服務也產生了若干功能性的延伸需求，甚至希望 DNS 可以當作一個一般性的資源命名系統。但是由於 DNS 服務原本是著重在基本的網路路由功能上，其特性間接使得它無法轉變成爲一個一般性的資源命名系統。另一個困難點是在於 DNS 的管理模式，通常是以 DNS 網域層級爲管理單位，因此無法提供針對單一識別碼的管理架構。也正因爲如此，DNS 服務較適合用在管理網域名稱的目的上，而非一個一般性的資源命名管理系統。相對來說，數位資源命名管理系統是希望設計用來處理大量的網路資源，同時又希望具備針對單一識別碼的管理架構。因此數位資源命名管理系統的資料模型允許對個別的識別碼資料進行定義，而每一個識別碼更可以定義自己的管理者，以便日後透過數位資源命名管理系統的授權協定來進行資料管理的工作。

二、 數位資源識別碼與 URL 的意義

由於識別碼就是一個永續性的代號單位，其所對應到的資料值可以因應實地的需要而改變其狀態，但是本身卻不需因此而更動，因此在某些特性上與網際網路上常用的 URL 相當雷同。所謂的 URL [4]，就是將網際網路的資源命名爲 DNS 名稱與區域性名稱的結合體。所謂的區域性名稱，可以是一個區域性的檔案系統名稱，也可以一個對某種區域性服務的參考（如 CGI-bin 程式），這種以 DNS 名稱與區域性名稱的結合體，造就了一個針對命名與管理網際網路資源的彈性管理模型。但是這種模型並非沒有限制。大多數的 URL 應用協定（如 HTTP）都只是定義用來做解析服務用途而已，而任何一個 URL 的管理都是必須透過該部主機或是特定的網路服務（如 NFS）來完成。因此使用 URL 來當作名稱服務基本上都會與該項網際網路資源所在的位置緊密結合，同時更與該項資源的檔案系統存放路徑息息相關。而當該項網路資源的存放位置移動之後，其所對應的 URL 就會變成無效。

三、數位資源命名管理系統的目標

由於數位資源命名管理系統的設計就是希望能克服上述各項限制，同時也希望能夠加入一些重要的功能，因此在數位資源命名管理系統的設計上，有以下幾項重要的目標要達成，如表一所示。

表一 數位資源命名管理系統的設計目標

特性	說明
唯一性	每一個識別碼對數位資源命名管理系統來說都是唯一的。
永續性	一個識別碼的訂定，並不是根據該項資源單位的名稱而來，而是個別獨立給予的。即使我們使用該項資源單位的名稱來訂定識別碼，該項資源與其識別碼也只有在數位資源命名管理系統中存在著對應。當然使用相同的名稱並不能確保其永續性，僅有管理上的便利

特性	說明
	而已。當網路資源的位置異動時，只需要修改數位資源命名管理系統中所紀錄的位置即可。
多重實體	一個單一的識別碼可以參照對應到多個資源的實體，也就是說，一個識別碼可以對應到多項不同網路存放地點的資源。因此，應用程式將可以依據這個特性，來增進其效能與可靠性。舉例來說，一個網路服務就可以利用單一識別碼對應到多個服務窗口來分散其服務負載。
具有可延伸性的命名空間	目前既有的區域性名稱服務，可以透過取得唯一命名授權的前提下，加入其全域的命名空間。在這樣的做法下，區域性的命名以及其關聯到的資料值便不會因加入數位資源命名管理系統而有所改變。同時也不會與現有的各項命名服務產生衝突。
支援國際編碼標準	數位資源命名管理系統中的識別碼命名空間是根據 Unicode 2.0 [1] 爲基礎，涵蓋了目前世界上絕大多數的文字，也間接便利各種語文的系統環境。而識別碼協定也使用 UTF-8 [5] 來當作基本的編碼方式。
分散式的服務模型	數位資源命名管理系統定義了一種階層式的服務模型，任何一個區域性的識別碼命名空間，均可以對應到一個區域性的服務，甚至是一個全域性的服務。對於全域性的服務來說，我們稱之爲全域性識別碼註冊資料庫（Global Handle Registry, 簡稱 GHR），其功能是在於將各種識別碼服務需求分配到適當對應的區域性識別碼服務。這種分散式的服務模型，允許任何定義好的資源擁有多個服務窗口，而每一項服務，又可以對應到多個個別獨立的伺服器，因此具有顯而易見

特性	說明
	的分散式服務架構。
安全的名稱存取服務	數位資源命名管理系統定義了安全的名稱解析過程，同時也定義了安全的管理方式。系統中的協定也定義了客戶端與伺服端的標準授權方式。同時系統也提供了服務完整性的確保以及資料的保密性。
有效率的名稱解析服務	識別碼協定定義的其中一個目的，便是用來提昇命名解析的效率。爲了避免在管理服務層面的運算代價影響名稱解析的效率，在管理服務方面與識別碼名稱解析被設計成可以分開定義的形式。

