

博物館情報横断検索のための記述内容レベル相互変換

画像電子学会 画像ミュージアム研究会 博物館・美術館 DTD-SG

山田 篤[†] 安達 文夫[‡] 小町 祐史^{*}

Atsushi YAMADA[†] Fumio ADACHI[‡] and Yushi KOMACHI^{*}

[†] 京都高度技術研究所

[†] ASTEM RI/Kyoto

[‡] 国立歴史民俗博物館

[‡] National Museum of Japanese History

* パナソニックコミュニケーションズ

* Panasonic Communications Co., Ltd.

1. はじめに

画像電子学会画像ミュージアム研究会「博物館・美術館文書の文書型定義 SG (DTD-SG)」では、博物館、美術館が公開する文書情報の横断検索について検討している。本研究グループが提唱する

- I) 情報記述構造レベル
- II) 情報記述内容レベル
- III) 情報ナビゲーションレベル

からなる 3 層フレームワークについては[1][2]で報告し、このうち、レベル I の記述構造レベルに関するプロトタイプとその評価について[3][4]で報告した。本稿ではこれらの考察をもとにして、次の段階として記述内容レベルの横断検索について述べる。

2. 記述内容レベルの相互変換の必要性

2.1 記述構造レベルの相互変換の限界

我々の提唱するフレームワークでは、各館の独自性、多様性を許容しながら、様々なレベルでの情報の相互変換を行うことにより、複数の館が公開するそれぞれの館蔵品に関する情報(文書)を横断検索し、検索者が望む情報を得ることを可能にすることを目的としている。

各館が公開する文書としては、データベースに格納することを前提としたメタデータ形式のものもあれば、解説文書へのアノテーション形式のものもあり、また扱う対象、専門分野による違いもあり得るが、記述構造レベルの相互変換により、これらを共通に検索対象とすることができる。

たとえば、[4]ではこれらの構造変換ターゲットとして Dublin Core Metadata Element Set V1.1 (DCMES)[5]を設定し、データベースのスキーマレベルで複数の館の公開する文書情報を横断検索する方式について述べた。これにより、title や creator といった共通のデータ構造を用いた検索が可能になるが、検索対象となる各フィールド

に書かれている内容については、記述構造のレベルでは取り扱えないため、対象としていなかった。この結果、たとえ同じ構造に変換できたとしても

- (1) 同じ内容に対し、複数の館で異なる表記をしていた場合
- (2) 異なる内容に対し、偶々同一の表記をしていた場合

に対応できないという問題がある。(1)の場合は本来ヒットすべきものが見つからないというエラーに、(2)の場合は本来ヒットすべきでないものが検索結果に含まれるというエラーになる。記述内容の異同の判定には、後述するように比較的単純なものから意味内容に関する高度な判断を要するものまでが含まれるが、いずれにしてもこの問題に対処するためには、レベル II の記述内容レベルの処理が必要となる。

2.2 語彙の共有

記述内容レベルでの横断検索を行うためには、そこで用いられている語彙の意味内容について、共通の理解が必要となる。これらの語彙を共有する手段としては、

- (a) はじめから同一の語彙体系を用いる
- (b) 異なる語彙体系の間で、必要に応じて可能な限りで相互変換を行う

というアプローチが考えられる。

(a)のアプローチは比較的小規模なコミュニティや特定の業界内部において用いる語彙の標準化などでは有効であるが、オープンな環境においてはユニバーサルな語彙体系を構築しなければならないという困難に直面する。また、その時々で必要とされる知識は異なることが考えられ、また多様な観点を統合するという目的からも、ここでは(b)のアプローチをとる。

さらに、単一の館を対象とした検索であっても、利用者の検索要求と情報提供側の語彙の不一致による検索の失敗は考えられ、そのような場合にもこのアプローチは有効である。

3. 記述内容レベルの相互変換

構造化されたフィールドに書かれている内容に関する相互変換の例として、以下のようなものが考えられる。

1) 異表記、alias 等の置き換え

漢字とかななど、単純に語彙のレベルで変換が可能なものがある。これはそういった語彙に関する辞書を用意しておくことで対応が可能である。多言語対応も単純なレベルではこういった対応で十分な場合がある。また、時間に関する情報を世紀で表記しているか、時代名で表しているか、あるいは年号を用いているか、年号は西暦か元号か、といった違いも同様の変換でほぼ吸収できると考えられる。これらは記述の表記（インスタンス）レベルの変換である。

