Internet-Draft (Informational) J. Kunze
draft-kunze-dchtml-02.txt
15 September 1999 Dublin Core
Expires 15 March 2000 Metadata Initiative

Encoding Dublin Core Metadata in HTML

(ftp://ftp.ietf.org/internet-drafts/draft-kunze-dchtml-02.txt)

1. Status of this Document

This document is an Internet-Draft and is in full conformance with all provisions of Section 10 of RFC2026.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF), its areas, and its working groups. Note that other groups may also distribute working documents as Internet-Drafts.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as ``work in progress.''

The list of current Internet-Drafts can be accessed at http://www.ietf.org/ietf/1id-abstracts.txt

The list of Internet-Draft Shadow Directories can be accessed at http://www.ietf.org/shadow.html.

To learn the current status of any Internet-Draft, please check the ``1id-abstracts.txt'' listing contained in the Internet-Drafts Shadow Directories on ftp.is.co.za (Africa), nic.nordu.net (Europe), munnari.oz.au (Pacific Rim), ds.internic.net (US East Coast), or ftp.isi.edu (US West Coast).

Distribution of this document is unlimited. Please send comments to jak@ckm.ucsf.edu or to the dc-general@mailbase.ac.uk discussion list.

2. Abstract

The Dublin Core [DC1] is a small set of metadata elements for describing information resources. This document explains how these elements are expressed using the META and LINK tags of HTML [HTML4.0]. A sequence of metadata elements embedded in an HTML file is taken to be a description of that file. Examples illustrate conventions allowing interoperation with current software that indexes, displays, and manipulates metadata, such as [SWISH-E], [freeWAIS-sf2.0], [GLIMPSE], [HARVEST], [ISEARCH], etc., and the Perl [PERL] scripts in the appendix.

3. HTML, Dublin Core, and Non-Dublin Core Metadata

The Dublin Core (DC) metadata initiative [DCHOME] has produced a small set of resource description categories [DC1], or elements of metadata (literally, data about data). Metadata elements are typically small relative to the resource they describe and may, if the resource format permits, be embedded in it. Two such formats are the Hypertext Markup Language (HTML) and the Extensible Markup Language (XML); HTML is currently in wide use, but once standardized, XML [XML] in conjunction with the Resource Description Framework [RDF] promise a significantly more expressive means of encoding metadata. The [RDF] specification actually describes a way to use RDF within an HTML document by adhering to an abbreviated syntax.

This document explains how to encode metadata using HTML 4.0 [HTML4.0]. It is not concerned with element semantics, which are defined elsewhere. For illustrative purposes, some element semantics are alluded to, but in no way should semantics appearing here be considered definitive.

The HTML encoding allows elements of DC metadata to be interspersed with non-DC elements (provided such mixing is consistent with rules governing use of those non-DC elements). A DC element is indicated by the prefix "DC", and a non-DC element by another prefix; for example, the prefix "AC" is used with elements from the A-Core [AC].

4. The META Tag

The META tag of HTML is designed to encode a named metadata element. Each element describes a given aspect of a document or other information resource. For example, this tagged metadata element,

```
    <meta name    = "DC.Creator"
          content = "Simpson, Homer">
```

says that Homer Simpson is the Creator, where the element named Creator is defined in the DC element set. In the more general form,

```
    <meta name    = "PREFIX.ELEMENT_NAME"
          content = "ELEMENT_VALUE">
```

the capitalized words are meant to be replaced in actual descriptions; thus in the example,

```
          ELEMENT_NAME   was:  Creator
          ELEMENT_VALUE  was:  Simpson, Homer
          and PREFIX     was:  DC
```

Within a META tag the first letter of a Dublin Core element name is capitalized. DC places no restriction on alphabetic case in an element value and any number of META tagged elements may appear together, in any order. More than one DC element with the same name may appear, and each DC element is optional. The next example is a book description with two authors, two titles, and no other metadata.

```
    <meta name    = "DC.Title"
          content = "The Communist Manifesto">
    <meta name    = "DC.Creator"
          content = "Marx, K.">
    <meta name    = "DC.Creator"
          content = "Engels, F.">
    <meta name    = "DC.Title"
          content = "Capital">
```