參、 數位資源識別碼的命名空間 (Name Space)

一、 數位資源識別碼的結構

每一個識別碼都會包含兩大部分：一是其授權命名的部分，也就是我們所謂的前綴（prefix）；另一則是接在授權命名後方一個唯一的區域性名稱，我們稱之爲後綴（suffix）。授權命名與區域性名稱之間，則使用 ASCII 碼的 "/" 來分開。因此，一個識別碼應該是定義成以下這種形式：

```
<Handle> ::= <Handle Naming Authority> "/" <Handle Local Name>
```

舉例來說，

"10.1045/january99-bearman" 是一個在 D-Lib 雜誌 [13] 發表文章的識別代號，它的授權命名部分是 "10.1045"，而區域性名稱則是 "january99-bearman"。

識別碼的命名空間可以視爲許多個區域性命名空間的總集合(superset)，而這些區域性的命名空間則有各自唯一的授權命名。授權命名可以代表建立該相關識別碼的命名空間管理單位，而每個授權命名在數位資源命名管理系統的系統中，

都是獨一無二的。任何現有的區域性命名空間都可以透過取得一個授權命名後加入全域性的識別命名空間。而由此定義出來的識別碼將具有上述所提的授權命名與區域性命名所組合而成。

二、數位資源識別碼所使用的編碼

識別碼是由任意個可以列印的字元所組成，這些字元是由 ISO/IEC 10646 統一字集（Universal Character Set）UCS-2 所定義，也就是與 Unicode 2.0 完全相容的字元集。UCS-2 涵蓋了大部分世界上目前所使用的文字，因此適合用在識別碼的構成上。而爲了與目前大多數的現存系統相容，而不致發生不同編碼的情形，因此識別碼協定使用了 UTF-8 來當作識別碼編碼的唯一選擇。UTF-8 保留了任何的 ASCII 編碼名稱，以確保和現有系統的最大相容性，避免命名時的任何規格衝突。部分與全域性命名空間的編碼的議題以及 UTF-8 編碼的相關討論請參見 [14]。

基本上來說，識別碼本身的字元大小寫有區分，但是爲了不必要的混淆，任何一個識別碼服務，包括全域性的服務，都會將自己的命名空間定義爲大小寫不區分的狀態，以避免不必要的麻煩。

三、識別碼的命名授權

識別碼的命名授權(Handle Naming Authority)也是以階層式的結構存在，也就是說，會以樹狀結構的 形式存在。每一個樹狀結構中的節點都會擁有一個標籤，用以對應到一個命名授權的區段（segment），而父節點代表了擁有其子節點的命名授權。與 DNS 服務不同的是，識別碼的命名授權是由左而右所構成的，分別由樹狀結構的根節點以下各節點串接而成，而不同的標籤之間則以 ASCII 字元 "." (0x2E)來銜接。舉例來說，在國會圖書館命名授權（"loc"）下的 National Digital Library Program 命名授權("ndlp") 就定義成 "loc.ndlp"。

每一個命名授權都可以擁有其子代的命名授權登記。任何一個子代命名授權只能向自己的父代登錄，在此同時父代的命名授權也必須先行登錄。雖然如此，任兩個命名空間之間的關係也僅止於父代與子代之間的關聯，除此之外，並沒有

其他的內隱管理關係。父代和子代的命名空間可以由不同的識別服務來提供，甚至父代與子代的命名服務之間也可以擁有完全不同管理權限。

每一個識別碼都會定義在某一個命名授權登記下，而在授權命名與區域性名稱之間，則使用 ASCII 碼的 "/" (0x2F)來分開。在某個授權命名下的區域性命名集合，便稱爲區域性命名空間。任何一個區域性名稱對該區域性命名空間來說，一定是唯一的，這樣才確保任何一個在數位資源命名管理系統中的識別碼都能夠維持全域唯一的特性。

肆、數位資源命名管理系統架構

一、階層式的模型

在數位資源命名管理系統設計中，定義了一個階層式的服務模型，其中最上層的服務稱爲全域服務，也就是之前所提到的全域性識別碼註冊資料庫（Global Handle Registry, 簡稱 GHR）。在這層服務之下的所有服務，也都與識別碼的各項功能有關，我們通稱爲區域性的識別碼服務。全域性識別碼註冊資料庫 GHR 提供了識別碼解析的服務，同時也具有管理各個識別碼命名空間的功能。GHR 最爲特別的地方在於它提供了管理識別碼命名授權的功能與服務，這些識別碼命名授權則是用來在各個命名空間中管理識別碼的一部份。而運用這些命名授權所產生的識別碼則包含了可以存取並使用區域服務的各項重要訊息。

