XEROX

Xerox System Integration
Standard

Clearinghouse Entry Formats

XEROX

Clearinghouse Entry Formats

Notice

This *Xerox System Integration Standard* describes the Clearinghouse Directory Service entry formats.

1. This standard includes subject matter relating to patent(s) of Xerox Corporation. No license under such patent(s) is granted by implication, estoppel, or otherwise, as a result of publication of this specification.

2. This standard is furnished for informational purposes only. Xerox does not warrant or represent that this standard or any products made in conformance with it will work in the intended manner or be compatible with other products in a network system. Xerox does not assume any responsibility or liability for any errors or inaccuracies that this document may contain, nor have any liabilities or obligations for any damages, including but not limited to special, indirect, or consequential damages, arising out of or in connection with the use of this document in any way.

3. No representations or warranties are made that this specification, or anything made in accordance with it, is or will be free of any proprietary rights of third parties.

Preface

This document is one of a family of publications that describes the network protocols underlying Xerox Network Systems (XNS).

Xerox Network Systems comprise a variety of digital processors interconnected by means of a variety of transmission media. System elements communicate both to transmit information between users and to economically share resources. For system elements to communicate with one another, certain standard protocols must be observed.

Comments and suggestions on this document and its use are encouraged. Please address communications to:

Xerox Corporation
Office Systems Division
Network Systems Administration Office
3450 Hillview Avenue
Palo Alto, California 94304

Table of contents

Appendices

1

Introduction

This document defines certain Clearinghouse™ Directory Service property types and the structure of their associated entries. The Clearinghouse entries are defined in terms of Courier [2] data types. This document is a supplement to the Clearinghouse Protocol standard [1] and assumes that the reader is familiar with it.

1.1 Purpose

The structure of the values of Clearinghouse entries are independent of the facilities that manipulate and search them. In addition to specifying the structure of the entry values, this document describes how the values are intended to be used. This information is required for sharing of Clearinghouse entries. Clearinghouse property numbers are administered as described in Appendix B.

1.2 Documentation conventions

Throughout this document, special fonts are used to depict Courier text instead of using quote marks or other delimiters. This convention also aids the eye in discriminating between Courier text and the exposition. Items in THIS FONT indicate elements of the Courier language and are almost always in upper case. This font indicates items that are defined using the Courier language.

Identifiers that are defined in this document (as opposed to being defined by Courier) will have their first letter capitalized if they are the name of a type; identifiers with a lowercase first letter are usually the names of variables.

An object stored in the Clearinghouse is really a logical collection of data which pertains to an object such as a Print Service, a File Service, or a user. This collection of data is called an entry and consists of a small constellation of property/value pairs. The *entry format* corresponding to a given object type is simply the definition of this constellation of properties and their associated values. The following notation is used to indicate the entry format of an object:

Object-type: { (*property*$_1$, *value*$_1$), ..., (*property*$_n$, *value*$_n$)}

where *property*$_i$ will be a name defined to be a **Clearinghouse.Property**, and *value*$_i$ will be either a Courier expression that defines a type, or the term **Group** indicating that it is a

group-type property. This notation is used to emphasize the notion that an object is an unordered set of property/value pairs.

1.3   Document organization

Section 2 of this document defines the formats of the standard Clearinghouse entries. Appendix A lists other documents which supplement the specification. Appendix B explains how to acquire a block of property names.

2

Entry formats

The set of Clearinghouse entry types is determined by the network resources and distributed objects that are present in the internetwork. New entries are suggested by new applications. The criteria used to add new entry types are: is this a new service that requires a property structure not currently defined?; or is there a reasonable need to enumerate all entries of this new type within some scope, such as within a domain? *Entry* is used to mean the collection of property values that an object may have. An *entry format* is a convention defining the property list describing objects of a given type.

This chapter defines the standard properties that may be assigned to a Clearinghouse object and defines their structure. For each property, the property number is defined and a description given of the property value. If a property is an item-type property, the structure of the item value is defined using Courier notation.

