

Dublin Core について - 最近の動向、特に qualifier について

杉本重雄

図書館情報大学

sugimoto@ulis.ac.jp

概要

Dublin Core Metadata Element Set はインターネット上における情報資源の発見を目的として開発されてきたメタデータ記述規則であり、これまで 15 エレメントからなる基本エレメントセットを決めてきた。一方、より詳細で明確なメタデータ記述のための **qualifier** と呼ばれるメタデータ記述のための要素が以前から議論されてきている。昨年開催された第 7 回ワークショップ (DC-7) 以降、Dublin Core Metadata Initiative として推奨する **qualifier** に関する議論を進め、2000 年 7 月 **qualifier** セットをはじめてアナウンスした。今回承認された **qualifier** にはエレメントの意味をより詳細化する **Element Refinement qualifier** と、エレメントに記述する値の形式を規定する **Encoding Scheme qualifier** がある。本稿では、承認された **qualifier** を中心として、DC-7 以降の話題について述べる。

キーワード

メタデータ、Dublin Core Metadata Element Set、Dublin Core Metadata Initiative、**qualifier**、情報資源の発見

Dublin Core - Recent Development and the First Qualifier Set

Shigeo Sugimoto

University of Library and Information Science

abstract

Dublin Core Metadata Element Set is a metadata schema developed for information resource discovery on the Internet. It defines a metadata element set consisting of 15 basic elements. Qualifiers for the metadata elements have been discussed for more semantically refined description and for more accurate interpretation of metadata. After the 7th Dublin Core Workshop (DC-7) held in October 1999, the Dublin Core Metadata Initiative (DCMI) discussed the qualifiers to announce the first set of qualifiers approved for interoperability. DCMI announced the first set in July 2000. The announced qualifiers include element refinement qualifiers and encoding scheme qualifiers. The former refines or narrows the meaning of an element. The latter specifies the encoding scheme or vocabulary of a value of an element. This article presents the overview of the recent development of Dublin Core since DC-7, especially the qualifiers.

keywords

Metadata, Dublin Core Metadata Element Set, Dublin Core Metadata Initiative, Qualifier, Information Resource Discovery

1. はじめに

Dublin Core Metadata Element Set¹は、1995年頃からインターネット上の情報資源の発見を目的として開発されてきたメタデータ記述規則であり、メタデータ記述のエレメントセット（メタデータの記述要素のセット）を決めている[1]。最近では、その利用も広まってきている。Dublin Coreの開発はメーリングリストとワークショップでの議論によって進められてきている。昨年開かれた第7回ワークショップ（1999年10月、ドイツ国立図書館（フランクフルト））では、それ以前とは異なりDublin Coreの応用や今後の開発活動の方向・維持に関する議論が行われた[2]。本稿では第7回ワークショップ(DC-7と記す)以降の話題、特に2000年7月に承認されたqualifierに関して詳しく述べる。

実質的には1997年秋の第5回ワークショップ(DC-5、フィンランド国立図書館)以来、基本15エレメントからなるDublin Core Metadata Element Set (Simple Dublin Coreと呼ばれる)は固定している。この基本エレメントセットの標準化が進められており、インターネットのコミュニティのための標準として作られるRFC文書として、Simple Dublin Coreを定義したRFC2413とDublin CoreメタデータのHTML上での表現を定めたRFC2731が作られている。現在欧米での標準化機関における作業も進められており、ヨーロッパではCENによる標準化が進み、一方アメリカではNISOでの投票が本年8月に行われている[3]。

基本エレメントに加えて、以前から議論されてきたより詳細な記述のための記述子であるqualifierに関するDCMIとしての推奨セットを2000年1月にはアナウンスすることをアナウンスしたが、DC-7でのもっとも大きな事柄であった。予定よりは遅れたものの2000年7月にはじめのセットをアナウンスした。このほかDC-7では、応用や将来に向けたDublin Coreの維持管理に関するワーキンググループを立ち上げた。以下本稿では、qualifierを中心として解説する。

2. 第7回ワークショップとその後

qualifier²はエレメントの意味をより正確に示すために、エレメントの表す意味を限定、あるいはエレメントに表す値のボキャブラリや記述形式を表現するものである。DC-7での最も重要な点はDCMIとして認めたqualifierを2000年1月までにアナウンスすることをアナウンスしたことであった。このほかには、今後Dublin Coreの開発を続けていく上でのバージョン管理に関する議論、実应用到した議論を進めるためのワーキンググループの設置等がDC-7での主たる話題であった。