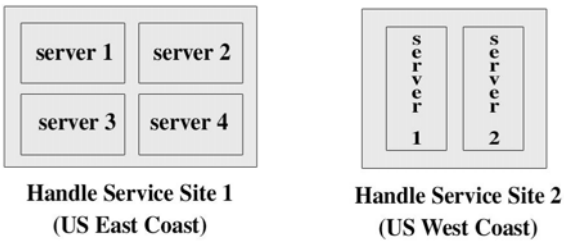
在區域性的識別碼服務層級中，則包含了所有用來管理在特定命名授權下的識別碼的服務，並提供解析與管理區域性命名的功能。這些區域性的服務是由各個具有管理專責的機構來負責，針對其管理範圍中的識別碼來做管理。

二、分散式的特性

另一個數位資源命名管理系統架構的重要層面就是具有分散式的特性。在數位資源命名管理系統的架構下，包含了許多個個別的識別碼服務，這些識別碼服務也是由一個或多個識別碼服務主機群所組成。而這些識別碼服務主機群，也會

根據自身的需求，完整或部分複製個別的識別碼服務。每一個提供識別碼服務的主機群也都包含了一或多個識別碼伺服器。由此可以得知，數位資源命名管理系統中可以包含的識別碼服務數量，是沒有總和上限的；同時，每項服務可以包含的主機群也沒有數目上限；再者，每個主機群可以包含的伺服器數目也沒有設定上限。因此，就複製個別的識別碼服務來說，並不需要各個主機群包含相同數目的伺服器，也就是說，每個主機群只需要複製相同數量的識別碼集合，使其分散在主機群範圍中的各個伺服器即可。這種分散式的做法是試圖帶入擴充特性，同時也避免了單一伺服節點的問題會影響整個服務的狀況。

圖三說明了一個單一的識別碼服務的架構。這項服務是由兩個服務主機群所構成，其中一部份設置在美國東岸（US East coast），另一部份則設置在美國西岸（US West coast）。設置在美國東岸的主機群是由四部伺服器來服務所有外來的客戶端請求，而設置在美國西岸的主機群，因爲配置了較高性能的電腦，因此只需由兩部伺服器來服務所有外來的客戶端請求即可。有此可知，針對任何數位資源命名管理系統所需要服務主機群的數量，或是每個主機群中所包含的伺服器數目，都可以因爲不同的服務需求而予以動態的增減。



圖三　單一識別碼服務架構

資料來源：Handle System Overview, http://www.handle.net/overview-current.html

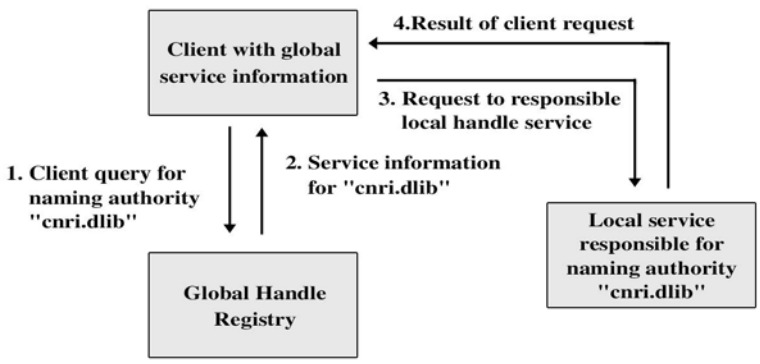
每一個識別碼服務的責任就是管理數位資源命名管理系統下的某個命名空間子集合，這個子集合主要是由一定數量、不同命名授權所構成的識別碼所組成。通常我們會將識別碼服務稱爲這些命名授權的"home"服務，同時這也是唯一提供所屬識別碼管理與解析服務的地方。在解析服務真正運作之前，客戶端必須先確

認該識別碼所屬的"home"服務，這個"home"服務正是預先登錄在全域性識別碼註冊資料庫（GHR）中的資訊。而這項確認的動作，則是由客戶端的軟體來完成。

三、全域性識別碼註冊資料庫的特性

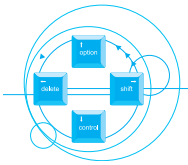
所謂的全域性識別碼註冊資料庫（GHR），其功能就是用來管理命名授權部分的識別資訊。這種命名授權的識別資訊，主要是包含了描述該項命名授權所代表"home"服務的相關服務資訊，而這樣的服務資訊內容包括了該項識別碼服務的服務主機群，以及各個識別碼伺服器所擁有的介面。因此，任何一個客戶端如果想要取得特定識別碼的"home"服務，就必須先根據其命名授權來向 GHR 查詢相關的服務資訊。這些提供給客戶端的服務資訊，就是將來可以讓客戶端與特定"home"服務溝通的必要資料。

