

Network Working Group  
Request for Comments: 2655  
Category: Experimental

T. Hardie  
Equinix  
M. Bowman  
Transarc  
D. Hardy  
Netscape  
M. Schwartz  
Affinia, Inc.  
D. Wessels  
NLNR  
August 1999

## CIP Index Object Format for SOIF Objects

### Status of this Memo

This memo defines an Experimental Protocol for the Internet community. It does not specify an Internet standard of any kind. Discussion and suggestions for improvement are requested. Distribution of this memo is unlimited.

### Copyright Notice

Copyright (C) The Internet Society (1999). All Rights Reserved.

### 1. Abstract

The Common Indexing Protocol (CIP) allows servers to form a referral mesh for query handling by defining a mechanism by which cooperating servers exchange hints about the searchable indices they maintain. The structure and transport of CIP are described in (Ref. 1), as are general rules for the definition of index object types. This document describes SOIF, the Summary Object Interchange Format, as an index object type in the context of the CIP framework. SOIF is a machine-readable syntax for transmitting structured summary objects, currently used primarily in the context of the World Wide Web.

Query referral has often been dismissed as an ineffective strategy for handling searches of Web resources, and Web resources certainly present challenges not present in structured directory services like Rwhois. In situations where a keyword-based free text search is desired, query referral is not likely to be effective because the query will probably be routed to every server participating in the referral mesh. Where a search can be limited by reference to a specific resource attribute, however, query referral is an effective tool. SOIF can be used to create such a known-attribute query mesh because it provides a method for associating attributes with net-addressable resources.

## 1.1 History

SOIF was first defined by the Harvest project [Ref 2.] in January 1994. SOIF was derived from a combination of the Internet Anonymous FTP Archives IETF Working Group (IAFA) templates [Ref 3.] and the BibTeX bibliography format [Ref 4.]. The combination was originally noted for its advantages of providing a convenient and intuitive way for delimiting objects within a stream, and setting apart the URL for easy object access or invocation, while still preserving compatibility with IAFA templates.

Mic Bowman, Darren Hardy, Mike Schwartz, and Duane Wessels each contributed to the creation of the SOIF format as part of the Harvest Project; later work took place as part of the FIND working group.

## 2. Name

The index object described below will have the MIME type of application/index.obj.HARVEST-SOIF-1.

## 3. Payload Format

Each summary object has 3 fundamental components: a template type, a URL, and zero or more ATTRIBUTE-VALUE pairs. Because the VALUES in the ATTRIBUTE-VALUE pairs may contain arbitrary data (cf. Section 3.5), SOIF objects should be encoded in Base64 unless the template type unambiguously establishes that the VALUES do not contain binary data.

### 3.1 Template Type

The Template type is used to identify the set of ATTRIBUTES contained within a particular SOIF object. SOIF does not define the template types themselves; it only provides a way to associate the summary object with a predefined template type name. Template types may be registered or unregistered. Unregistered template types provide an indication of available ATTRIBUTE-VALUE pairs, but these may vary both according to the original resource and the method by which the summary object was generated. Registered template types must refer to a formally specified description of all mandatory and optional ATTRIBUTE-VALUE pairs available for that type. See [10] for a description of the process of registering template types with the IANA.

Historically, the template types used by SOIF were derived from IAFA template types (Ref. 3). SOIF objects generated by the Harvest system have a "FILE" template type; in current practice this is the most common template type. The "FILE" template type is a generic template

type meant to handle a large variety of web-based resources. No formal specification of it is available, though a list of ATTRIBUTE-VALUE pairs common to the "FILE" template type is found in Appendix A. "DOCUMENT" and "OBJECT" are other generic template-types.

The use of unregistered template types obviously presents some problems to the correct operation of query referral. Two efforts have been mounted to allow peer-to-peer agreement on the association of template types with specific attribute sets: Netscape's RDM (Ref. 6) and the STARTS project (Ref. 7). Initially, CIP meshes based on systems which use unregistered template types may need to use these or similar methods to associate template types with specific attribute sets.