バージョン管理に関しては、現在決められている15エレメントからなる定義をバージョン1.1(DC 1.1)としている³。これはいわゆるSimple Dublin Coreと呼ばれるもので、そこにはqualifierの定義はまったく含まれていない。今回承認されたqualifierのセットはDC1.1に対して決められたもので、qualifierのセットそのものにはまだバージョンは与えられていない。現時点ではバージョンに関する合意はまだ得られておらず、今後の議論によることになっている。

ここでqualifierに関して少し解説する。DCMESの基本エレメントのみでメタデータを表そうとすると意味が十分に詳しく表現できないという要求が以前からあり、qualifierはかなり以前から議論されてきたものである。第4回ワークショップ(DC-4、オーストラリア国立図書館、1997年3月)では、Dublin Coreをできるだけシンプルにとどめるべきとする考え方と、内容を正確に表現するためにエレメントをより詳しく記述するための枠組みを入れるべきであるという考え方が議論された。

¹ Dublin Coreと呼ぶ場合にメタデータのエレメントセットであるDublin Core Metadata Element Set (DCMES)を意味する場合と、Element Setの開発を行ってきた活動ならびにそれを支える組織であるDublin Core Metadata Initiative (DCMI)を意味する場合がある。ここでは明確に区別する必要がある場合には前者DCMES、後者をDCMIと呼んで区別する。

² 制約子あるいは限定子と訳すのが適切とも思われるが、ここでは訳語は決めずにqualifierと記す。

³ DC 1.1とDC 1.0の間に意味的な違いはなく、表現が改訂されたのみである。4節でバージョンに関して簡単に触れる。

(前者を *minimalist*、後者を *structuralist* と呼んだ。) たとえば、論文の著者を著す場合、その人の「名前」、「所属」といったことを表したくなる。また、ある資料に関する日付を表す日付 (Date) エレメントを考えた場合、表されている日付が、この資料が「作られた日付」、「公開された日付」、「更新された日付」、「無効になる日付」などいろいろな可能性がある。したがって、こうした日付の種類を明確に表す記述方法が欲しくなる。また、主題 (Subject) エレメントの場合、表された主題語がどのような統制語彙、たとえば LCSH や MeSH、に基づいて表されているのかを書き表したい場合がある。このように要素の表す意味をより厳密にしたり、要素に与えられる値の範囲を制約したりするために用いられるものが *qualifier* である。

DC-7 の後、エレメントを議論してきたワーキンググループから *qualifier* の提案が出され、それに関する投票が進められた。最終的な投票は DC-7 での計画より遅れ 2000 年 4 月に行われた。その後 *qualifier* の定義の表現方法を改めた後 2000 年 7 月に正式にアナウンスされた[4]。承認された *qualifier* について次節で詳しく述べる。

3. *qualifier* について

3.1 概要

先に述べたように、基本 15 エレメントは DC-5 で実質的に固定し、それ以降変更はされていない。*qualifier* に関して、第 6 回ワークショップ (DC-6、アメリカ議会図書館、1998 年秋) では Dumb-Down 原理が提案された。これは、次のように *qualifier* を定義する際の原則を与えるものである。

Dumb-Down 規則：

qualifier を定義する場合、*qualifier* を含めて書き表したメタデータから *qualifier* を取り除いても値とエレメントの間に矛盾が生じてはならない。

qualifier の種類には、Element Refinement と Encoding Scheme の 2 種類がある。前者はエレメントの意味をより詳しく表すもので、後者はエレメントの値の表現方法を明示するものである。たとえば、情報資源が作成された日付や出版された日付は Element Refinement にあたる。一方、日付を yyyy-mm-dd (年-月-日) のように表すための記述形式を指示する W3C-DTF や、LCSH や MeSH のように件名 (主題) を記述するための語彙を表す名前は Encoding Scheme と呼ばれる。

Element Refinement *qualifier* と Encoding Scheme *qualifier* のより詳しい定義を下の段落に示す⁴。

Element Refinement (エレメント詳細化) *qualifier*: これはあるエレメントの意味をより狭くあるいは特化させるものである。*qualifier* によって詳細化されたエレメントは、より限定された範囲の意味を表すとはいえ、*qualifier* を用いずに書かれたエレメントと共通の意味を持たねばならない。エレメントの詳細化を意味する *qualifier* を理解できないクライアントの場合、*qualifier* を無視し、*qualifier* を持たないエレメント、すなわちより広い意味を表すエレメントとしてその値を処理できねばならない。*qualifier* に関してエレメント詳細化の定義は公開されていなければならない。

