DOCUMENT RESUME

ED 450 775   IR 058 031

AUTHOR   Lannom, Laurence
TITLE   Handle System Overview.
PUB DATE   2000-08-00
NOTE   8p.; In: IFLA Council and General Conference: Conference
                Proceedings (66th, Jerusalem, Israel, August 13-18, 2000);
                see IR 057 981.
AVAILABLE FROM For full text:
                http://www.ifla.org/IV/ifla66/papers/032-82e.htm.
PUB TYPE   Reports - Descriptive (141) -- Speeches/Meeting Papers (150)
EDRS PRICE   MF01/PC01 Plus Postage.
DESCRIPTORS   Access to Information; *Computer System Design; Electronic
                Libraries; Electronic Publishing; *Information Management;
                Information Networks
IDENTIFIERS   Digital Data; *Distributed Computing; *Names

ABSTRACT
        The Handle System is a distributed computer system that
stores names, or handles, of digital items and can quickly resolve those
names into the information necessary to locate and access the items. It was
designed by the Corporation for National Research Initiatives as a general
purpose global system for the reliable management of information on networks
such as the Internet over long periods of time and is currently in use in a
number of production and prototype projects. This paper provides a brief
history and technical overview of the Handle System, including the need for a
general purpose naming system, design objectives, handle name space, and
system architecture. Issues in its use in the digital library and electronic
publishing arenas are identified, including multiple resolution and the
appropriate copy problem. (Contains 10 references.) (Author/MES)

ED 450 775

IFLANET

I F L A

66th IFLA Council and General Conference

Jerusalem, Israel, 13-18 August

Conference
Proceedings

Handle System Overview

Laurence Lannom
Corporation for National Research Initiatives
E-mail: llannom@cnri.reston.va.us

Abstract

The Handle System is a distributed computer system which stores names, or handles, of digital items and which can quickly resolve those names into the information necessary to locate and access the items. It was designed by CNRI as a general purpose global system for the reliable management of information on networks such as the Internet over long periods of time and is currently in use in a number of production and prototype projects. This talk will provide a brief history and technical overview and identify issues in its use in the digital library and electronic publishing arenas.

Paper

Introduction

The Handle System® is a general purpose distributed information system designed to provide an efficient, extensible, and secured global name service for use on networks such as the Internet. The Handle System includes an open set of protocols, a namespace, and a reference implementation of the protocols. The protocols enable a distributed computer system to store names, or handles, of digital resources and resolve those handles into the information necessary to locate, access, and otherwise make use of the resources. These associated values can be changed as needed to reflect the current state of the identified resource without changing the handle, thus allowing the name of the item to persist over

IR058031

2/13/01 8:27 AM

changes of location and other current state information. Each handle may have its own administrator(s) and administration can be done in a distributed environment. The name-to-value bindings may also be secured, allowing handles to be used in trust management applications.

This paper covers the evolution of the Handle System, including its origins and current use, provides a technical overview of the system, and concludes with a discussion of some of the more interesting and important issues which are currently being addressed in its use in digital library and electronic publishing applications.

Evolution

The Handle System was originally conceived and developed at CNRI as part of the Computer Science Technical Reports (CSTR) project, funded by the Defense Advanced Projects Agency (DARPA) under Grant No. MDA-972-92-J-1029. One aspect of this early digital library project, which was also a major factor in the evolution of the Networked Computer Science Technical Reference Library (NCSTRL - see http://cs-tr.cs.cornell.edu/) and related activities, was to develop a framework for the underlying infrastructure of digital libraries. It is described in a paper by Robert Kahn and Robert Wilensky[1]. Subsequent work on the Handle System has been supported in part by the Defense Advanced Research Projects Agency under Grant No. MDA972-92-J-1029.

Early adopters of the Handle System have included the Library of Congress, the Defense Technical Information Center (DTIC), the International DOI Foundation (IDF), and, most recently, the CrossRef service offered by the newly formed Publishers International Linking Association, Inc. (PILA). Feedback from these organizations as well as NCSTRL, other digital library projects, and related IETF efforts have all contributed to the evolution and deployment of the Handle System. Current status and available software, both client and server, can be found at http://www.handle.net/. This web site, as well as the DOI site (http://www.doi.org) also provide many examples of the use of handles.