Mesh operators are strongly encouraged, however, to migrate to registered template types as soon as is practical. Registered template types allow CIP meshes to derive the definitions of attributes, which enables multiple-language interfaces to the base attributes. In addition, registered template types allow CIP meshes and other users of SOIF to establish the permitted data types and encodings of the VALUES associated with each ATTRIBUTE. This makes deriving the appropriate matching semantics for a particular VALUE much more straightforward and eliminates the limitations of the default octet-by-octet matching (cf. Section 4.).

### 3.2 URL

Uniform Resource Locators (URLs) (Ref 5.) are used by SOIF as object IDENTIFIERS. SOIF associates its summary objects with net-addressable resources by using the URL by which the resource was addressed as the initial field of the object body. See section 3.4 for the formal grammar associated with SOIF objects.

This association allows the same resource to have multiple summary objects, differentiated only by the URL by which the resource was accessed. This possibility does not, however, impact the usability of the URL as an object IDENTIFIER. Furthermore, since it can be argued that the net address is a salient part of the metadata, there may be compensating benefits to using the URL as an object IDENTIFIER.

As noted in Appendix A, the Harvest project used several additional identity attributes ("Gatherer-Name", "Gatherer-Host", "Gatherer-Port" and "Gatherer-Version") to further identify the provenance of a particular object. Within the context of CIP, it may be useful to identify the base sources of particular index objects; see Appendix B for one example of how a SOIF-based CIP hint could use the base source URL.

### 3.3 ATTRIBUTE-VALUE pairs.

Each summary object has zero or more ATTRIBUTE-VALUE pairs, which contain metadata about the net-addressable resource referenced by the URL. Pairs are composed of an ATTRIBUTE IDENTIFIER, the length of the VALUE, a delimiter, and the VALUE. It should be stressed that ATTRIBUTE VALUE pairs are not CR/LF terminated, but parsed according to grammar set out in section 3.4. In the examples in Section 3.6 and in many other representations of SOIF objects, ATTRIBUTE-VALUE pairs are represented on individual lines to enhance readability. VALUES may contain CR/LF, however, and implementors must be careful to parse the full VALUE. Implementors of SOIF parsers MUST ignore <CR>, <LF>, <TAB>, <SPACE>, or other whitespace found between the VALUE of an ATTRIBUTE-VALUE pair and the ATTRIBUTE-IDENTIFIER of the subsequent pair.

The SOIF syntax does not explicitly allow for a single ATTRIBUTE to have multiple VALUES. To handle multiple VALUES for the same ATTRIBUTE, SOIF uses an ATTRIBUTE naming convention; a hyphen and positive integer are appended to the ATTRIBUTE name to create an ATTRIBUTE IDENTIFIER VALUE associated with a specific ATTRIBUTE. For example, the ATTRIBUTE IDENTIFIERS "Author-1", "Author-2", and "Author-3" can be used to represent three VALUES associated with the ATTRIBUTE "Author" where a specific resource has three authors. See section 4 for the implications of this strategy on matching semantics.

### 3.4 SOIF Grammar

The SOIF syntax is defined by the following grammar:

```

SOIF          ::= OBJECT SOIF |
                  OBJECT
OBJECT         ::= @ TEMPLATE-TYPE { URL ATTRIBUTE-LIST }
TEMPLATE-TYPE ::= IDENTIFIER
ATTRIBUTE-LIST ::= ATTRIBUTE ATTRIBUTE-LIST |
                  ATTRIBUTE |
                  NULL
ATTRIBUTE      ::= IDENTIFIER {VALUE-SIZE} DELIMITER VALUE
URL            ::= RFC1738-URL-Syntax | "-"
IDENTIFIER     ::= ALPHA-NUMERIC-STRING
VALUE          ::= ARBITRARY-DATA
VALUE-SIZE     ::= NUMERIC-STRING
DELIMITER      ::= " :<TAB> "
```

### 3.5 Grammar Description

#### URL

a Uniform Resource Locator encoded in the syntax defined by RFC 1738 [3]. If the summary object has no URL associated with it, then a Latin-1 hyphen (octal \055) is used instead.