Encoding Scheme (コード化スキーム) *qualifier*: これはエレメントの値の解釈を助けるために値のスキーム (体系、仕組み) を明示するための *qualifier* である。このスキームには統制語彙 (controlled vocabulary) や、形式的記述形式あるいは構文解析規則が含まれる。したがって、Encoding Scheme を用いて表された値は統制語彙から得たトークン (たとえば、分類システムや件名標目から得た語)、あるいは形式的記述形式に適合するよう構成された文字列 (たとえば、日付の標準形式として "2000-01-01") である。ある Encoding Scheme をクライアントやエージェントが解釈できない場合であっても、そうした値は人間の読者にとって有用である。Encoding Scheme の定義記述は明確に示されねばならず、かつ公開されていなければならない。

⁴ この段落は資料[4]の当該個所の訳である。

DC-4 以来の `qualifier` に関する議論の中で、構造を持つ値を表す `qualifier` (structured value `qualifier`) を含めて議論がなされてきた。たとえば、ある人を表すためにその人の名前、所属、連絡先を組にすることが一般的に行われ、それに基づく `qualifier` も提案されてきた。しかしながら、最終的に structured value は `qualifier` から除かれることになった。次段落に、Dumb-Down に関する説明と合わせて structured value が除かれた理由を示す。

たとえば、情報資源が作成された日付(Created)と出版された日付(Available)を表すために下のよう

```
<Date><Created>2000-8-30</Created></Date>
```

```
<Date><Available>2000-9-27</Available></Date>
```

と書いたとする。ここで Created、Available はいずれも Date に対する `qualifier` である。この `qualifier` をとってしまうと、下のよう日付の種類はわからなくなるが矛盾は生じず、それぞれの日付が何らかの理由で当該資源に関連していることがわかる。

```
<Date>2000-8-30</Date>
```

```
<Date>2000-9-27</Date>
```

一方、作者(Creator)エレメントに関して、名前(Name)、所属(Affiliation)、連絡先(Contact)というサブエレメント (`qualifier`) を定義するとする。

```
<Creator><Name>杉本, 重雄</Name></Creator>
```

```
<Creator><Affiliation>図書館情報大学</Affiliation></Creator>
```

```
<Creator><Contact>sugimoto@ulis.ac.jp</Contact></Creator>
```

この 3 つのサブエレメントのうち、名前は直接作者を表すものであるが、他の二つは作者に付随する情報である。そのため、上と同じように `qualifier` を取り去ると

```
<Creator>杉本, 重雄<Creator>
```

```
<Creator>図書館情報大学<Creator>
```

```
<Creator>sugimoto@ulis.ac.jp<Creator>
```

となり、たとえば「図書館情報大学」が作者になってしまう。また、構造をより反映して

```
<Creator>
```

```
<Name>杉本, 重雄</Name>
```

```
<Affiliation>図書館情報大学</Affiliation>
```

```
<Contact>sugimoto@ulis.ac.jp</Contact>
```

```
</Creator>
```

として表現した場合、

```
<Creator>杉本, 重雄, 図書館情報大学, sugimoto@ulis.ac.jp</Creator>
```

というように書き直すことができるが、この場合は作者エレメントに表された文字列の解釈が問題になる。

structured value を表す `qualifier` の場合、何らかの対象に関する要素情報を表すことになる。たとえば、「人」という対象の要素情報「名前」、「所属」、「連絡先」である。上の例で、「所属」は対象資源に関する直接の要素情報ではなく作者である「人」の要素情報であるために、Dumb-Down がうまくいかなかったと言うことができる。一方、作者エレメントとしては「人」である何かを記述すればよいので、次のような記述方法が考えられる。

- ・ 名前を書く。(「人」が持つ属性としてではなく、「人」そのものを表すものとして)
- ・ ある個人を表す何らかの識別子や識別名を書く。
- ・ Dublin Core の `qualifier` とは別の枠組みで決める要素セットを用いて表す。(たとえば、人を表すためのエレメントセットを準備し、それを用いる。)