圖四說明了一個典型的識別碼解析程序的各項步驟，其中包括區域性識別碼服務（也就是"home"服務）的運作內涵。在這個範例中，客戶端試圖去解析 "cnri.dlib/july95-arms" 這個識別碼，並且先向 GHR 查詢這個識別碼所屬的"home"服務。藉由向 GHR 發出的查詢，可以得知識別碼命名授權部分"home"服務的所在位置。GHR 根據客戶端的需要，傳回命名授權爲"cnri.dlib"的相關服務資訊，也就是其區域性識別碼服務的相關資訊。當然了，這樣的資訊一定也包含了識別碼爲 "cnri.dlib/july95-arms" 的相關服務。客戶端得到這些資訊之後，便可以得知區域性識別碼服務的溝通方式，並藉以繼續之後的識別碼解析動作。



圖四　典型識別碼解析程序

資料來源：Handle System Overview, http://www.handle.net/overview-current.html



而爲了增進解析程序的效率，任何一個對 GHR 發出過查詢的客戶端都可以因其需要將 GHR 傳回的服務資訊暫存（cache）起來，並利用這項暫存的資訊來處理接下來需要使用到的各項查詢。不論是利用一個獨立運作的暫存伺服器，抑或是使用一個小型的暫存機制，都可以提供區域性使用者的查詢需要，利用資料共享進而增進查詢效率。一旦有這樣的暫存機制在運作，對於同一個識別碼的解析查詢就不必再經過任何其他識別碼服務，而僅僅需要暫存服務的資料即可。而對於客戶端查詢命名授權而言，也不必再去查詢 GHR 來取得相關資訊，只需要由存伺服器中提供即可。

伍、數位資源命名管理系統的服務與安全機制

一、數位資源識別碼管理服務

數位資源命名管理系統一方面提供了識別碼解析的服務，同時也提供了透過網際網路做識別碼管理方面的服務。每一個識別碼都有一組指定給它的資料值，客戶端可以運用識別碼解析服務來取得特定識別碼所內含的資料值。每一組資料值也包含了一個資料型態以及一個唯一的資料索引，因此客戶端可以透過資料型態以及資料索引來查詢特定的識別碼資料值。

識別碼管理服務主要是處理來自於客戶端的識別碼管理請求，包括新增識別碼、刪除識別碼、以及更新識別碼所關聯到的資料值。這項服務也包括了透過使用命名授權識別碼來進行對識別授權的管理。每一個識別碼都可以定義自己所屬的管理者，同時每一個管理者也被賦予一定的管理權限。所有這些管理性的服務，則都必須先透過使用識別碼系統的授權協定才能夠進行。

數位資源命名管理系統提供了客戶端各種有關授權確認以及資料完整性的各項服務，但是就基本的客戶端請求而言，是不需要任何的授權確認動作的。但無論如何，這種識別碼解析的請求，如果是針對具有機密性資料的識別資料的話（如與管理者相關的資料），抑或是所有與管理相關的請求（如新增或刪除識別碼的資料值），這時候就需要必要的客戶端授權確認動作。而在確認動作需求發生的同

時，識別碼伺服器就會對發出請求的客戶端送出確認的認證要求，而爲了通過這項認證，客戶端必須將必要的管理者身分資料或是其他必要的權限資料，以取得接下來必要的操作需求。識別碼伺服器的工作就是去完成處理最初所發生的授權確認需求，而相對的發出請求的客戶端則必須使用私密鎖鑰（secret key）或是公開鎖鑰（public key）來進行資料的保密編碼，以確保授權後的資料溝通。

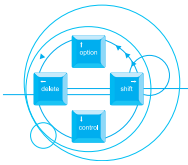
對查詢識別碼的客戶端來說，也可以向識別碼伺服器請求取得具有數位簽章的回應，以確保資料的安全與完整。除此之外，任何的識別碼伺服器或客戶端都有建立安全通訊階段的選擇，資料只要是透過這樣的安全通訊階段來進行溝通的話，都會經過特殊鎖鑰的編碼後再行傳輸，以確保資料的安全與完整。

數位資源命名管理系統提供了在伺服端與客戶端之間的安全通訊機制，但這並不是指稱所有的識別碼資料值都會具有認證的特性，也就是說，錯誤的識別碼資料值也有可能誤導使用這種資料的客戶端。相對來說，任何的資料值紀錄也有可能包含間接參考到那些具有認證特性的識別碼資料值。舉例來說，一個識別碼資料值紀錄 R 可以包含對另一個識別碼資料值的參考，而這個被參考到的識別碼資料值則有可能包含了紀錄 R 中的數位簽章，而經過這項簽章認證之後，客戶端便可以確認紀錄 R 的安全性與資料完整性。