The prefix "DC" precedes each Dublin Core element encoded with META, and it is separated by a period (.) from the element name following it. Each non-DC element should be encoded with a prefix that can be used to trace its origin and definition; the linkage between prefix and element definition is made with the LINK tag, as explained in the next section. Non-DC elements, such as Email from the A-Core [AC], may appear together with DC elements, as in

```
    <meta name    = "DC.Creator"
          content = "Da Costa, Jos&eacute;">
    <meta name    = "AC.Email"
          content = "dacostaj@peoplesmail.org">
    <meta name    = "DC.Title"
          content = "Jesse &#34;The Body&#34; Ventura--A Biography">
```

This example also shows how some special characters may be encoded. The author name in the first element contains a diacritic encoded as an HTML character entity reference -- in this case an accented letter E. Similarly, the last line contains two double-quote characters encoded so as to avoid being interpreted as element content delimiters.

5. The LINK Tag

The LINK tag of HTML may be used to associate an element name prefix with the reference definition of the element set that it identifies. A sequence of META tags describing a resource is incomplete without one such LINK tag for each different prefix appearing in the sequence. The previous example could be considered complete with the addition of these two LINK tags:

```
    <link rel     = "schema.DC"
          href    = "http://purl.org/DC/elements/1.0/">
    <link rel     = "schema.AC"
          href    = "http://metadata.net/ac/2.0/">
```

In general, the association takes the form

```
    <link rel     = "schema.PREFIX"
          href    = "LOCATION_OF_DEFINITION">
```

where, in actual descriptions, PREFIX is to be replaced by the prefix and LOCATION_OF_DEFINITION by the URL or URN of the defining document.

When embedded in the HEAD part of an HTML file, a sequence of LINK and META tags describes the information in the surrounding HTML file itself. Here is a complete HTML file with its own embedded description.

```
    <html>
    <head>
    <title> A Dirge </title>
    <link rel     = "schema.DC"
          href    = "http://purl.org/DC/elements/1.0/">
    <meta name    = "DC.Title"
          content = "A Dirge">
    <meta name    = "DC.Creator"
          content = "Shelley, Percy Bysshe">
    <meta name    = "DC.Type"
          content = "poem">
    <meta name    = "DC.Date"
          content = "1820">
    <meta name    = "DC.Format"
          content = "text/html">
    <meta name    = "DC.Language"
          content = "en">
    </head>
    <body><pre>
            Rough wind, that moanest loud
              Grief too sad for song;
            Wild wind, when sullen cloud
              Knells all the night long;
            Sad storm, whose tears are vain,
            Bare woods, whose branches strain,
            Deep caves and dreary main, -
              Wail, for the world's wrong!
    </pre></body>
    </html>
```

6. Encoding Recommendations

HTML allows more flexibility in principle and in practice than is recommended here for encoding metadata. Limited flexibility encourages easy development of software for extracting and processing metadata. At this early evolutionary stage of internet metadata, easy prototyping and experimentation hastens the development of useful standards.

Adherence is therefore recommended to the tagging style exemplified in this document as regards prefix and element name capitalization, double-quoting (") of attribute values, and not starting more than one META tag on a line. There is much room for flexibility, but choosing a style and sticking with it will likely make metadata manipulation and editing easier. The following META tags adhere to the recommendations and carry identical metadata in three different styles:

```
    <META NAME="DC.Format"
```

```
      CONTENT="text/html; 12 Kbytes">
    <meta
            Content = "text/html; 12 Kbytes"
            Name = "DC.Format"
    >
    <meta name = "DC.Format" content = "text/html; 12 Kbytes">
```

Use of these recommendations is known to result in metadata that may be harvested, indexed, and manipulated by popular, freely available software packages such as [SWISH-E], [freeWAIS-sf2.0], [GLIMPSE], [HARVEST], and [ISEARCH], among others. These conventions also work with the metadata processing scripts appearing in the appendix, as well as with most of the [DCPROJECTS] applications referenced from the [DCHOME] site. Software support for the LINK tag and qualifier conventions (see the next section) is not currently widespread.

Ordering of metadata elements is not preserved in general. Writers of software for metadata indexing and display should try to preserve relative ordering among META tagged elements having the same name (e.g., among multiple authors), however, metadata providers and searchers have no guarantee that ordering will be preserved in metadata that passes through unknown systems.