Structured value を含めないことにしたのは、これまでにワーキンググループから提案された `qualifier` の一部を否定することになった。また、この結論を導く上で Resource Description Framework(RDF)が提案した情報資源記述のためのモデルが下敷きになっている⁵。

⁵ RDF の記述モデルは、資源、属性、値の 3 つ組みで記述するものである。たとえば、「この論文の著者は杉本重雄である」場合、資源 = 「この論文」、属性 = 「著者 (作者)」、値 = 「杉本重雄」である。「杉本重雄」が属性を持つ場合、たとえば、「杉本重雄の名前は「杉本, 重雄」である。」「杉本重雄の所属は図書館情報大学である。」というようにとらえる。詳しくは資料[5]を参照されたい。

表1 DCMI Qualifier の一覧

(<http://purl.org/dc/documents/rec/dcmes-qualifiers-20000711.htm> より)

DCMES Element	Element Refinement(s)	Element Encoding Scheme(s)
Title	Alternative	—
Creator	—	—
Subject	—	LCSH MeSH DDC LCC UDC
Description	Table Of Contents Abstract	—
Publisher	—	—
Contributor	—	—
Date	Created Valid Available Issued Modified	DCMI Period W3C DTF
Type	—	DCMI Type Vocabulary
Format	Extent	—
	Medium	IMT
Identifier	—	URI
Source	—	URI
Language	—	ISO 639-2 RFC 1766
Relation	Is Version Of Has Version Is Replaced By Replaces Is Required By Requires Is Part Of Has Part Is Referenced By References Is Format Of Has Format	URI
Coverage	Spatial	DCMI Point ISO 3166 DCMI Box TGN
	Temporal	DCMI Period W3C-DTF
Rights	—	—

3.2 承認された qualifier について

表1に承認された qualifier の一覧を示す。また、以下に承認された qualifier の定義を示す。これに加えてタイプエレメントの統制語彙としてDCMIが用意したDCMI Type Vocabularyも示す。これらはそれぞれ資料[4]と[6]の当該部分の翻訳である。なお、下の記述において各 qualifier に関して下

記の属性を示す。

- ・ 名前 (Name) : qualifier を一意に表すトークン (token)。
- ・ ラベル (Label) : qualifier に割り当てられる人間が読む目的で用意されるラベル。
- ・ 定義 (Definition) : qualifier の概念もしくは本質的な性質を表す文。
- ・ コメント (Comment) : qualifier に関する付加的な情報の記述 (必須ではない)。
- ・ をも見よ (See Also) : qualifier に関するより多くの情報へのリンク (必須ではない)。

タイトル (Title) エlement

- ・ タイトルElementの意味を詳細化するもの (Element Refinement)

Alternative

名前: alternative

ラベル: Alternative

定義: 対照情報資源の正式なタイトルの代わりもしくは別タイトルとして利用されるタイトル。

コメント: この qualifier はタイトルの翻訳と同じく省略形を表すのに用いてもかまわない。

主題 (Subject) Element

- ・ 主題Elementの値の記述形式を与えるもの (Encoding Schemes)

LCSH

名前: LCSH

ラベル: LCSH

定義: Library of Congress Subject Headings

MeSH

名前: MESH

ラベル: MeSH

定義: Medical Subject Headings

をも見よ: <http://www.nlm.nih.gov/mesh/meshhome.html>

DDC

名前: DDC

ラベル: DDC

定義: Dewey Decimal Classification

をも見よ: <http://www.oclc.org/dewey/index.htm>

LCC

名前: LCC

ラベル: LCC

定義: Library of Congress Classification

をも見よ: <http://lccweb.loc.gov/catdir/cpsolcco/lcco.html>

UDC

名前: UDC

ラベル: UDC

定義: Universal Decimal Classification

をも見よ: <http://www.udcc.org/>

内容記述 (Description) Element

- ・ 内容記述Elementの意味を詳細化するもの (Element Refinement)

Table Of Contents

名前: tableOfContents

ラベル: Table Of Contents

定義: 対象情報資源の内容の構成要素のリスト

Abstract

名前: abstract

ラベル: Abstract

定義: 対象情報資源の内容の要約

日付 (Date) エレメント

- ・日付エレメントの意味を詳細化するもの (Element Refinement)

Created

名前: created

ラベル: Created

定義: 対象情報資源が作成された日付

Valid

名前: valid

ラベル: Valid

定義: 対象情報資源の有効性を表す日付 (期間を表す場合もある)