二、公開鎖鑰機制 (Public Key System)

要能夠順利使用數位資源命名管理系統中的識別碼管理服務（如新增識別碼、更改識別碼、刪除識別碼等等），首先客戶端自身必須先通過數位資源命名管理系統的授權認證，然後才可以順利使用這些服務。至於要如何通過授權認證呢？首先客戶端必須先傳送自己的 ID 代號，同時這個 ID 代號在數位資源命名管理系統中是全域唯一的代號。當然了，基於數位資源命名管理系統本身的特性，這樣的代號也是用識別碼來表示。

這種管理用的識別碼包含了一個公開鎖鑰或一個私密鎖鑰（也就是用來當作密碼的部分），這樣的鎖鑰可以用來表示並確認該識別碼。如果在數位資源命名管理系統中有某一個管理用的識別碼具有特定的權限，可以執行特殊的程序的話，在整個特殊程序的執行過程中，就可以透過這種公開鎖鑰或私密鎖鑰來取得認證。



當我們欲請求使用某個命名授權時，系統中就會建立一個命名授權識別碼，並用來當作管理性的識別碼。因此這個這個命名授權識別碼（如 0.NA/123456）就可以同時當作管理用的識別碼，也就是同時具有兩種用途。接下來我們就把重點放在命名授權識別碼管理方面的功能，並加以討論。

你可以使用具有識別碼功能的客戶端來查詢你的授權識別碼，並觀察其中所包含的資料值，或者是透過一個識別碼代理伺服器來完成這項工作（可以經由查詢 http://hdl.handle.net/得知，請注意，如果你在 http://hdl.handle.net/之後加上某個識別碼的話，這個伺服器將會自動將你所發出的識別碼轉換成相對應的網路資源 URL），這樣你將會可以觀察到與這個授權識別碼相關的公開鎖鑰或私密鎖鑰資料值。而當你查詢某個一般的識別碼時，你將會收到一個或多個與這個識別碼相關的資料值。除此之外，每一個識別碼資料值都會擁有一個唯一的數字索引值，也就是所謂的型態標示代碼。以下列出一些在數位資源命名管理系統中具有特殊意義的識別碼資料值：

表二　數位資源命名管理系統的識別碼資料值

資料值	說明
管理性資料值	每一個識別碼都必須包含一個管理性的資料值。這項管理性的資料值的型態標示代碼爲 HS_ADMIN，同時這個資料值指定了識別碼管理者的操作權限，如變更識別碼或是刪除識別碼等等。同時這個管理性的資料值的數字索引是自 100 起算，如果具有一個以上的管理性資料值的話，便會以 101、102、103 的順序逐一編列下去。這個管理性的資料值主要目的是要定義管理工作的使用權。
公開鎖鑰資料值	如果某個特定的識別碼擁有一個公開鎖鑰的話，就會擁有 HS_DSAPUBKEY 型態的資料值。公開鎖鑰資料值的數字索引是自 300 起算，如果具有一個以上的公開鎖鑰資料值的話，便會以 301、302、303 的順序逐一編列下去。
私密鎖鑰資料值	如果某個特定的識別碼擁有一個私密鎖鑰的話，就會擁有

	HS_SECKEY 型態的資料值。私密鎖鑰資料值的數字索引是自 300 起算，如果具有一個以上的私密鎖鑰資料值的話，便會以 301、302、303 的順序逐一編列下去。請注意，這項資料值通常無法在公開的場合閱讀，因此在一般的非授權性的查詢上，這個私密鎖鑰的資料值是不會呈現出來的。
群組資料值	所謂的群組資料值，就是包含了一個識別碼資料值的列表，其中包含了公開鎖鑰、私密鎖鑰、以及其他群組的資料。如果管理性的資料值中指定某個群組資料值爲管理者的話，這便意味著這個群組中的各項資料值都可以是該識別碼的管理者。

三、安全性認證方面

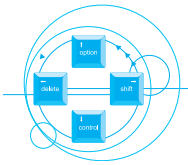
一旦你擁有了一個管理性的識別碼之後，也就代表了你擁有一定程度的授權，可以進行某些管理性的工作。而當你要進行這些管理性工作的同時，你必須同時提供以下的資訊來進行運作：