The Handle System has evolved within the digital library and electronic publishing communities, particularly as part of the continuing move of scholarly and technical publication from paper-centric to digital-centric systems, but it was conceived and built as the naming component of an overarching digital object architecture, as described in Kahn/Wilensky[1] and subsequent papers[2, 3, 4]. It has potential application not only beyond the early adopters such as the IDF, DTIC, and LC, but also well beyond the digital library area. As a general purpose indirection system that resolves identifiers into state information, the Handle System can be used to advantage in any dynamic network environment as part of the overall process of managing digital objects. Interest has been expressed by organizations in application areas as diverse as telephony (linking individuals with multiple phone numbers, 'telephone number for life', etc.), and crisis management (resource tracking). Any given application area would have to build its own tools and approaches, but the Handle System, especially as part of the larger digital object architecture referenced above, can serve as an information management substrate for a wide variety of application areas.

Technical Overview

Need for a General Purpose Naming System. The need for a general purpose naming system has increased with Internet growth. While there are existing services and protocols that cover some of the functionality proposed in the

Handle System, and while we make no claim that the Handle System is the only such service that is now or ever will be needed, we do believe that the Handle System provides needed functionality that is not otherwise available.

There are several services that are in use today to provide name service for Internet resources, of which the Domain Name System (DNS)[5, 6] is the most widely used. DNS is designed "to provide a mechanism for naming resources in such a way that the names are mappable into IP addresses and are usable in different hosts, networks, protocol families, internets, and administrative organizations"[6]. The growth of the Internet has increased demands for various extensions to DNS, and even its use as a general purpose resource naming system, but its importance in basic network routing has led to great caution in implementing such extensions and a general conclusion that DNS is not the place to look for general purpose resource naming. An additional factor which argues against using DNS as a general purpose naming system is the DNS administrative model. DNS names are typically managed by the network administrator(s) at the DNS zone level, with no provision for a per name administrative structure, and no facilities for anyone other than network administrators to create or manage names. This is appropriate for domain name administration but less so for general purpose resource name administration. The Handle System has been designed from the start to serve as a naming system for very large numbers of entities and to allow administration at the name level.

URLs (Uniform Resource Locators)[7] allow certain Internet resources to be named as a combination of a DNS name and local name. The local name may be a local file path, or a reference to some local service, e.g. a cgi-bin script. This combination of DNS name and local name provides a flexible administrative model for naming and managing individual Internet resources. There are, however, several key limitations. Most URL schemes (e.g., http) are defined for resolution service only. Any URL administration has to be done either at the local host, or via some other network service such as NFS. Using a URL as a name typically ties the Internet resource to its current network location, and to its local file path when the file path is part of the URL. When the resource moves from one location to another, for whatever reason, the URL breaks.

The Handle System is designed to overcome these limitations and to add significant increased functionality. Specifically, the Handle System is designed with the following objectives:

Uniqueness. Every handle is globally unique, within the Handle System.

Persistence. A handle is not derived in any way from the entity which it names, but is assigned to it independently. While an existing name, or even a mnemonic, may be included in a handle for convenience, the only operational connection between a handle and the entity it names is maintained within the Handle System. This of course does not guarantee persistence, which is a function of administrative care, but it does allow the same name to persist over changes of location, ownership, and other state conditions. For example, when a named resource moves from one location to another, the handle may be kept valid by updating its value to reflect the new location.

Multiple Instances. A single handle can refer to multiple instances of a resource, at different and possibly changing locations in a network. Applications can take advantage of this to increase performance and reliability. For example, a network service may define multiple entry points

2/13/01 8:27 AM

for its service with a single handle name and so distribute the service load.