#### IDENTIFIER

an ASCII character string that only contains alphanumeric characters and hyphens or underscores. IDENTIFIERS should avoid including hyphens followed by positive integers except when constructing multiple-VALUE ATTRIBUTE IDENTIFIERS.

#### VALUE

a buffer of VALUE-SIZE octets containing the VALUE. The VALUE may contain data in arbitrary formats or encodings, which recipients recognize based on Template-Type.

#### VALUE-SIZE

a non-negative integer encoded as an ASCII character string. The integer indicates how many octets the VALUE occupies after the DELIMITER.

#### DELIMITER

a two octet delimiter which is a Latin-1 colon (:) and a tab (\t), (octal \072\011).

{ } the Latin-1 curly braces (octal \173 and \175) are used to wrap the VALUE-SIZE (no spaces) as well as the URL and ATTRIBUTE-LIST combination.

#### @TEMPLATE-TYPE

the Latin-1 @ (octal \100) and TEMPLATE-TYPE (no space between them) is used to mark the beginning of the SOIF object.

#### NUMERIC-STRING

Zero or more ASCII numerals.

#### ALPHA-NUMERIC-STRING

Zero or more ASCII letters or numerals, plus hyphens or underscore. [a-z,A-Z,0-9,- and \_].

#### ARBITRARY-DATA

Octets of data in arbitrary formats or encodings.

#### 4. Matching Semantics

As was discussed in Section 1, query referral of SOIF objects will be most effective when a query identifies a particular ATTRIBUTE or set of ATTRIBUTES as the target of the query match. A query-identified ATTRIBUTE should be considered to match a SOIF ATTRIBUTE when a case-insensitive character-by-character comparison matches that portion of the ATTRIBUTE IDENTIFIER prior to any hyphen-integer suffix. For example, a query which asks for a match on the ATTRIBUTE "author" should match the IDENTIFIERS "author", "Author", "AUTHOR", and "Author-1". [10] discourages the registration of template types containing ATTRIBUTES which have previously been registered with substantially different definitions. This will help eliminate mis-referral, but a CIP mesh may nonetheless need to maintain a thesaurus matching ATTRIBUTES from particular template-types to those of other, especially unregistered, template-types.

The matching semantics appropriate for a particular VALUE are derived from its data type and encoding. For VALUES associated with ATTRIBUTES which are part of a registered template type, the data type and encoding are readily available. For VALUES associated with ATTRIBUTES associated with unregistered template-types, an octet-by-octet comparison is the default. In cases where previous experience has demonstrated that a particular ATTRIBUTE contains string data, a case-insensitive substring match may be used. For example, in a query against the "AUTHOR" ATTRIBUTE of the generic "DOCUMENT" template type, the query VALUE "Garcia" should match the SOIF VALUES "Garcia", "GARCIA", and "Jose Garcia y Montes".

Over time, there may well emerge an understanding of which attributes tend to produce correct query referrals within a mesh. As such understandings emerge, mesh maintainers may wish to define a particular SOIF TEMPLATE-TYPE which restricts included ATTRIBUTES to those likely to foster correct referrals.

#### 5. Internationalization

The internationalization of SOIF depends on the registration of template-types. Since TEMPLATE-TYPES and ATTRIBUTE IDENTIFIERS must be in ASCII characters, only languages which use the ASCII character set are fully supported for unregistered TEMPLATE-TYPES. For registered template types, in contrast, the specification of an ATTRIBUTE's definition will allow UI designers to present a native-language mapping of the ATTRIBUTE to the end user. Further, the inclusion of data type and encoding information in the description of VALUES means that any language encoding or character set required by a particular application may be supported. For unregistered template types, the ability of peer servers to pass schema definitions may

provide a form of "private registration" which could provide some of the facilities for internationalization available to registered template types. (See above, section 3.1 and Refs. 6 and 7.)