2) 分類に基づく検索

たとえば狩野派の作品を探しているとしよう。このとき、各館の原データには作者名の記述しかないとする。狩野派というクラスと銘々の作者名（インスタンス）との対応関係を提供する知識源が必要となる。古伊万里の器を探している場合も同様である。この場合は、時代、産地、さらには対象物の分類（クラス）情報を統合した検索となる。一般的にはクラスとインスタンスの対応関係が問題となる。

3) 異なる分類の統合

単一の知識源から提供される情報ですべてを尽くすことができればよいが、ユニバーサルな知識源の実現は難しい。また、個々の知識源はそれぞれ特定の見方を反映していると考え、それらを統合することで利用価値を高めることができるかもしれない。このとき、異なる分類（クラス階層）の対応関係を解決する必要が生じる。

4. 記述内容レベル相互変換のアーキテクチャ

前節で述べたような記述内容レベルの相互変換を行う際に必要となる知識源は、構造レベルの変換時に用いた変換テーブルのような単純なものでは間に合わない。少なくとも同義語やクラスの定義、クラスとインスタンスの関係を記述できる必要がある。これは専門家によるその専門分野に関する（部分的な）知識を記述するための仕組みとみなすことができる。専門知識の提供は各館が行ってもよいし、館とは独立に第三者が行ってもよい。

このような知識表現のための仕組みとしてオントロジと呼ばれる枠組みがあり、たとえば OWL[6]等の記述方法が検討されている。ここでは、専門家がいかにしてオントロジを構築するかという問題とは別に、ある専門家の知識を表現したオントロジがあった場合に、それを用いて記述内容レベルの相互変換を行うための基本的な

アーキテクチャについて検討する。

特定のオントロジを用いて複数館の横断検索を行う場合は、そのオントロジで規定されているクラスと各館の提供するインスタンスに関する情報との対応付けができればよい。最終的にはあるインスタンスが特定のクラスに属するための要件を満たすか否かを判定することになる。その結果、特定のクラスと対応付けられたインスタンスを検索結果として取り出す。

次に、複数のオントロジを統合した横断検索の場合は、これらのオントロジを仲介して、クラス階層間の対応関係を解決する仕組みが必要になる。一つ一つのオントロジはそれぞれインスタンスとの対応を持っているので、仲介者はあるオントロジの特定のクラスに対応付けられているインスタンスと、別のオントロジで先のクラスと equivalent なクラスに対応付けられているインスタンスをあわせたものを検索結果とする。

6. おわりに

本稿では、博物館、美術館が公開する文書情報の横断検索を目指して、共通の記述構造にマッピングした後に、記述内容レベルでの相互変換を行うことにより、各館毎に異なる表記を採用していても、それらの間の横断検索を可能とする方式について検討した。記述内容レベルの相互変換のための知識表現としてはオントロジを用いることにより、異なる知識源の統合も視野に入れた記述内容レベルの相互変換を行える可能性がある。

今後、具体例を用いたプロトタイピングを通じて、さらにこのフレームワークに関する検討を重ねていく予定である。

文 献

- [1] 画像電子学会 VMA 研究会 博物館・美術館 DTD-SG, “博物館情報の知的横断検索のためのフレームワーク,” 2002 画像電子学会第 30 回年次大会画像電子ミュージアムテクニカルセッション, pp.75-76, Jun. 2002.
- [2] 画像電子学会 VMA 研究会 博物館・美術館 DTD-SG, “博物館情報の知的横断検索の試み,” 2002 画像電子学会第 30 回年次大会画像電子ミュージアムテクニカルセッション, pp.77-78, Jun.2002.
- [3] 画像電子学会 VMA 研究会 博物館・美術館 DTD-SG, “博物館情報横断検索その役割と課題,” 画像電子学会第 10 回 VMA 研究会, Jan.2003.
- [4] 画像電子学会 VMA 研究会 博物館・美術館 DTD-SG, “博物館情報横断検索のための記述構造相互変換のプロトタイピングと評価,” 画像電子学会第 10 回 VMA 研究会, Jan.2003.
- [5] Dublin Core Metadata Element Set (<http://www.dublincore.org/usage/terms/dc/current-elements/>).
- [6] OWL Web Ontology Language (<http://www.w3.org/TR/owl-ref/>).