Available

名前: available

ラベル: Available

定義: 対象情報資源が利用可能になる、あるいは利用可能になった日付 (期間を表す場合もある)

Issued

名前: issued

ラベル: Issued

定義: 対象情報資源の正規に発行された (例: 出版) 日付

Modified

名前: modified

ラベル: Modified

定義: 対象情報資源が更新された日付

- ・日付エレメントの値の記述形式を与えるもの (Encoding Schemes)

DCMI Period

名前: Period

ラベル: DCMI Period

定義: ある時間的区間の両端を与える仕様

をも見よ: <http://purl.org/dc/documents/dcmi-period>

W3C-DTF

名前: W3CDTF

ラベル: W3C-DTF

定義: W3C が決める日付と時間の記述形式。ISO8601 に基づいて決められたプロファイル

をも見よ: <http://www.w3.org/TR/NOTE-datetime>

タイプ (Resource Type) エレメント

- ・タイプエレメントの値の記述形式を与えるもの (Encoding Schemes)

DCMI Type Vocabulary

名前: DCMIType

ラベル: DCMI Type Vocabulary

定義: 対照情報資源の内容の性質もしくはジャンルを分類するために用いられるタイプのリスト

をも見よ: <http://purl.org/dc/documents/dcmi-type-vocabulary> (本節の最後にリストを示す。)

フォーマット (Format) エレメント

- ・フォーマットエレメントの意味を詳細化するもの (Element Refinement)

Extent

名前: extent

ラベル: Extent

定義: 対象情報資源のサイズもしくは持続期間 (時間)

Medium

名前: medium

ラベル: Medium

定義: 対象情報資源の素材、もしくは対象情報資源を格納する物理的入れ物

- ・フォーマットエレメントの値の記述形式を与えるもの (Encoding Schemes)

IMT

名前: IMT

ラベル: IMT

定義: 対照情報資源のインターネットメディアタイプ (Internet media type)

をも見よ: <http://www.isi.edu/in-notes/iana/assignments/media-types/media-types>

資源識別子 (Resource Identifier) エレメント

- ・資源識別子エレメントの値の記述形式を与えるもの (Encoding Schemes)

URI

名前: URI

ラベル: URI

定義: Uniform Resource Identifier

をも見よ: <http://www.ietf.org/rfc/rfc2396.txt>

言語 (Language) エレメント

- ・言語エレメントの値の記述形式を与えるもの (Encoding Schemes)

ISO 639-2

名前: ISO639-2

ラベル: ISO 639-2

定義: ISO 639-2: 言語名を表すためのコード

をも見よ: <http://lcweb.loc.gov/standards/iso639-2/langhome.html>

RFC 1766

名前: RFC1766

ラベル: RFC 1766

定義: Internet RFC 1766 “言語を識別するためのタグ”、ISO 639 からとられた 2 文字のコードに続けて ISO 3166 からとられた国を表す 2 文字のコード。

をも見よ: <http://www.ietf.org/rfc/rfc1766.txt>

関係 (Relation) エレメント

- ・関係エレメントの意味を詳細化するもの (Element Refinement)

Is Version Of

名前: isVersionOf

ラベル: Is Version Of

定義: 対象情報資源がこのエレメントで参照される資源のあるバージョンであるか、あるエディションであるか、あるいは翻案 (adaptation) であることを意味する。バージョンの変更はフォーマットの変換ではなく内容の本質的な修正を意味する。