7. Dublin Core in Real Descriptions

In actual resource description it is often necessary to qualify Dublin Core elements to add nuances of meaning. While neither the general principles nor the specific semantics of DC qualifiers are within scope of this document, everyday uses of the qualifier syntax are illustrated to lend realism to later examples. Without further explanation, the three ways in which the optional qualifier syntax is currently (subject to change) used to supplement the META tag may be summarized as follows:

```
    <meta lang    = "LANGUAGE_OF_METADATA_CONTENT" ... >

    <meta scheme  = "CONTROLLED_FORMAT_OR_VOCABULARY_OF_METADATA" ... >

    <meta name    = "PREFIX.ELEMENT_NAME.SUBELEMENT_NAME" ... >
```

Accordingly, a posthumous work in Spanish might be described with

```
    <meta name    = "DC.Language"
          scheme  = "rfc1766"
          content = "es">
    <meta name    = "DC.Title"
          lang    = "es"
          content = "La Mesa Verde y la Silla Roja">
    <meta name    = "DC.Title"
          lang    = "en"
          content = "The Green Table and the Red Chair">
    <meta name    = "DC.Date.Created"
```

```
          content = "1935">
    <meta name    = "DC.Date.Available"
          content = "1939">
```

Note that the qualifier syntax and label suffixes (which follow an element name and a period) used in examples in this document merely reflect current trends in the HTML encoding of qualifiers. Use of this syntax and these suffixes is neither a standard nor a recommendation.

8. Encoding Dublin Core Elements

This section consists of very simple Dublin Core encoding examples, arranged by element.

Title (name given to the resource)

```
    <meta name    = "DC.Title"
          content = "Polycyclic aromatic hydrocarbon contamination">

    <meta name    = "DC.Title"
          content = "Crime and Punishment">

    <meta name    = "DC.Title"
          content = "Methods of Information in Medicine, Vol 32, No 4">

    <meta name    = "DC.Title"
          content = "Still life #4 with flowers">

    <meta name    = "DC.Title"
          lang    = "de"
          content = "Das Wohltemperierte Klavier, Teil I">
```

Creator (entity that created the content)

```
    <meta name    = "DC.Creator"
          content = "Gogh, Vincent van">
    <meta name    = "DC.Creator"
          content = "van Gogh, Vincent">

    <meta name    = "DC.Creator"
          content = "Mao Tse Tung">
    <meta name    = "DC.Creator"
          content = "Mao, Tse Tung">

    <meta name    = "DC.Creator"
          content = "Plato">
    <meta name    = "DC.Creator"
          lang    = "fr"
          content = "Platon">
```

```
    <meta name    = "DC.Creator.Director"
          content = "Sturges, Preston">
    <meta name    = "DC.Creator.Writer"
          content = "Hecht, Ben">
    <meta name    = "DC.Creator.Producer"
          content = "Chaplin, Charles">

Subject (topic or keyword)
-------

    <meta name    = "DC.Subject"
          content = "heart attack">
    <meta name    = "DC.Subject"
          scheme  = "MESH"
          content = "Myocardial Infarction; Pericardial Effusion">

    <meta name    = "DC.Subject"
          content = "vietnam war">
    <meta name    = "DC.Subject"
          scheme  = "LCSH"
          content = "Vietnamese Conflict, 1961-1975">

    <meta name    = "DC.Subject"
          content = "Friendship">
    <meta name    = "DC.Subject"
          scheme  = "ddc"
          content = "158.25">

Description (account, summary, or abstract of the content)
-----------

    <meta name    = "DC.Description"
          lang    = "en"
          content = "The Author gives some Account of Himself and Family
                     -- His First Inducements to Travel -- He is
                     Shipwrecked, and Swims for his Life -- Gets safe on
                     Shore in the Country of Lilliput -- Is made a
                     Prisoner, and carried up the Country">

    <meta name    = "DC.Description"
          content = "A tutorial and reference manual for Java.">

    <meta name    = "DC.Description"
          content = "Seated family of five, coconut trees to the left,
                     sailboats moored off sandy beach to the right,
                     with volcano in the background.">

Publisher (entity that made the resource available)
---------

    <meta name    = "DC.Publisher"
          content = "O'Reilly">
```

```
    <meta name    = "DC.Publisher"
          content = "Digital Equipment Corporation">

    <meta name    = "DC.Publisher"
          content = "University of California Press">

    <meta name    = "DC.Publisher"
          content = "State of Florida (USA)">

Contributor (other entity that made a contribution)
-----------

    <meta name    = "DC.Contributor"
          content = "Curie, Marie">

    <meta name    = "DC.Contributor.Photographer"
          content = "Adams, Ansel">
    <meta name    = "DC.Contributor.Artist"
          content = "Sendak, Maurice">
    <meta name    = "DC.Contributor.Editor"
          content = "Starr, Kenneth">

Date (of an event in the life of the resource; [WTN8601] recommended)
----

    <meta name    = "DC.Date"
          content = "1972">

    <meta name    = "DC.Date"
          content = "1998-05-14">
    <meta name    = "DC.Date"
          scheme  = "WTN8601"
          content = "1998-05-14">

    <meta name    = "DC.Date.Created"
          content = "1998-05-14">
    <meta name    = "DC.Date.Available"
          content = "1998-05-21">
    <meta name    = "DC.Date.Valid"
          content = "1998-05-28">

    <meta name    = "DC.Date.Created"
          content = "triassic">
    <meta name    = "DC.Date.Acquired"
          content = "1957">

    <meta name    = "DC.Date.Accepted"
          scheme  = "WTN8601"
          content = "1998-12-02T16:59">

    <meta name    = "DC.Date.DataGathered"
```

```
          scheme  = "ISO8601"
          content = "98-W49-3T1659">

    <meta name    = "DC.Date.Issued"
          scheme  = "ANSI.X3.X30-1985"
          content = "19980514">
```