Extensible Namespace. Existing local namespaces may join the handle namespace by acquiring a unique handle naming authority. This allows local namespaces to be introduced into a global context while avoiding conflict with existing namespaces. Use of naming authorities also allows delegation of service, both resolution and administration, to a local handle service.

International Support. The handle namespace is based on Unicode 2.0[8], which includes most of the characters currently used around the world, facilitating the use of the system in any native environment. The handle protocol mandates UTF-8[9] as the encoding used for handles.

Distributed Service Model. The Handle System defines a hierarchical service model such that any local handle namespace may be serviced either by a corresponding local handle service or by the global service or by both. The global service, known as the Global Handle Registry[TM], can be used to dispatch any handle service request to the responsible local handle service. The distributed service model allows replication of any given service into multiple service sites and each service site may further distribute its service into a cluster of individual servers. (Note that local here refers only to namespace and administrative concerns. A local handle service could in fact have many service sites distributed across the Internet.)

Secured Name Service. The handle protocol allows handle servers to authenticate their clients and to provide data integrity service upon client request. Public key and/or secret key cryptography may be used. This may be used to prevent eavesdroppers from forging client requests or tampering with server responses.

Distributed Administration Service. Each handle may define its own administrator(s) or administrative group(s). This, combined with the Handle System authentication protocol, allows handles to be managed securely over the public network by authorized administrators at any network location.

Efficient Resolution Service. The handle protocol is designed to allow highly efficient name resolution performance. To avoid resolution being affected by computationally costly administration service, separate service interfaces (i.e., server processes and their associated communication ports) for handle name resolution and administration may be defined by any handle service.

Handle Name Space

Every handle consists of two parts: its naming authority, otherwise known as its prefix, and a unique local name under the naming authority, otherwise known as its suffix. The naming authority and local name are separated by the ASCII character "/" (octet 0x2F). A handle may thus be defined as < Handle> ::= < Handle Naming Authority> "/" < Handle Local Name> For example, " 10. 10.1045/march2000-owen " is a handle for an article published in the D-LIB magazine [10]. It is defined under the Handle Naming Authority "10.1045", and its Handle Local Name is " march2000-owen ".

Handle System Architecture

The Handle System has a two-level hierarchical service model. The top level consists of a single global service, known as the Global Handle Registry. The lower level consists of all other handle services, which are generically known as local handle services. The global service is a handle service like any other and can be used to manage any handle namespace. It is unique among handle services only in that it provides the service used to manage the namespace of handle naming authorities, all of which are managed as handles. The state information of these naming authority handles is the service information that clients can use to access and utilize associated local services. The local handle service layer consists of all local handle services managing all handles under the relevant naming authorities, providing both resolution and administration services for these local names. Local services are intended to be hosted by organizations with administrative responsibility for the handles within the service or acting on behalf of the responsible organizations. The most convenient way to define local namespaces, and the most likely way to optimize overall Handle System performance, is by naming authority and it is anticipated that in most cases all handles under a given naming authority will be maintained by one service. This is not required, however, and it is possible for handles under a single naming authority to be split among multiple handle services. Handle services may be responsible for more than one naming authority. Another way of stating all of this is that the relation of handle naming authorities and handle services is allowed to be many-to-many in both directions, but that the relationship of naming authority to handle service is most likely to be one-to-one and that the relationship of handle service to naming authority is likely to be one-to-many.

A second important component of Handle System architecture is distribution. The Handle System as a whole consists of a number of individual handle services, each of which consists of one or more handle service sites, where each site replicates the complete individual handle service, at least for the purposes of handle resolution. Each handle service site in turn consists of one or more handle servers. There are no design limits on the total number of handle services which constitute the Handle System, there are no design limits on the number of sites which make up each service, and there are no limits on the number of servers which make up each site. Replication by site, within a service, does not require that each site contain the same number of servers, that is, while each site will have the same replicated set of handles, each site may allocate that set of handles across a different number of handle servers. This distributed approach is intended to aid scalability and to mitigate problems of single point failure.

Current Issues

A number of interesting and important issues have come to the fore over the last few years as a result of early use of the Handle System in library and publishing environments. Two particularly compelling issues are multiple resolution and the appropriate copy problem.