The Clearinghouse associates a *name* with a *property list*, a list of zero or more property/ value pairs. A *property* is simply a number. Each Clearinghouse property determines the format of the value associated with it. A value is either an *item* (a variable length data block, uninterpreted by the Clearinghouse), or a *group* (a list of names that the Clearinghouse can manipulate directly through the group operations). The Clearinghouse knows the difference between items and groups and will not allow item operations on groups or vice versa. The Clearinghouse supports special group operations in order to provide a convenient name list facility for mailing and access control. The elements, or *members*, of a group stored in the Clearinghouse have the form of Clearinghouse names, although the Clearinghouse makes no attempt to guarantee that these names are actually registered in its database.

The type (item or group) of a value is determined by the operation used to create that value, not by the property used to store it. The usage of properties is strictly a matter of convention; it is not something which the Clearinghouse enforces.

The specific properties associated with the various object types fall into three general categories. *Primary properties* are associated one-to-one with specific object types, and are particularly useful for enumerating all objects of a given type (e.g., all Print Services, or all users). The only value associated with a primary property is a human-readable descriptive string, so in many cases other object-specific properties, called *secondary properties*, are defined to contain additional type-specific information. Finally, *generic properties* are used to describe aspects of objects that are independent of their specific type (for example, a network address, which is an attribute shared by many different object types). Note that

the categorization of properties as primary, secondary, and generic is strictly a matter of convention, and is neither recognized nor enforced by the Clearinghouse.

This document describes a subset of the complete set of Clearinghouse entry formats—specifically, those entries needed by clients of protocols that have been (or soon will be) standardized. These protocols are: Authentication, Clearinghouse, Filing, Mailing, and Printing. For each property, the property number is defined and a description given of the value of each item property.

The entry specifications here should not be taken as a guarantee that objects of the defined types will have only the properties specified. All client software should be robust in the face of additional unexpected properties.

2.1   Generic properties

Generic properties describe aspects of registered objects that are independent of their precise object type, and are thus sharable among different types of objects. The entry format specifications in the following sections make use of the generic properties described in this section. They also make use of the common value type **Description** in each primary property:

Description: TYPE = STRING;

A **Description** contains descriptive text that is by convention limited to a maximum of 100 bytes (not characters), and is conventionally used to indicate the physical location of the service.

2.1.1 AddressList property

An **addressList** property is used to register the network address of an object (e.g., a service) that is addressable via the internetwork. Since it is possible for a single processor to be attached to several different Ethernet networks, an **AddressListValue** is actually a SEQUENCE OF **NetworkAddress** (see the Clearinghouse Protocol [1]).

addressList: Clearinghouse.Property = 4;

AddressListValue: TYPE = **Clearinghouse.NetworkAddressList;**

Most of the time an **AddressListValue** will contain only one network address. When multiple addresses are present, they will differ only in their "network" field.

2.1.2 MailBoxes property

A **mailboxes** property lists the names of the Mail Service(s) on which the mailbox(es) of the object reside. Any object possessing a **mailboxes** property is a valid mail recipient. The most obvious example of a mail recipient is a user, but any other entity (e.g., a service) can be given a **mailboxes** property and thereby receive mail.

mailboxes: Clearinghouse.Property = 31;

```
MailboxesValue: TYPE = RECORD[
    time: Time.Time,
    mailService: ARRAY OF Clearinghouse.Name];
```

The association of a **mailboxes** property with a mail recipient is intended to be performed by the Mail Service itself, as a side-effect of the creation of an actual mailbox. It is therefore neither necessary nor appropriate for Mail Service clients to attempt to manipulate this property. The **time** field is the time stamp of when this value was created and is in the standard time format [3].

2.1.3 Authentication levels supported

A service may support client access using either or both of the Authentication levels, simple or strong. This information is conveyed with the **authenticationLevel** property:

authenticationLevel: Clearinghouse.Property = 8;

```
AuthenticationLevelValue: TYPE = RECORD[
    simpleSupported, strongSupported: BOOLEAN];
```

A value of TRUE for each component of the record indicates that the corresponding level of authentication is supported by the service. The level of authentication used by the service to authenticate itself when accessing other services is independent of this property.

