人文系データベース横断検索のためのメタデータ自動マッピング

鳥羽 拓志
立命館大学
理工学研究科

木村 文則
立命館大学
情報理工学部

手塚 太郎
立命館大学
情報理工学部

前田 亮
立命館大学
情報理工学部

近年，保持している資料をディジタル化し公開している機関が増加している．その中で公開されているデータベースへの横断検索の研究が幾つかされている．本研究では，横断検索の実現に際しての課題であるデータ項目名のメタデータ要素へのマッピングを自動で行うことによって課題を克服することを目的とする．自動化は，Web 上で公開されているデータベースから集めた各メタデータのデータ項目名セットに対する一致と正規表現を用いたルールの適用によって行う．マッピングするメタデータ要素には Dublin Core を使用した．自動マッピングの結果と人手によるマッピングの比較から正解率を求める実験を行った結果，ある程度の有効性が得られた．

Automatic Metadata Mapping for Federated Search of Humanities Databases

Takushi Toba
Graduate School of Science and Engineering, Ritsumeikan University

Fuminori Kimura
College of Information Science and Engineering, Ritsumeikan University

Taro Tezuka
College of Information Science and Engineering, Ritsumeikan University

Akira Maeda
College of Information Science and Engineering, Ritsumeikan University

Recently, an increasing number of organizations are digitizing their holding materials and making them available on the web. There are some researches on federated search of these databases. A major problem in realizing federated search of these heterogeneous databases is that most of the databases use their own metadata schema. In this study, we aim at solving this problem by automatically mapping the metadata elements with various different names into a standard metadata element set. This automatic mapping is done by the combination of two methods: 1) similarity matching between element names collected from various humanities databases available on the web, and 2) rule-based matchingusing regular expression. We conducted a preliminary experiment to test the accuracy of mapping the metadata elements of 50 Japanese humanities databases into Dublin Core metadata element set, and we achieved adequate effectiveness.

1．はじめに

近年，様々な図書館や美術館，研究機関が所蔵している資料をディジタル化し，公開している．多種多様なデータベースがある中で，ユーザが求めている情報を見つけ出すには時間と手間がかかるのが現状である．そこでこれらのデータベースに対し，個別にアクセスするのではなく複数のデータベースに対して一度に検索を行う横断検索の研究がなされている．

図 1 はデータベースの検索画面例を示す．しかし，実際に横断検索を行おうとすると，図 1 のようにデータベースごとに要求される入力項目やその入力形式が異なるため，一度の入力で全てのデータベースを横断検索することは困難である．そこで横断検索の実現には，各データベースのデータ項目名にメタデータ要素を付与したり，データの提供者が横断検索のシステムに合ったプロトコルを用意したり，横断検索のシステムに実際にデータを登録する必要がある．

本研究では，一度の問合せの入力で，ユーザが選択した全てのデータベースを検索できる横断検索システムの構築を目指している．横断検索の実現のために，データ項目名に対してメタデータ要素の自動マッピングを行う手法を提案する．これにより，データの提供者の手間やユーザの検索の手間を減らすことができる．

2．既存の横断検索システム

現在，横断検索のシステムは大きく分けて２つのタイプがある．分散型[1]と集中型 [2]である．

分散型は，横断検索のシステムが Web 上にあるデータベースにアクセスし検索を行うもので，実際にシステムがデータを持つことなく横断検索を実現できる．データのやり取りには各種通信プロトコル（Z39.50 や SRW/SRU など）を使用する．

集中型は，横断検索のシステム内に検索したいデータベースのデータを集めることによって，ネットワークの制限を受けることなく横断検索を実現することができる．

本研究では，データベースのデータ項目に対してメタデータ要素の自動マッピングを行うことにより．ユーザが検索したいデータベースを指定できるような，集中型ではなく分散型の横断検索システムを実現することを目標とする．

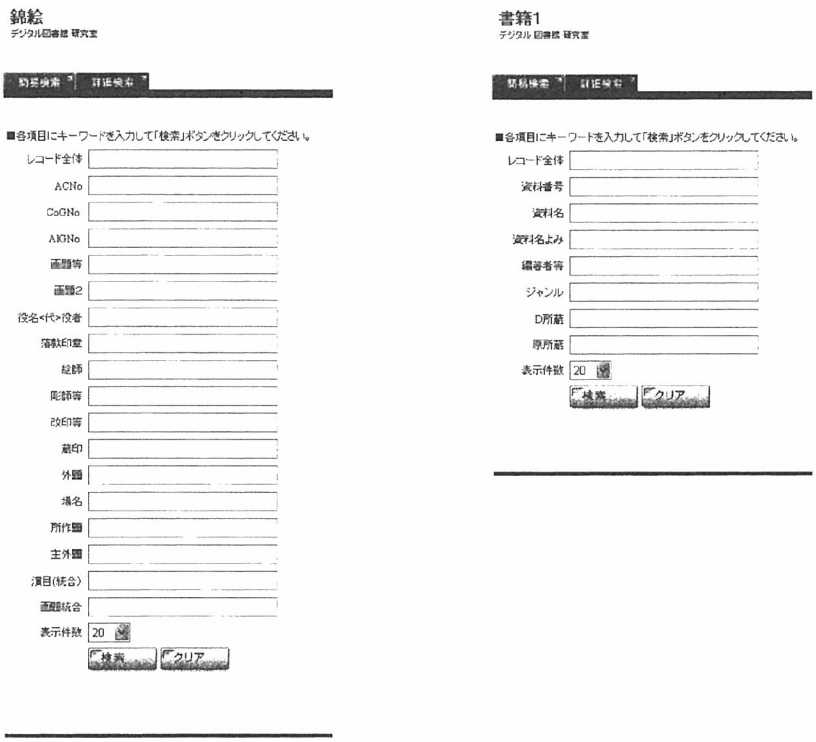

図１　検索画面例

3. 対象とするデータベース

多くの人文科学に関する資料が電子化され公開されている．そして，それらのデータベースに対する横断検索の研究も行われている．その中で共通のメタデータにデータベースの個々のデータ項目名を対応付けることは課題とされている[3].