## 6. Example Summary Objects

The appendices contain example summary objects encoded using specific template types. The following are some example summary objects using the generic "DOCUMENT" SOIF template-type:

```
@DOCUMENT { http://home.netscape.com:80/
Title{19}: Welcome to Netscape
Content-Type{9}: text/html
Content-Length{5}: 33262
}
```

```
@DOCUMENT { http://home.netscape.com/eng/ssl3/ssl-toc.html
Title{19}: SSL Protocol V. 3.0
Content-Type{9}: text/html
Content-Length{5}: 5870
Author-1{14}: Alan O. Freier
Author-2{14}: Philip Karlton
Author-3{14}: Paul C. Kocher
Abstract{318}: This document specifies Version 3.0 of the
<B>Secure Sockets Layer (SSL V3.0)</B> protocol, a security
protocol that provides communications privacy over the Internet.
The protocol allows client/server applications to communicate in
a way that is designed to prevent eavesdropping, tampering, or
message forgery.
}
```

```
@DOCUMENT { http://www.nissanmotors.com/1996/300ZX/pictures/300zx.jpg
Content-Type{10}: image/jpeg
Content-Length{5}: 25940
Last-Modified{31}: Tuesday, 11-Jun-96 19:18:44 GMT
Thumbnail{259}: .....
}
```

## 7. Security

Please see (Ref. 1) for a general discussion of Security concerns for the CIP framework.

SOIF currently contains no requirement that any template type contain an authentication ATTRIBUTE. SOIF summary objects lacking authentication ATTRIBUTES must, therefore, be treated as unreliable indicators of the referenced resource's content. A hostile party could create a summary object which significantly misrepresented a

resource's content. As part of a CIP mesh, this data could either channel a large number of requestors to a resource (possibly resulting in a denial of service) or away from a resource (possibly resulting in a loss of appropriate visibility).

## 8. References

- [1] Allen, J. and M. Mealling, "The Architecture of the Common Indexing Protocol (CIP)", RFC 2651, August 1999.
- [2] The Harvest Information Discovery and Access System:  
<URL:<http://harvest.transarc.com/>>.
- [3] D. Beckett, IAFA Templates in Use as Internet Metadata, 4th Int'l WWW Conference, December 1995,  
<URL:<http://www.hensa.ac.uk/tools/www/iafatools/>>
- [4] L. Lamport, LaTeX: A Document Preparation System, Addison-Wesley, Reading, Mass., 1986.
- [5] Berners-Lee, T., Masinter, L. and M. McCahill, "Uniform Resource Locators (URL)", RFC 1738, December 1994.
- [6] D. Hardey, Resource Description Messages (RDM), W3C Note-rdm-960724, July 24, 1996, <URL:<http://www.w3.org/pub/WWW/TR/NOTE-rdm.html>>
- [7] L. Gravano, K. Chang, H. Garcia-Molina, C. Lagoze, A. Paepcke, STARTS: Stanford Protocol Proposal for Internet Retrieval and Search, January 1997, <URL:<http://www-db.stanford.edu/~gravano/starts.html>>
- [8] S. Weibel, J. Kunze, C. Lagoze, Dublin Core Metadata for Simple Resource Description, Work in Progress.
- [9] E. Miller, Dublin Core Element Set Crosswalk, January 1997,  
<URL:<http://www.oclc.org:5046/~emiller/DC/crosswalk.html>>
- [10] Hardie, T., "Registration Procedures for SOIF Template Types", RFC 2656, August 1999.