2.2   Clearinghouse Service entry

A Clearinghouse Service entry provides the information necessary to access the Clearinghouse Service for lookups, updates, and other operations, via the Clearinghouse Protocol.

```
ClearinghouseService: {
    (clearinghouseService, Description),
    (addressList, AddressListValue),
    (mailboxes, MailboxesValue),
    (authenticationLevel, AuthenticationLevelValue)}
```

clearinghouseService: Clearinghouse.Property = 10021;

2.3   File Service entry

A File Service entry provides the information needed to access data stored on the default volume of the named File Service via the Filing Protocol.

```
FileService: {
    (fileService, Description),
    (addressList, AddressListValue),
    (authenticationLevel, AuthenticationLevelValue)}
```

fileService: Clearinghouse.Property = 10000;

2.4 Mail Service entry

A Mail Service entry provides the information necessary to access the Mail Service for posting and delivery of mail, via the MailTransport and Inbasket Protocols.

```
MailService: {
    (mailService, Description),
    (addressList, AddressListValue),
    (authenticationLevel, AuthenticationLevelValue)}

mailService: Clearinghouse.Property = 10004;
```

2.5 Print Service entry

A Print Service entry provides the information necessary to access the Print Service for submission of Interpress™ Electronic Printing Standard masters via the Printing Protocol.

```
PrintService: {
    (printService, Description),
    (addressList, AddressListValue),
    (authenticationLevel, AuthenticationLevelValue)}

printService: Clearinghouse.Property = 10001;
```

2.6 User entry

A user entry describes a human user for purposes of authentication, mail delivery, and so forth.

```
User: {
    (user, Description),
    (userData, UserDataValue),
    (mailboxes, MailboxesValue)}

user: Clearinghouse.Property = 10003;

userData: Clearinghouse.Property = 20000;

UserDataValue: TYPE = RECORD[
    lastNameIndex: CARDINAL,
    fileService: Clearinghouse.Name];
```

lastNameIndex is the byte offset in this user's Clearinghouse name which is the first character of the user's last name. It is used to help clients to sort user names. The **fileService** component names the user's home File Service (that is, where the user's 8010 desktop is stored).

2.7 UserGroup entry

A **userGroup** entry represents a logical grouping of users for access control or mail delivery purposes. The addition of a user group name to an access control list (e.g., on a file

drawer) grants uniform access rights to all members of the group. The use of a group name as a recipient of an electronic mail message results in delivery of the message to all members of the group.

```
UserGroup: {
    (userGroup, Description),
    (members, Group)}
```

userGroup: Clearinghouse.Property = 10022;

members: Clearinghouse.Property = 3;

A

Appendix A
References

The following documents supplement this specification of Clearinghouse entry formats.

[1] Xerox Corporation. Clearinghouse Protocol. Xerox System Integration Standard. Stamford, Connecticut; 1983 August; XSIS 078308.
This reference defines the Clearinghouse Protocol which manipulates the entries defined in this document.

[2] Xerox Corporation. Courier: The Remote Procedure Call Protocol. Xerox System Integration Standard. Stamford, Connecticut; 1981 December; XSIS 038112.
This reference defines the Courier language, in terms of which the formats are defined.

[3] Xerox Corporation. Time Protocol. Xerox System Integration Standard. Stamford, Connecticut; 1982 October; XSIS 088210.
This reference defines the format of those fields whose values are time.

B

Appendix B
Type assignment procedures

As stated in this document, property types are assigned 32-bit numbers that are unique throughout the distributed system. The number space is administered by Xerox Corporation. To obtain a block of numbers, submit a written request to:

Xerox Corporation
Office Systems Division
Network Systems Administration Office
3450 Hillview Avenue
Palo Alto, California 94304

Implementors are encouraged to apply for unique blocks of numbers for their particular applications. Uniqueness allows systems to freely interconnect without having to worry about overlapping values for critical fields.

Property numbers should be used with economy as the total number of blocks is limited. If an implementation is using a large quantity of these property type numbers, the designer has probably misunderstood their purpose.

XEROX

XEROX

Clearinghouse Entry Formats

Xerox System Integration Standard

XSIS 168404

610P72588