Has Version

名前: hasVersion

ラベル: Has Version

定義: 対象情報資源が参照された資源をあるバージョン、エディション、あるいは翻案として持つことを意味する。

Is Replaced By

名前: isReplacedBy

ラベル: Is Replaced By

定義: 対象情報資源は参照された資源によって取って代わられた、置き換えられた、あるいは取り替えられたことを意味する。

Replaces

名前: replaces

ラベル: Replaces

定義: 対照情報資源が参照された資源に取って代わる、置き換わる、あるいは成り代わることを意味する。

Is Required By

名前: isRequiredBy

ラベル: Is Required By

定義: 対象情報資源が参照された情報資源によって物理的あるいは論理的に要求されることを意味する。

Requires

名前: requires

ラベル: Requires

定義: 対象情報資源がその機能、提供、あるいは内容の首尾一貫性のために参照された資源を必要とすることを意味する。

Is Part Of

名前: isPartOf

ラベル: Is Part Of

定義: 対象情報資源が参照された資源の物理的あるいは論理的な部分であることを意味する。

Has Part

名前: hasPart

ラベル: Has Part

定義: 対照情報資源が参照された資源を物理的あるいは論理的な一部分として含むことを意味する。

Is Referenced By

名前: isReferencedBy

ラベル: Is Referenced By

定義: 対象情報資源が参照された資源から参照される、引用される、あるいは指し示されることを意味する。

References

名前: references

ラベル: References

定義: 対照情報資源が参照された資源を参照する、引用する、あるいは指し示すことを意味する。

Is Format Of

名前: isFormatOf

ラベル: Is Format Of

定義: 対照情報資源と参照された資源は同じ知的内容であるが、異なるフォーマットで表現されていることを意味する。

Has Format

名前: hasFormat

ラベル: Has Format

定義: 対象情報資源が参照された資源より以前に存在し、参照された資源は対象資源と同一の知的内容を別のフォーマットで表しているものであることを意味する。

- ・ 関係エレメントの値の記述形式を与えるもの (Encoding Schemes)

URI

名前: URI

ラベル: URI

定義: Uniform Resource Identifier

をも見よ: <http://www.ietf.org/rfc/rfc2396.txt>

時間的・空間的範囲 (Coverage) エレメント

- ・ 範囲エレメントの意味を詳細化するもの (Element Refinement)

Spatial

名前: spatial

ラベル: Spatial

定義: 対象情報資源の知的内容に関する空間的特性

Temporal

名前: temporal

ラベル: Temporal

定義: 対象情報資源の知的内容の時間的属性を表す。

Spatial、Temporal それぞれに Encoding Scheme が定義されている。下にはそれぞれの Encoding Scheme を示す。

- ・ Spatial qualifier の値の記述形式を与えるもの (Encoding Schemes)

DCMI Point

名前: Point

ラベル: DCMI Point

定義: 地理的なある地点を識別するための地理的座標

をも見よ: <http://purl.org/dc/documents/dcmi-point>

ISO 3166

名前: ISO3166

ラベル: ISO 3166

定義: 国名を表すための ISO 3166 コードによる表現

をも見よ: <http://www.din.de/gremien/nas/nabd/iso3166ma/codlstp1/index.html>

DCMI Box

名前: Box

ラベル: DCMI Box

定義: 地理的な境界によって示されるある領域を表す。

をも見よ: <http://purl.org/dc/documents/dcmi-box>

TGN

名前: TGN

ラベル: TGN

定義: 地理的な名前に関する Getty Thesaurus

をも見よ: http://shiva.pub.getty.edu/tgn_browser/

(以上、Spatial qualifier のための Encoding Scheme)

- ・ Temporal qualifier の値の記述形式を与えるもの (Encoding Schemes)

DCMI Period

名前: Period

ラベル: DCMI Period

定義: ある時間的区間の両端を与える仕様

をも見よ: <http://purl.org/dc/documents/dcmi-period>

W3C-DTF

名前: W3CDTF

ラベル: W3C-DTF

定義: W3C が決める日付と時間の記述形式。ISO8601 に基づいて決められたプロファイル

をも見よ: <http://www.w3.org/TR/NOTE-datetime>

(以上、Temporal qualifier のための Encoding Scheme)

DCMI Type Vocabulary

qualifier の承認と同時に、タイプエレメントを記述するのに用いる統制語彙を決めた。それが DCMI Type Vocabulary である [6]。DCMI Type Vocabulary に含まれる語は、Collection、Dataset、Event、Image、Interactive Resource、Service、Software、Sound、Text の 9 語である。以下に各語の定義を示す。なお、記述方法は上の方法に従う。

1. Collection

名前: Collection

ラベル: Collection

定義: Collection は個々のものの集まりである。Collection という語は対象情報資源がグループとして記述されることを意味し、したがって、Collection の要素は個別に記述され、かつそこに到達可能である。

2. Dataset

名前: Dataset

ラベル: Dataset

定義: Dataset はなんらかの定義された構造（たとえば、リスト、テーブル、データベース）にコード化された情報であり、コンピュータによる直接の処理を目的としたものである。