Type (nature, genre, or category; [DCT1] recommended)

```
    <meta name    = "DC.Type"
          content = "poem">

    <meta name    = "DC.Type"
          scheme  = "DCT1"
          content = "software">
    <meta name    = "DC.Type"
          content = "software program source code">
    <meta name    = "DC.Type"
          content = "interactive video game">

    <meta name    = "DC.Type"
          scheme  = "DCT1"
          content = "dataset">

    <meta name    = "DC.Type"
          content = "web home page">
    <meta name    = "DC.Type"
          content = "web bibliography">

    <meta name    = "DC.Type"
          content = "painting">
    <meta name    = "DC.Type"
          content = "image; woodblock">
    <meta name    = "DC.Type"
          scheme  = "AAT"
          content = "clipeus (portrait)">
    <meta name    = "DC.Type"
          lang    = "en-US"
          content = "image; advertizement">

    <meta name    = "DC.Type"
          scheme  = "DCT1"
          content = "event">
    <meta name    = "DC.Type"
          content = "event; periodic">
```

Format (physical or digital data format, plus optional dimensions)

```
    <meta name    = "DC.Format"
          content = "text/xml">
```

```
    <meta name    = "DC.Format"
          scheme  = "IMT"
          content = "text/xml">

    <meta name    = "DC.Format"
          scheme  = "IMT"
          content = "image/jpeg">
    <meta name    = "DC.Format"
          content = "A text file with mono-spaced tables and diagrams.">

    <meta name    = "DC.Format"
          content = "video/mpeg; 14 minutes">

    <meta name    = "DC.Format"
          content = "unix tar archive, gzip compressed; 1.5 Mbytes">

    <meta name    = "DC.Format"
          content = "watercolor; 23 cm x 31 cm">

Identifier (of the resource)
----------

    <meta name    = "DC.Identifier"
          content = "http://foo.bar.org/zaf/">

    <meta name    = "DC.Identifier"
          content = "urn:ietf:rfc:1766">

    <meta name    = "DC.Identifier"
          scheme  = "ISBN"
          content = "1-56592-149-6">

    <meta name    = "DC.Identifier"
          scheme  = "LCCN"
          content = "67-26020">

    <meta name    = "DC.Identifier"
          scheme  = "DOI"
          content = "10.12345/33-824688ab">

Source (reference to the resource's origin)
------

    <meta name    = "DC.Source"
          content = "Shakespeare's Romeo and Juliet">

    <meta name    = "DC.Source"
          content = "http://a.b.org/manon/">

Language (of the content of the resource; [RFC1766] recommended)
--------
```

```
    <meta name    = "DC.Language"
          content = "en">
    <meta name    = "DC.Language"
          scheme  = "rfc1766"
          content = "en">
    <meta name    = "DC.Language"
          scheme  = "ISO639-2"
          content = "eng">

    <meta name    = "DC.Language"
          scheme  = "rfc1766"
          content = "en-US">

    <meta name    = "DC.Language"
          content = "zh">
    <meta name    = "DC.Language"
          content = "ja">
    <meta name    = "DC.Language"
          content = "es">
    <meta name    = "DC.Language"
          content = "de">

    <meta name    = "DC.Language"
          content = "german">
    <meta name    = "DC.Language"
          lang    = "fr"
          content = "allemand">

Relation (reference to a related resource)
--------