## 9. Authors' Addresses

Ted Hardie  
Equinix  
901 Marshall Street  
Redwood City, CA 94063 USA

EMail: hardie@equinix.com

Mic Bowman  
Transarc Corporation  
The Gulf Tower  
707 Grant Street  
Pittsburgh, PA 15219 USA

Phone: +1 412 338 4400  
EMail: mic@transarc.com

Darren Hardy  
Netscape Communications Corp.  
685 E. Middlefield Road  
Mountain View, CA 94043 USA

Phone: +1 415 937 2555  
EMail: dhardy@netscape.com

Mike Schwartz  
Affinia, Inc.  
621 17th Street, Suite 1700  
Denver, CO 80293

Phone: +1 (303) 292-4818  
E-mail: mfs@affinia.net

Duane Wessels  
National Laboratory for Applied Network Research

Phone: +1 303 497 1822  
EMail: wessels@nlanr.net

## Appendix A.

Common Attributes for "FILE" Template-type Summary Objects created by Harvest:

## Abstract

Brief abstract about the object.

## Author

Author(s) of the object.

## Description

Brief description about the object.

## File-Size

Number of bytes in the object.

## Full-Text

Entire contents of the object.

## Gatherer-Host

Host on which the Gatherer ran to extract information from the object.

## Gatherer-Name

Name of the Gatherer that extracted information from the object.  
(eg. Full-Text, Selected-Text, or Terse).

## Gatherer-Port

Port number on the Gatherer-Host that serves the Gatherer's information.

## Gatherer-Version

Version number of the Gatherer.

## Update-Time

The time that Gatherer updated the content summary for the object.

## Keywords

Searchable keywords extracted from the object.

## Last-Modification-Time

The time that the object was last modified.

## MD5

MD5 16-byte checksum of the object.

**Refresh-Rate**

The number of seconds after Update-Time when the summary object is to be re-generated. Defaults to 1 month.

**Time-to-Live**

The number of seconds after Update-Time when the summary object is no longer valid. Defaults to 6 months.

**Title**

Title of the object.

**Type** The object's type. Some example types are:

- Archive
- Audio
- Awk
- Backup
- Binary
- C
- CHeader
- Command
- Compressed
- CompressedTar
- Configuration
- Data
- Directory
- DotFile
- Dvi
- FAQ
- FYI
- Font
- FormattedText
- GDBM
- GNUCompressed
- GNUCompressedTar
- HTML
- Image
- Internet-Draft
- MacCompressed
- Mail
- Makefile
- ManPage
- Object
- OtherCode
- PCCompressed
- Patch
- Perl
- PostScript

RCS  
README  
RFC  
SCCS  
ShellArchive  
Tar  
Tcl  
Tex  
Text  
Troff  
Uuencoded  
WaisSource

#### Update-Time

The time that the summary object was last updated. REQUIRED field, no default.

#### URL-References

Any URL references present within HTML objects.

### Appendix B.

#### Proposed Attributes for a "CIP-HINT" Template Type

##### Attribute-Identifier-List

A comma-delimited list whose entries take the form Template-Type:Attribute . This list identifies the attributes against which queries are supported. Because of the current limitation on Identifiers, this list must be in ASCII.

##### Source

The URI of the service which created some or all of the index objects to which this hint applies. Note that this service may be and often is distinct from the server which provides query access to those objects.

##### Total-Object-Count

The total number of index objects in the collection for which the Hint applies. This should be a positive integer.

##### Weightlist-[Attribute-Identifier]

This construction allows the HINT to contain a weighted list of values for a specific Attribute-Identifier. There may be as many Weightlist entries as there Attribute-Identifiers in the Attribute-Identifier-List. Each Weightlist entry takes the form of Value;Object-Count, where the object count is a positive integer representing the number of objects within the collection which contain that value. Weightlists are comma- delimited.

Should a Value contain a comma, it should be escaped when incorporated into the weightlist.

#### Threshold-[Attribute-Identifier]

If a server wishes not to report infrequently occurring Values in a specific Weightlist, it may declare a threshold under which it will not report Values.

#### Certification-Type

The type of Certification used for this object

#### Certification

The Value of the Certification.