3. Event

名前: Event

ラベル: Event

定義: Event は非永続的で、かつ時間に依存して生じるものである。Event に対するメタデータは目的、場所、期間、責任を持つエージェント、ならびに関連する Event や情報資源の発見の基礎となる記述的な情報を与えるものである。Event タイプのリソースは記述の対象となったものの有効期限が過ぎていたり、あるいはまだ実際に起きていない場合には、検索不可能であることも有得る。たとえば、展示会、Web 上での放送、会議、open-day、パフォーマンス、戦闘、裁判、婚礼、お茶会、火災などが例である。

4. Image

名前: Image

ラベル: Image

定義: Image はテキスト以外の形式で表される基本的にシンボリックな視覚的表現である。たとえば、物理的な物体のイメージや写真、絵画、版画、図、その他のイメージやグラフィックス、アニメーションや動画、フィルム、図式、地図、楽譜などである。Image には電子的なものも物理的なものも含まれることに注意。

5. Interactive Resource

名前: InteractiveResource

ラベル: Interactive Resource

定義: Interactive Resource は、その内容を理解するため、実行するため、また利用してみるために利用者との対話（利用者による操作）を必要とするものである。たとえば、Web ページの中の form、applet、マルチメディアで作成された学習用の資料、チャットサービス、仮想現実などである。

6. Service

名前: Service

ラベル: Service

定義: Service はエンドユーザに対して何らかの価値を提供するシステムである。たとえば、コピーサービス、バンキングサービス、証明サービス、図書館間相互貸借 (ILL)、Z39.50 や Web サーバなどである。

7. Software

名前: Software

ラベル: Software

定義: Software は別のマシンにインストールして利用するために用意されたソースプログラムあるいはコンパイルされたプログラムを意味する。対話環境を作り出すためにのみ用いられるソフトウェアについては、Software の代わりに Interactive Resource を用いる。

8. Sound

名前: Sound

ラベル: Sound

定義: Sound は、その内容が音響・音声として再生されることを第 1 の目的として作られた情報資源である。たとえば、音楽再生用のファイル形式、オーディオ・コンパクトディスク、録音されたスピーチや音などである。

9. Text

名前: Text

ラベル: Text

定義: Text は、その内容を読むために用意された語（の並び）である資源である。たとえば、本、手紙、学位論文、詩、新聞、記事、メーリングリストのアーカイブなどである。テキストのフクシミリやイメージはテキストとして扱う。

4. 今後の開発について

はじめての qualifier の承認が終わり一段楽した感じがある一方、将来に向けた話題として次のような点が議論の対象となっている。

(1) バージョンについて

現時点で基本 15 エレメントについてバージョン 1.0 と 1.1 があるが、これは参照記述の表現を改めたものに過ぎないので実質的には単一のバージョンしか存在していない。また、qualifier セットに関しても現時点ではひとつのバージョンしかない。今後、利用経験が増えるにしたがってエレメントや qualifier の改訂の要求が出てくることが考えられる。長期に渡る Dublin Core メタデータの安定した利用を考えると、一貫したデータの利用を可能にするためのバージョン定義の必要性が認められている。しかしながら現時点では、バージョンの定義や今後のバージョンの管理の方針などについての結論はまだ出ていない。

(2) qualifier の追加と今後の維持管理に関して

今回承認された qualifier は、Dublin Core を利用するグローバルなコミュニティ全体にとって現時点で共通して有用であるものであることを、DCMI として認めたものである。今後の Dublin Core の利用の拡大、利用経験の増加に基づき新しい qualifier の利用の要求が出てくることが予想できる。そのため、現時点では、qualifier を承認していくための下に示す 3 段階のプロセスが DCMI によって提案されている。

- ・ 第 1 段階：ある qualifier がローカルなコミュニティ（地理的に形作られるコミュニティ、応用分野毎のコミュニティ）の中で認知され、利用される段階。
- ・ 第 2 段階：その qualifier がローカルなコミュニティからグローバルなコミュニティによる利用のために DCMI に提出され、グローバルなコミュニティにおける相互利用性のために推奨できるかどうかはわからないが、Dublin Core の qualifier として問題ないと DCMI によって認められた段階。
- ・ 第 3 段階：その qualifier が DCMI の中に設けられた利用に関する委員会（Usage Committee）によって審査され、Dublin Core のグローバルなコミュニティとしての相互利用性のために推奨するに値すると判断した段階。