    <meta name    = "DC.Relation.IsPartOf"
          content = "http://foo.bar.org/abc/proceedings/1998/">

    <meta name    = "DC.Relation.IsFormatOf"
          content = "http://foo.bar.org/cd145.sgml">

    <meta name    = "DC.Relation.IsVersionOf"
          content = "http://foo.bar.org/draft9.4.4.2">

    <meta name    = "DC.Relation.References"
          content = "urn:isbn:1-56592-149-6">

    <meta name    = "DC.Relation.IsBasedOn"
          content = "Shakespeare's Romeo and Juliet">

    <meta name    = "DC.Relation.Requires"
          content = "LWP::UserAgent; HTML::Parse; URI::URL;
                     Net::DNS; Tk::Pixmap; Tk::Bitmap; Tk::Photo">

Coverage (extent or scope of the content)
--------
```

```
    <meta name    = "DC.Coverage"
          content = "US civil war era; 1861-1865">

    <meta name    = "DC.Coverage"
          content = "Columbus, Ohio, USA; Lat: 39 57 N Long: 082 59 W">

    <meta name    = "DC.Coverage"
          scheme  = "TGN"
          content = "Columbus (C,V)">

    <meta name    = "DC.Coverage.Jurisdiction"
          content = "Commonwealth of Australia">
```

Rights (text or identifier of a rights management statement)

```
    <meta name    = "DC.Rights"
          lang    = "en"
          content = "Copyright Acme 1999 - All rights reserved.">

    <meta name    = "DC.Rights"
          content = "http://foo.bar.org/cgi-bin/terms">
```

9. Security Considerations

The syntax rules for encoding Dublin Core metadata in HTML that are documented here pose no direct risk to computers and networks. People can use these rules to encode metadata that is inaccurate or even deliberately misleading (creating mischief in the form of "index spam"), however, this reflects a general pattern of HTML META tag abuse that is not limited to the encoding of metadata from the Dublin Core set. Even traditional metadata encoding schems (e.g., [MARC]) are not immune to inaccuracy, although they are generally followed in environments where production quality greatly exceeds that of the average Web site.

Systems that process metadata encoded with META tags need to consider issues related to its accuracy and validity as part of their design and implementation, and users of such systems need to consider the design and implementation assumptions. Various approaches may be relevant for certain applications, such as adding statements of metadata provenance, signing of metadata with digital signatures, and automating certain aspects of metadata creation; but these are far outside the scope of this document and the underlying META tag syntax that it describes.

10. Copyright Notice

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.

The IETF invites any interested party to bring to its attention any copyrights, patents or patent applications, or other proprietary rights which may cover technology that may be required to practice this standard. Please address the information to the IETF Executive Director.

11. Appendix -- Perl Scripts that Manipulate HTML Encoded Metadata

This section contains two simple programs that work with versions 4 and 5 of the Perl [PERL] scripting language interpreter. They may be taken and freely adapted for local organizational needs, research proposals, venture capital bids, etc. A variety of applications are within easy reach of implementors that choose to build on these scripts.