#### Date

The Date at which the hint was generated

#### Example:

```
@CIP-HINT{ http://nic.nasa.gov:80/Harvest/brokers/NASA/
Attribute-Identifier-list{49}:
DOCUMENT:Author, DOCUMENT:Keywords, IMAGE:Subject
Source-1{45}: http://nic.nasa.gov/Harvest/gatherers/Eureka/
Source-2{46}: http://techreports.larc.nasa.gov/cgi-bin/NTRS/
Total-Object-Count{5}:      10000
Weightlist-[IMAGE:Subject]{40}:
Shuttle;100, Planet;227, Moon;15, Sun;33
Threshold-[IMAGE:Subject]{2}:      10
Weightlist-[DOCUMENT:Author]{49}:
Grizzard;12, Aldrin\, Buzz;15, Aldrin\, James;45,
Threshold-[DOCUMENT:Author]{1}:      5
Certification-Type{13}:      PGP-Signature
Certification{51}: mQCNAzFNm5QAEEALUBOolOWKpby+=YtmtBxUZWQgSGFyZGllID
Date{29}:      Sun, 05 Jan 1997 08:33:33 GMT
}
```

#### Appendix C.

A "Dublin-Core" Template Type [Ref. 8,9]

#### TITLE

The name given to the resource by the CREATOR or PUBLISHER.

#### CREATOR

The person(s) or organization(s) primarily responsible for the intellectual content of the resource. For example, authors in the case of written documents, artists, photographers, or illustrators in the case of visual resources.

**SUBJECT**

The topic of the resource, or keywords or phrases that describe the subject or content of the resource. The intent of the specification of this element is to promote the use of controlled vocabularies and keywords. This element might well include scheme-qualified classification data (for example, Library of Congress Classification Numbers or Dewey Decimal numbers) or scheme-qualified controlled vocabularies (such as Medical Subject Headings or Art and Architecture Thesaurus descriptors) as well.

**DESCRIPTION**

A textual description of the content of the resource, including abstracts in the case of document-like objects or content descriptions in the case of visual resources. Future metadata collections might well include computational content description (spectral analysis of a visual resource, for example) that may not be embeddable in current network systems. In such a case this field might contain a link to such a description rather than the description itself.

**PUBLISHER**

The entity responsible for making the resource available in its present form, such as a publisher, a university department, or a corporate entity. The intent of specifying this field is to identify the entity that provides access to the resource.

**CONTRIBUTOR**

Person(s) or organization(s) in addition to those specified in the CREATOR element who have made significant intellectual contributions to the resource but whose contribution is secondary to the individuals or entities specified in the CREATOR element (for example, editors, transcribers, illustrators, and convenors).

**DATE**

The date the resource was made available in its present form. The recommended best practice is an 8 digit number in the form YYYYMMDD as defined by ANSI X3.30-1985. In this scheme, the date element for the day this is written would be 19961203, or December 3, 1996. Many other schema are possible, but if used, they should be identified in an unambiguous manner.

**TYPE**

The category of the resource, such as home page, novel, poem, working paper, technical report, essay, dictionary. It is expected that RESOURCE TYPE will be chosen from an enumerated list of types.

**FORMAT**

The data representation of the resource, such as text/html, ASCII, Postscript file, executable application, or JPEG image. The intent of specifying this element is to provide information necessary to allow people or machines to make decisions about the usability of the encoded data (what hardware and software might be required to display or execute it, for example). As with RESOURCE TYPE, FORMAT will be assigned from enumerated lists such as registered Internet Media Types (MIME types). In principal, formats can include physical media such as books, serials, or other non-electronic media.

**IDENTIFIER**

String or number used to uniquely identify the resource. Examples for networked resources include URLs and URNs (when implemented). Other globally-unique identifiers, such as International Standard Book Numbers (ISBN) or other formal names would also be candidates for this element.