> **Multiple resolution.** The Handle System has been designed to resolve handles into one or more pieces of current state data, each of which is fundamentally a type-value pair, e.g., a URL for content or an email address for contact information. The ability to resolve a single identifier into multiple typed values has several clear potential benefits. One is to identify multiple network locations for a single named entity, which has great potential for increasing network performance and robustness. A second potential benefit is go beyond the obvious single level of indirection for content and to use the identifier to link to other types of relevant current

data, such as descriptive metadata, rights information, and so forth. The basic facility exists in the Handle System now, but there is not yet much use of this facility in client applications. As of the writing of this article (March 2000), however, a number of experiments and prototypes employing multiple resolution are under way or being discussed.

Appropriate copy. A second pressing issue has become known as the 'appropriate copy' problem. While there are many benefits to having a reliable global resolution system for globally unique identifiers, one problem is that all resolution questions yield the same answer and the same answer may not be appropriate in all cases. Consider the situation of an institution or enterprise holding a local copy of an electronic publication or other kind of digital object which is identified by a DOI or other kind of handle or global identifier. Unless the global resolution system contains all information on all local copies, arguably a poor idea and in any event one that seems unlikely, resolving the identifier in the global system will not yield a pointer to the local copy. This is clearly an issue of concern to both libraries and publishers and one that has generated a great deal of discussion over recent years. Whether this problem is most effectively solved with some local library system, some special purpose boundary layer mechanism, such as a proxy/cache, or in some other fashion remains to be seen. CNRI has been in discussion with the Digital Library Federation (DLF), individual publishers, the IDF, and CrossRef on this issue and it seems clear that one or more prototypes will be attempted in the coming months and years.

Conclusion

Deployment of the Handle System to date has served to confirm the basic design concepts, as described in this article, and significant progress has been made in understanding the complexities and issues involved in designing effective digital object naming and resolution systems. It is a large problem space, however, and a great deal of work remains in this area as well as many others as we attempt to navigate from the current world to one in which the primary sources of information are digital objects on networks.

References

1. Kahn, Robert and Wilensky, Robert. "A Framework for Distributed Digital Object Services", May, 1995.
   (http://www.cnri.reston.va.us/k-w.html)
2. Arms, William Y., Christophe Blanchi, Edward A. Overly. "An Architecture for Information in Digital Libraries, D-Lib Magazine, February 1997".
   (http://www.dlib.org/dlib/february97/cnri/02arms1.html)
3. 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)
4. Payette, Sandra Payette, Christophe Blanchi, Carl Lagoze, Edward A. Overly. "Interoperability for Digital Objects and Repositories".
   (http://www.dlib.org/dlib/may99/payette/05payette.html)
5. P. Mockapetris, "DOMAIN NAMES - CONCEPTS AND FACILITIES", RFC1034, November 1987.
   (http://info.internet.isi.edu:80/in-notes/rfc/files/rfc1034.txt)
6. P. Mockapetris, "DOMAIN NAMES - IMPLEMENTATION AND

SPECIFICATION", RFC1035, November 1987.
(http://info.internet.isi.edu:80/in-notes/rfc/files/rfc1035.txt)

7. Berners-Lee, T., Masinter, L., McCahill, M., et al., "Uniform Resource
Locators (URL)", RFC1738, December 1994.
(http://info.internet.isi.edu:80/in-notes/rfc/files/rfc1738.txt)

8. The Unicode Consortium, "The Unicode Standard, Version 2.0",
Addison-Wesley Developers Press, 1996. ISBN 0-201-48345-9.

9. Yergeau, Francois, "UTF-8, A Transform Format for Unicode and
ISO10646", RFC2044, October 1996.
(http://info.internet.isi.edu:80/in-notes/rfc/files/rfc2044.txt)

10. D-Lib Magazine.
(http://www.dlib.org)

Latest Revision: *May 12, 2000*

2/13/01 8:27 AM

NOTICE

REPRODUCTION BASIS