Script 1: Metadata Format Conversion

Here is a simple Perl script that correctly recognizes every example of metadata encoding in this document. It shows how a modest scripting effort can produce a utility that converts metadata from one format to another. Minor changes are sufficient to support a number of output formats.

```
#!/depot/bin/perl
#
# This simple perl script extracts metadata embedded in an HTML file
# and outputs it in an alternate format.  Issues warning about missing
# element name or value.
#
# Handles mixed case tags and attribute values, one per line or spanning
# several lines.  Also handles a quoted string spanning multiple lines.
```

```
# No error checking.  Does not tolerate more than one "<meta" per line.

print "@(urc;\n";
while (<>) {
        next if (! /<meta/i);
        ($meta) = /(<meta.*$)/i;
        if (! /<meta.*>/i) {
                while (<>) {
                        $meta .= $_;
                        last if (/>/);
                }
        }
        $name     =    $meta =~ /name\s*=\s*"([^"]*)"/i
                        ? $1 : "MISSING ELEMENT NAME";
        $content  = $meta =~ /content\s*=\s*"([^"]*)"/i
                        ? $1 : "MISSING ELEMENT VALUE";
        ($scheme) =  $meta =~ /scheme\s*=\s*"([^"]*)"/i;
        ($lang)   =    $meta =~ /lang\s*=\s*"([^"]*)"/i;

        if ($lang || $scheme) {
                $mod = " ($lang";
                if (! $scheme)
                        { $mod .= ")"; }
                elsif (! $lang)
                        { $mod .= "$scheme)" }
                else
                        { $mod .= ", $scheme)"; }
        }
        else
                { $mod = ""; }

        print "    @|$name$mod; $content\n";
}
print "@)urc;\n";
# ---- end of Perl script ----
```

When the conversion script is run on the metadata file example from the LINK tag section (section 5), it produces the following output.

```
        @(urc;
            @|DC.Title; A Dirge
            @|DC.Creator; Shelley, Percy Bysshe
            @|DC.Type; poem
            @|DC.Date; 1820
            @|DC.Format; text/html
            @|DC.Language; en
        @)urc;
```

Script 2: Automated Metadata Creation

The creation and maintenance of high-quality metadata can be extremely expensive without automation to assist in processes such as supplying pre-set or computed defaults, validating syntax, verifying value ranges, spell checking, etc. Considerable relief could be had from a script that reduced an individual provider's metadata burden to just the title of each document. Below is such a script. It lets the provider of an HTML document abbreviate an entire embedded resource description using a single HTML comment statement that looks like

```
    <!--metablock Little Red Riding Hood -->
```

Our script processes this statement specially as a kind of "metadata block" declaration with attached title. The general form is

```
    <!--metablock TITLE_OF_DOCUMENT -->
```

This statement works much like a "Web server-side include" in that the script replaces it with a fully-specified block of metadata and triggers other replacements. Once installed, the script can output HTML files suitable for integration into one's production Web server procedures.

The individual provider keeps a separate "template" file of infrequently changing pre-set values for metadata elements. If the provider's needs are simple enough, the only element values besides the title that differ from one document to the next may be generated automatically. Using the script, values may be referenced as variables from within the template or within the document. Our variable references have the form "(--mbVARNAME)", and here is what they look like inside a template:

```
    <title> (--mbtitle) </title>
    <meta name    = "DC.Creator"
          content = "Simpson, Homer">
    <meta name    = "DC.Title"
          content = "(--mbtitle)">
    <meta name    = "DC.Date.Created"
          content = "(--mbfilemodtime)">
    <meta name    = "DC.Identifier"
          content = "(--mbbaseURL)/(--mbfilename)">
    <meta name    = "DC.Format"
          content = "text/html; (--mbfilesize)">
    <meta name    = "DC.Language"
          content = "(--mblanguage)-BUREAUCRATESE">
    <meta name    = "RC.MetadataAuthority"
          content = "Springfield Nuclear">
    <link rel     = "schema.DC"
          href    = "http://purl.org/DC/elements/1.0/">
    <link rel     = "schema.RC"
          href    = "http://nukes.org/ReactorCore/rc">
```

The above template represents the metadata block that will describe the document once the variable references are replaced with real values.

By the conventions of our script, the following variables will be replaced in both the template and in the document:

```
    (--mbfilesize)            size of the final output file
    (--mbtitle)               title of the document
    (--mblanguage)            language of the document
    (--mbbaseURL)             beginning part of document identifier
    (--mbfilename)            last part (minus .html) of identifier
    (--mbfilemodtime)         last modification date of the document
```

Here's an example HTML file to run the script on.

```
    <html>
    <head>
    <!--metablock Nutritional Allocation Increase -->
    <meta name    = "DC.Type"
          content = "Memorandum">
    </head>
    <body>
    <p>
    From:  Acting Shift Supervisor
    To:    Plant Control Personnel
    RE:    (--mbtitle)
    Date:  (--mbfilemodtime)
    <p>
    Pursuant to directive DOH:10.2001/405aec of article B-2022,
    subsection 48.2.4.4.1c regarding staff morale and employee
    productivity standards, the current allocation of doughnut
    acquisition funds shall be increased effective immediately.
    </body>
    </html>
```