**SOURCE**

The work, either print or electronic, from which this resource is derived, if applicable. For example, an html encoding of a Shakespearean sonnet might identify the paper version of the sonnet from which the electronic version was transcribed.

**LANGUAGE**

Language(s) of the intellectual content of the resource. Where practical, the content of this field should coincide with the NISO Z39.53 three character codes for written languages.

**RELATION**

Relationship to other resources. The intent of specifying this element is to provide a means to express relationships among resources that have formal relationships to others, but exist as discrete resources themselves. For example, images in a document, chapters in a book, or items in a collection. A formal specification of RELATION is currently under development. Users and developers should understand that use of this element should be currently considered experimental.

**COVERAGE**

The spatial locations and temporal durations characteristic of the resource. Formal specification of COVERAGE is currently under development. Users and developers should understand that use of this element should be currently considered experimental.

## RIGHTS

The content of this element is intended to be a link (a URL or other suitable URI as appropriate) to a copyright notice, a rights-management statement, or perhaps a server that would provide such information in a dynamic way. The intent of specifying this field is to allow providers a means to associate terms and conditions or copyright statements with a resource or collection of resources. No assumptions should be made by users if such a field is empty or not present.

## Example:

```
@Dublin-Core-1 { ftp://ds.internic.net/internet-drafts/
                  draft-kunze-dc-00.txt
TITLE{52}:       Dublin Core Metadata for Simple Resource Description
CREATOR-1{9}:    S. Weibel
CREATOR-2{8}:    J. Kunze
CREATOR-3{9}:    C. Lagoze
SUBJECT{44}:     The Dublin Core Set of Elements for Metadata
DESCRIPTION{46}: Reference description of Dublin Core elements.
PUBLISHER{31}:   Internet Engineering Task Force
CONTRIBUTOR-1{11}: Nick Arnett
CONTRIBUTOR-2{15}: Eliot Christian
CONTRIBUTOR-3{14}: Martijn Koster
CONTRIBUTOR-4{18}: Christian Mogensen
CONTRIBUTOR-5{14}: Timothy Niesen
CONTRIBUTOR-6{11}: Andrew Wood
CONTRIBUTOR-7{10}: Mic Bowman
CONTRIBUTOR-8{11}: Dan Connolly
CONTRIBUTOR-9{15}: Michael Mauldin
CONTRIBUTOR-10{12}: Wick Nichols
DATE{16}:       February 9, 1997
TYPE{14}:       Internet draft
FORMAT{4}:      Text
IDENTIFIER:{21} draft-kunze-dc-00.txt
SOURCE{41}:     http://purl.oclc.org/metadata/dublin_core
LANGUAGE{3}:    eng
RELATION{24}:   Draft Reference Standard
COVERAGE{22}:  Expires August 8, 1997
RIGHTS{58}:     Unlimited Distribution;
                  readers must not cite as standard.
}
```



## 11. Full Copyright Statement

Copyright (C) The Internet Society (1999). All Rights Reserved.

This document and translations of it may be copied and furnished to others, and derivative works that comment on or otherwise explain it or assist in its implementation may be prepared, copied, published and distributed, in whole or in part, without restriction of any kind, provided that the above copyright notice and this paragraph are included on all such copies and derivative works. However, this document itself may not be modified in any way, such as by removing the copyright notice or references to the Internet Society or other Internet organizations, except as needed for the purpose of developing Internet standards in which case the procedures for copyrights defined in the Internet Standards process must be followed, or as required to translate it into languages other than English.

The limited permissions granted above are perpetual and will not be revoked by the Internet Society or its successors or assigns.

This document and the information contained herein is provided on an "AS IS" basis and THE INTERNET SOCIETY AND THE INTERNET ENGINEERING TASK FORCE DISCLAIMS ALL WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY WARRANTY THAT THE USE OF THE INFORMATION HEREIN WILL NOT INFRINGE ANY RIGHTS OR ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.

## Acknowledgement

Funding for the RFC Editor function is currently provided by the Internet Society.