1、 你所擁有的管理性識別碼：這同時也是命名授權的識別碼，此外還必須包含該識別碼所擁有的公開鎖鑰或私密鎖鑰。

2、 你所擁有的公開鎖鑰或私密鎖鑰：請注意，你所擁有的公開鎖鑰或私密鎖鑰絕對不會由數位資源命名管理系統透過網際網路來傳送，同時你也不應該將公開鎖鑰或私密鎖鑰透過網際網路來進行傳遞。你只需要將這項資料輸入到你的數位資源命名管理系統客戶端軟體之中，這樣你便可以向任何一個識別碼伺服器證明你擁有特定的資料。

爲了達到建立識別碼的目的，該命名授權的擁有者必須授權給欲使用建立識別碼功能的對象。要達到這項工作，只需要加入新的管理性識別碼，以及相對應的鎖鑰值，這樣便完成加入新的管理者的工作。也就是說，這樣新的管理者便開始有權限進行建立識別碼的程序。

接下來，當我們真的對識別碼伺服器送出建立識別碼的請求時，就必須將我



們得到授權的相關資訊提供出去，識別碼伺服器會確認我們是否具有相關操作程序的權限，也就是去確定我們是否是某特定命名授權的管理者之一。一旦確定，我們便可以進行建立識別碼的工作。

陸、數位資源識別碼解析的步驟

一、 解析流程

數位資源命名管理系統解析識別碼的方式是以分散式的模型爲基礎，而客戶端方面則有幾種可能：可以是具有這方面功能的客戶端、或是一般具有特殊外掛程式（plug-in）的 Web 瀏覽器、抑或是一個透過特定代理伺服器（proxy）的一般客戶端。不論是以上哪一種情形，這些客戶端都會透過數位資源命名管理系統所定義的通訊協定來與數位資源命名管理系統溝通，而這些通訊協定都是已經經過正式定義，甚至有些已經實作出來了。解析的流程步驟，可以由圖五來簡要表示之：

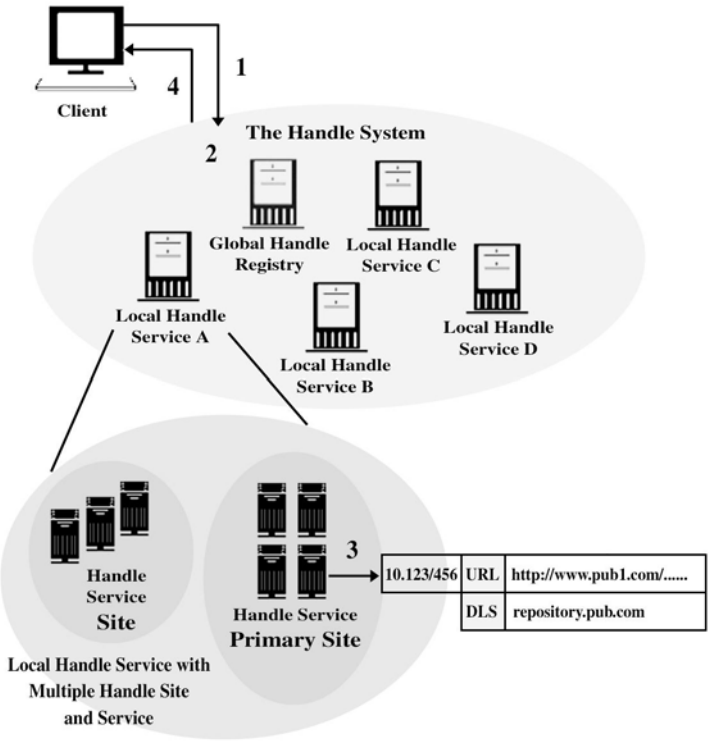


Figure 2- Handle SystemArchitecture&Operation

圖五　數位資源命名管理系統解析流程

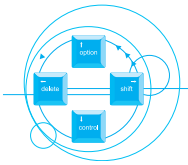
資料來源：Handle Resolution, http://www.handle.net/overviews/overview.html

二、流程討論

如圖五所示，我們可以將識別碼解析的程序，分爲以下四個步驟（即圖五所標示的 1~4）來完成，其中：

步驟 1：某個客戶端（如 Web 瀏覽器）遇到了解析識別碼的需求（如 10.123/456），通常這樣的情形會發生在某個特殊的超連結，抑或是某種特別的連結參考情形下。不論是在網際網路上，或是在個別的內部網路中，這個客戶端都會針對數位資源命名管理系統送出解析這個識別碼的需求。而如上節所述，這項工作可以由客戶端自己直接完成，也可以透過特定代理伺服器（proxy）來執行。

步驟 2：在數位資源命名管理系統中，包含著由識別碼服務所形成的集合，每項服



務則包含了至少一個主要的主機群以及一定數量的次要主機群，而每個主機群也都是由一定數量的識別碼伺服器所構成。就解析的程序來看，每一個主機群都複製了隸屬於該項識別碼服務的所有識別碼。而其中一個特別的服務，也就是所謂的全域性識別碼註冊資料庫 GHR，則是負擔著指引出各項區域性命名服務的重責大任。每個區域性的服務也會知道如何去存取 GHR。這樣的關係允許識別碼解析請求在任何一處發生，同時也可以順利的將請求引導到負責該項識別碼解析的特定服務。