Note that because replacement occurs throughout the document, the provider need only enter the title once instead of twice (normally the title must be entered once in the HTML head and again in the HTML body). After running the script, the above file is transformed into this:

```
    <html>
    <head>
        <title> Nutritional Allocation Increase </title>
    <meta name    = "DC.Creator"
          content = "Simpson, Homer">
    <meta name    = "DC.Title"
          content = "Nutritional Allocation Increase">
    <meta name    = "DC.Date.Created"
          content = "1999-03-08">
    <meta name    = "DC.Identifier"
          content = "http://moes.bar.com/doh/homer.html">
    <meta name    = "DC.Format"
          content = "text/html;    1320  bytes">
    <meta name    = "DC.Language"
```

```
          content = "en-BUREAUCRATESE">
    <meta name    = "RC.MetadataAuthority"
          content = "Springfield Nuclear">
    <link rel     = "schema.DC"
          href    = "http://purl.org/DC/elements/1.0/">
    <link rel     = "schema.RC"
          href    = "http://nukes.org/ReactorCore/rc">
    <meta name    = "DC.Type"
          content = "Memorandum">
    </head>
    <body>
    <p>
    From:  Acting Shift Supervisor
    To:    Plant Control Personnel
    RE:    Nutritional Allocation Increase
    Date:  1999-03-08
    <p>
    Pursuant to directive DOH:10.2001/405aec of article B-2022,
    subsection 48.2.4.4.1c regarding staff morale and employee
    productivity standards, the current allocation of doughnut
    acquisition funds shall be increased effective immediately.
    </body>
    </html>
```

Here is the script that accomplishes this transformation.

```
#!/depot/bin/perl
#
# This Perl script processes metadata block declarations of the form
# <!--metablock TITLE_OF_DOCUMENT --> and variable references of the
# form (--mbVARNAME), replacing them with full metadata blocks and
# variable values, respectively.  Requires a "template" file.
# Outputs an HTML file.
#
# Invoke this script with a single filename argument, "foo".  It creates
# an output file "foo.html" using a temporary working file "foo.work".
# The size of foo.work is measured after variable replacement, and is
# later inserted into the file in such a way that the file's size does
# not change in the process.  Has little or no error checking.

$infile = shift;
open(IN, "< $infile")
        or die("Could not open input file \"$infile\"");
$workfile = "$infile.work";
unlink($workfile);
open(WORK, "+> $workfile")
        or die("Could not open work file \"$workfile\"");

@offsets = ();          # records locations for late size replacement
$title = "";            # gets the title during metablock processing
$language = "en";       # pre-set language here (not in the template)
```

```
$baseURL = "http://moes.bar.com/doh";   # pre-set base URL here also
$filename = "$infile.html";             # final output filename
$filesize = "(--mbfilesize)";           # replaced late (separate pass)

($year, $month, $day) = (localtime( (stat IN) [9] ))[5, 4, 3];
$filemodtime = sprintf "%s-%02s-%02s", 1900 + $year, 1 + $month, $day;

sub putout {            # outputs current line with variable replacement
        if (! /\(--mb/) {
                print WORK;
                return;
        }
        if (/\(--mbfilesize\)/)                 # remember where it was
                { push @offsets, tell WORK; }   # but don't replace yet
        s/\(--mbtitle\)/$title/g;
        s/\(--mblanguage\)/$language/g;
        s/\(--mbbaseURL\)/$baseURL/g;
        s/\(--mbfilename\)/$filename/g;
        s/\(--mbfilemodtime\)/$filemodtime/g;
        print WORK;
}

while (<IN>) {                          # main loop for input file
        if (! /(.*)<!--metablock\s*(.*)/) {
                &putout;
                next;
        }
        $title = $2;
        $_ = $1;
        &putout;
        if ($title =~ s/\s*-->(.*)//) {
                $remainder = $1;
        }
        else {
                while (<IN>) {
                        $title .= $_;
                        last if (/(.*)\s*-->(.*)/);
                }
                $title .= $1;
                $remainder = $2;
        }
        open(TPLATE, "< template")
                or die("Could not open template file");
        while (<TPLATE>)                # subloop for template file
                { &putout; }
        close(TPLATE);
        $_ = $remainder;
        &putout;
}
close(IN);
```

```
# Now replace filesize variables without altering total byte count.
select( (select(WORK), $| = 1) [0] );   # first flush output so we
if (($size = -s WORK) < 100000)         # can get final file size
        { $scale = 0; }                 # and set scale factor or
else {                  # compute it, keeping width of size field low
        for ($scale = 0; $size >= 1000; $scale++)
                { $size /= 1024; }
}
$filesize = sprintf "%7.7s %sbytes",
        $size, (" ", "K", "M", "G", "T", "P") [$scale];

foreach $pos (@offsets) {       # loop through saved size locations
        seek WORK, $pos, 0;             # read the line found there
        $_ = <WORK>;
        # $filesize must be exactly as wide as "(--mbfilesize)"
        s/\(--mbfilesize\)/$filesize/g;
        seek WORK, $pos, 0;             # rewrite it with replacement
        print WORK;
}

close(WORK);
rename($workfile, "$filename")
        or die("Could not rename \"$workfile\" to \"$filename\"");
# ---- end of Perl script ----
```