こうした段階に加えて qualifier の中には利用されなくなり、無効であるとされるものもでてくる。

コミュニティにおける新しい qualifier に関する合意の形成方法、それらのグローバルなコミュニティへの提案方法等、今後の実践の中で明らかになっていくものと思われる。たとえば、日本語で書くメタデータの場合、名前の読みは必要な情報であるが、現在の qualifier には含まれていない。「読み」を表す qualifier をグローバルなコミュニティで利用できるようにするには、日本語のメタデータを定義、利用するコミュニティを形成し、そこでの議論を元に DCMI に提案する必要がある。

(3) レジストリに関して

今後、Dublin Core を長期にわたって実利用していくためには、Dublin Core のエレメントや qualifier の定義を蓄積し、ネットワークを介して提供するサービスが必要とされる。そのサービスを Dublin Core レジストリと呼ぶ。Dublin Core レジストリに格納されるデータには、DCMI がグローバルなコミュニティでの利用を推奨するものから、ローカルなコミュニティでの利用のみのものまである。そのため、グローバルなコミュニティとローカルなコミュニティの両方に利用できるものが必要がある。また、利用者は世界中に広がるので、各国語での提供も望まれる。エレメントや qualifier の定義を各国語に翻訳するにはそれなりの時間がかかる。そのため、言語によって翻訳されたもののバージョンが異なることも有得る。こうした問題を議論するため、レジストリに関するワーキンググ

ループが構成され、どのような機能が要求されるのかの議論を進めている。筆者等はレジストリに関する研究開発を進めているが、それに関しては別稿[7]に任せたい。

5. おわりに

最近、Dublin Core に基づいてメタデータの枠組みを決めていることをしばしば耳にするようになってきた。その使い方いろいろのようである。これは、Simple Element とはいっても、15 のエレメントそれぞれがかなり一般的に定義されているため、15 エレメント全体ではカバーする範囲が広く、個々の応用には必ずしも全てのエレメントが用いられないことを意味していると言えよう。また、Dublin Core に基づいて新たにデータを作ることはせず、Dublin Core を異なるメタデータ規則に基づくメタデータを横断的に見るための窓として利用することもできる。

一方、教育用資料の場合に必要な「利用者の種類」を表すエレメントのように[8]、本質的に15 エレメントには含まれないものもある。こうした情報については現在のエレメント定義では扱えず、Dublin Core と他のメタデータ規則を組み合わせて記述しなければならない。こうした応用に依存して必要とされる情報については複数の規則を組み合わせて記述していくことになるのであろうと筆者は考えている。また、そうした応用ごとのメタデータ記述の規則を定義する application profile が必要とされることになるのであろう。

第8回ワークショップ(DC-8)は2000年10月、カナダ国立図書館で開催されることになっている。これまでのワークショップが参加のための招待を必要としたのに対し、DC-8 ははじめて一般に公開されたワークショップとなる。その意味ではこれまでとは異なり、Dublin Core に基づくシステムの開発者、利用者が集まる Conference 的色彩の会議となることが期待されている。一方、これまで OCLC を中心として草の根参加者によって支えられてきた組織を、今後の長期にわたる維持・運営に適した組織に作り変えたいという動きもある。開発の開始から6年近く経過して利用が広がるとともに、変わり目にあるように思われる。

参考文献

- [1] Dublin Core Metadata Initiative, <http://purl.org/dc/>
- [2] 杉本重雄, Dublin Core に関する最近の話題から—第7回 Dublin Core ワークショップほか, デジタル図書館(ISSN 1340-7287), No.17, pp.32-36, 2000.2,
http://www.dl.ulis.ac.jp/DLjournal/No_17/3-sugimoto/3-sugimoto.html
- [3] Draft Standard Z39.85-200X The Dublin Core Metadata Element Set,
<http://www.niso.org/Z3985.html>
- [4] Dublin Core Qualifiers, <http://purl.org/DC/documents/rec/dcmes-qualifiers-20000711.htm>, 2000.7
- [5] Resource Description Framework (RDF) Model and Syntax Specification,
<http://www.w3.org/TR/REC-rdf-syntax/>, 1999.2
- [6] DCMI Type Vocabulary, <http://purl.org/dc/documents/rec/dcmi-type-vocabulary-20000711.htm>, 2000.7
- [7] 永森光晴他, RDF Schema に基づくメタデータレジストリ, 情報学基礎研究会, 2000.9
- [8] Education Working Group: Draft Proposal, <http://purl.org/DC/documents/wd/education-20000430.htm>, 2000.4