步驟 3&4： 每一個識別碼都會關聯到一個或多個具有特定型態的資料值。在本例中，識別碼 10.123/456 就會關聯到兩種資料值，其中一個是 URL 型態的資料，另一個則是一種新的協定叫做 DLS，這樣的資料最後會回傳到客戶端供其參考。請注意，這所謂多個具有特定型態的資料值，也有可能會是同一種資料型態，如多個 URL 等等。也就是說，一個單一的識別碼資料可以對應到多個 URL 資料值，以供後續的運用。但是在此同時，數位資源命名管理系統並不會對這些資料的後續使用做任何的假設，也就是說，這些資料的運用完全取決於客戶端的需求。在這樣的情況下，便可以允許應用程式擁有最大的彈性來利用數位資源命名管理系統所提供的命名服務。在本例中，該客戶端可以使用資料值的通訊協定來定位該項網路資源，當然了，這些都完全取決於客戶端的使用目的。

柒、目前已定義的數位資源識別碼資料型態

每一個識別碼都有一組指定給它的資料值，而這些可以視爲一個由多項欄位所構成的一筆記錄。每一項資料值都是由特定種類的資料型態所組成，這些資料型態可由下表表示，以供非管理方面使用。至於管理方面所使用的資料型態，本文不另討論

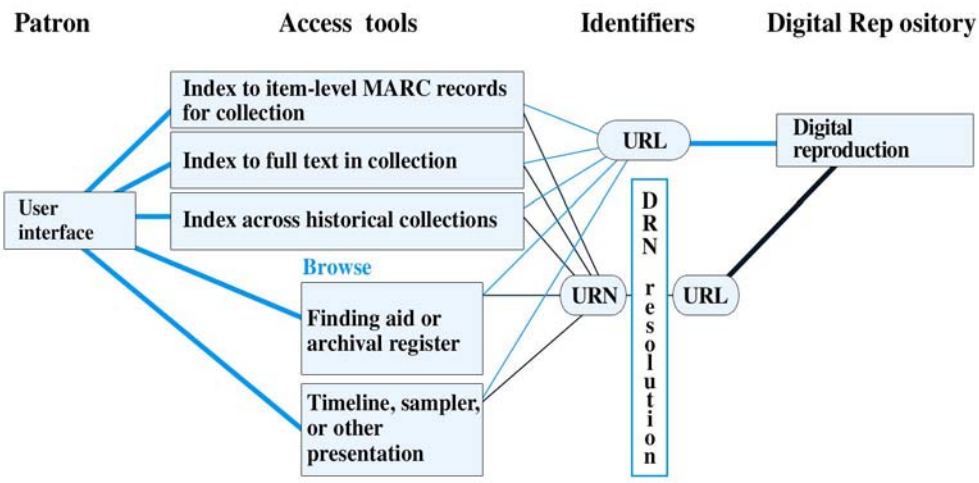
表三 常見的識別碼資料值

型態	資料值
URL	URL 型態的資料值是由 UTF-8 編碼所構成的一般 URL 字串，這項 URL 字串主要是用來表示特定識別碼所代表的資源物件的所在位置。
EMAIL	EMAIL 型態的資料值是由 UTF-8 編碼所構成的一般 email 位址。

所謂的”型態”定義的資料值，可以是任意的 UTF-8 編碼所構成的字串，目前預先定義好的型態就如上表所示，有 URL 與 EMAIL 兩種，數位資源命名管理系統的使用者在使用型態資料值得時候就必須特別注意。至於其他種型態的資料值，目前也正廣泛在討論之中。

捌、數位資源命名管理系統在數位典藏的應用實例

西方國家的數位圖書館系統，皆會建立數位資源命名管理系統以連結詮釋資料與數位資源，這也是傳統的書目系統中不用處理的部份。如美國國會圖書館 American Memory 系統中的每一數位檔案，將採用 URN 命名，並將檔案儲存在階層式目錄中，透過數位資源命名管理系統將邏輯名稱解析出實際儲存資料的位置。如圖六所示：[21]



圖六 美國國會圖書館 American Memory 系統

資料來源：The Library of Congress, National Digital Library Program, “Identifiers for Digital Resources”, 13 Aug 1996, http://lcweb2.loc.gov/ammem/award/docs/identifiers.html (2002/4/12).