12. Author's Address

John A. Kunze
Center for Knowledge Management
University of California, San Francisco
530 Parnassus Ave, Box 0840
San Francisco, CA 94143-0840, USA
Email: jak@ckm.ucsf.edu
Fax: +1 415-476-4653

13. References

[AAT] Art and Architecture Thesaurus, Getty Information Institute,
http://www.gii.getty.edu/vocabulary/aat.html

[AC] The A-Core: Metadata about Content Metadata, (in progress)
http://metadata.net/ac/draft-iannella-admin-01.txt

[DC1] RFC 2413, Dublin Core Metadata for Resource Discovery,
September 1998, ftp://ftp.isi.edu/in-notes/rfc2413.txt

[DCHOME] Dublin Core Initiative Home Page,
http://purl.org/DC/

[DCPROJECTS]
Projects Using Dublin Core Metadata,
http://purl.org/DC/projects/index.htm

[DCT1] Dublin Core Type List 1, DC Type Working Group, March 1999,
http://www.loc.gov/marc/typelist.html

[freeWAIS-sf2.0]
The enhanced freeWAIS distribution, February 1999,
http://ls6-www.cs.uni-dortmund.de/ir/projects/freeWAIS-sf/

[GLIMPSE] Glimpse Home Page,
http://glimpse.cs.arizona.edu/

[HARVEST] Harvest Web Indexing,
http://www.tardis.ed.ac.uk/harvest/

[HTML4.0] Hypertext Markup Language 4.0 Specification, April 1998,
http://www.w3.org/TR/REC-html40/

[ISEARCH] Isearch Resources Page,
http://www.etymon.com/Isearch/

[ISO639-2]
Code for the representation of names of languages, 1996,
http://www.indigo.ie/egt/standards/iso639/iso639-2-en.html

[ISO8601] ISO 8601:1988(E), Data elements and interchange formats --
Information interchange -- Representation of dates and times,
International Organization for Standardization, June 1988.
http://www.iso.ch/markete/8601.pdf

[MARC] USMARC Format for Bibliographic Data, US Library of Congress,
http://lcweb.loc.gov/marc/marc.html

[PERL] L. Wall, T. Christiansen, R. Schwartz, Programming Perl,
Second Edition, O'Reilly, 1996.

[RDF] Resource Description Framework Model and Syntax Specification,
February 1999, http://www.w3.org/TR/REC-rdf-syntax/

[RFC1766] RFC 1766, Tags for the Identification of Languages,
http://ds.internic.net/rfc/rfc1766.txt

[SWISH-E] Simple Web Indexing System for Humans - Enhanced,
http://sunsite.Berkeley.EDU/SWISH-E/

[TGN] Thesaurus of Geographic Names, Getty Information Institute,
http://www.gii.getty.edu/tgn_browser/

[WTN8601] W3C Technical Note - Profile of ISO 8601 Date and Time Formats
http://www.w3.org/TR/NOTE-datetime

[XML] Extensible Markup Language (XML),
http://www.w3.org/TR/REC-xml