國內在建立數位圖書館相關系統時，對於數位資源的命名與數位資源命名管理系統的研究或實作者少，事實上，這兩件事一定要和 metadata 一起被考慮，如此才能滿足資源的擴充、彈性有效的系統管理、及聯合目錄的建立之需。

玖、結語

由於近年來大量累積的數位資源，使得針對網路資源進行良好的管理與利用，成爲有效管理這些數位資源最重要的課題。至於要便利的運用這些寶貴的資料，則有賴於良好的蒐尋機制來給予輔助，而這所謂的蒐尋機制包含了詮釋資料、數位資源識別碼以及命名管理系統，其中數位資源識別碼和命名管理系統也正是本文討論的主要議題。一旦有了數位資源唯一的識別碼，數位物件的版權才得以控制；而若數位資源移位時，則需透過彈性的命名管理系統，才可找到該數位物件。本文介紹了數位資源命名管理系統完整的語法與架構，同時也藉機探討此命名管理系統之意義、功能、命名空間、架構、服務與安全機制。希望未來國內在建立數位圖書館相關系統時，能夠考量這些相關議題，才能夠對數位資源作良好的管理與妥善的運用。

【原刊載於檔案管理局出版之檔案季刊（九十一年九月）第一卷第三期】

參考資料：

[1] The Unicode Consortium, "The Unicode Standard, Version 2.0", Addison-Wesley Developers Press, 1996, ISBN 0-201-48345-9

[2] P. Mockapetris, "DOMAIN NAMES - CONCEPTS AND FACILITIES", RFC1034, November 1987, http://www.ietf.org/rfc/rfc1034.txt

[3] P. Mockapetris, "DOMAIN NAMES - IMPLEMENTATION AND SPECIFICATION", RFC1035, November 1987, http://www.ietf.org/rfc/rfc1035.txt

[4] Berners-Lee, T., Masinter, L., McCahill, M., et al., "Uniform Resource Locators (URL)", RFC1738, December 1994, http://www.ietf.org/rfc/rfc1738.txt

[5] Yergeau, Francois, "UTF-8, A Transformation Format for Unicode and ISO10646", RFC2044, October 1996, http://www.ietf.org/rfc/rfc2044.txt

[6] ITU-T Rec. X.500, "The Directory: Overview of Concepts, Models, and Services", 1993.

[7] D W Chadwick, "Understanding X.500 - The Directory", Chapman & Hall ISBN: 0-412-43020-7.

[8] Wahl, M., Howes, T., and S. Kille, "Lightweight Directory Access Protocol (v3)", RFC 2251, December 1997, http://www.ietf.org/rfc/rfc2251.txt

[9] Sollins, K., and L. Masinter, "Functional Requirements for Uniform Resource Names", RFC 1737, December 1994, http://www.ietf.org/rfc/rfc1737.txt

[10] Sollins, K. "Architectural Principles of Uniform Resource Name Resolution", RFC 2276, January 1998, http://www.ietf.org/rfc/rfc2276.txt

[11] Sun, S., Reiley, S., Lannom, L., "Handle System Namespace and Service Definition", IETF draft, http://www.ietf.org/internet-drafts/draft-sun-handle-system-def-04.txt

[12] Sun, S., Reiley, S., Lannom, L., "Handle System Protocol (ver 2.0) Specification", IETF draft, http://www.ietf.org/internet-drafts/draft-sun-handle-system-protocol-01.txt

[13] D-Lib Magazine, http://www.dlib.org/

[14] Sam X. Sun, "Internationalization of the Handle System - A Persistent Global Name Service", Proceeding of 12th International Unicode Conference, April, 1998, http://www.cnri.reston.va.us/unicode-paper.ps

[15] D Goodman, C Robbins, "Understanding LDAP & X.500", August 1997.

[16] Deutsch P., Schoultz R., Faltstrom P., and C. Weider, "Architecture of the Whois++ service", RFC 1835, August 1995, http://www.ietf.org/rfc/rfc1835.txt?number=1835

[17] Weider, C., J. Fullton, and S. Spero, "Architecture of the Whois++ Index Service", RFC 1913, February 1996, http://www.ietf.org/rfc/rfc1914.txt

[18] Kahn, Robert and Wilensky, Robert. "A Framework for Distributed Digital Object Services", May, 1995, http://www.cnri.reston.va.us/tmp_hp/k-w.html

[19] The Networked Computer Science Technical Reports Library (NCSTRL) http://www.ncstrl.org/

[20] IETF Uniform Resource Names (URN) Working Group, 30 Mar 01, http://www.ietf.org/html.charters/urn-charter.html

[21] The Library of Congress, National Digital Library Program, “Identifiers for Digital Resources”, 13 Aug 1996, http://lcweb2.loc.gov/ammem/award/docs/identifiers.html (2002/4/12).