人文系のデータベースは，公開する資料に関するデータ項目名をつけることが多く，関係性の深いデータベースでは共通したデータ項目名が使われていることが多い．また公開するデータベースはユーザの利用を想定して作られているため，ユーザに分かりやすいデータ項目名をつけていることが多い．そのため，使用されるデータ項目名は比較的類似していることが多くなると考えられる．

そこで本研究では人文系のデータベースから共通メタデータに該当するデータ項目名を収集して，そのデータ項目名の集合を作成する．その集合と実際にメタデータ要素にマッピングしたいデータ項目名とを比較することによってメタデータ要素を特定する．

またデータ項目名中に特定の語が含まれていれば，その語に対応したメタデータ要素へマッピングされる可能性が高い（たとえばデータ項目名中に～版～という語があれば Publisher の可能性が高い）と考えられる．

表 1 は本研究で用いるルールの一覧を示す．本研究では，表 1 のように，特定の単語の使用位置において，特定のメタデータ要素となる可能性が高くなるように考慮したルールを用いてマッピングの精度が高まるかどうかについても検討した．

共通のメタデータには汎用性の高い Dublin Core [4]の基本 15 要素（Title, Creator, Subject, Description, Publisher, Contributor, Date, Type, Format, Identifier, Source, Language, Relation, Coverage, Rights）を使用することにした．

表1　メタデータ要素の特徴を考慮するルール一覧

ルール	スコア
「～名」で終わっている	Title+1,Creator+0.5,Publisher+0.5,Contributor+0.5
「者」を含んでいる	Creator+1,Publisher+1,Contributor+1
「訳」を含んでいる	Contributor+1
「版」を含んでいる	Publisher+1
「ID,No」を含んでいる	Identifier+1
「暦」を含んでいる	Coverage+1
「地」を含んでいる	Coverage+1
「～年」で終わっている	Coverage+2,Date+2
「言語」を含んでいる	Lnaguage+2
「～番号」で終わっている	Identifier+2

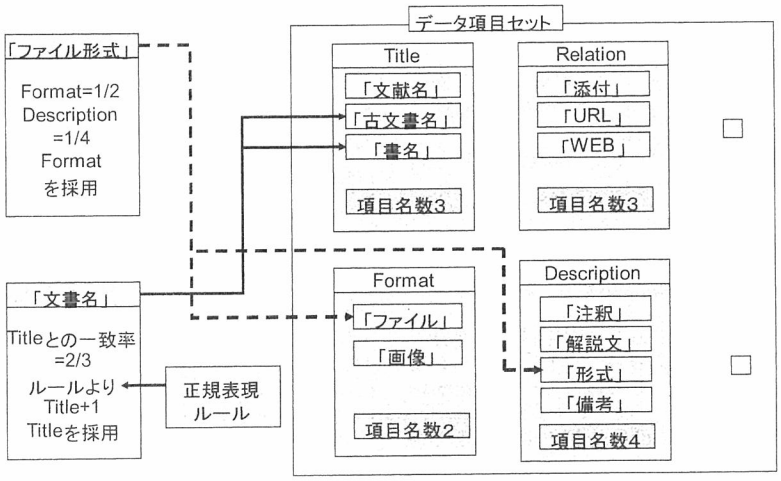

図2　メタデータ要素のマッピング例

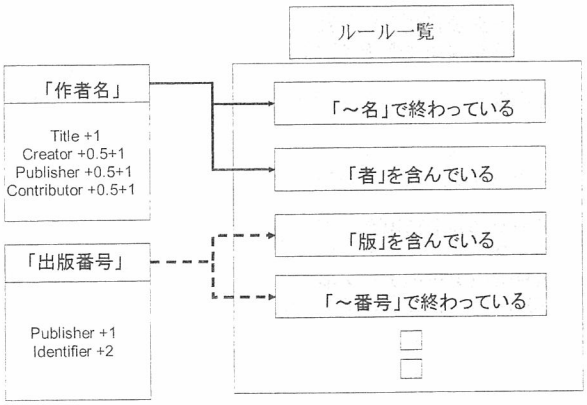

図3　ルールの適用例

4．メタデータの自動マッピング

メタデータの自動マッピングを行うにあたり，まず Dublin Core の基本 15 要素に当てはまるデータ項目名を集め，データ項目セットを作成した．

データ項目名は Web 上に公開されている人文系データベース 30 件から収集した．データ項目名セットに含まれるデータ項目名の数は，最大で Creator の 23 個，最小で Contributor，Format の 3 個であり，平均は約 8 個である．

なお，自動マッピングにおいて，データ項目セットと判別したいデータ項目名との部分一致を取るが，ここでの部分一致とはデータ項目セット中のデータ項目名が判別したいデータ項目名の文字列に含まれている，もしくは判別したいデータ項目名がデータ項目名セット中のデータ項目名の文字列に含まれているときとする．

Dublin Core において Description のメタデータ要素はレコードの説明に当たるデータ項目名に対して付与される．レコードの説明に当たるデータ項目名はデータベースが対象にしているレコードによって様々であり，データベース特有のデータ項目名になることがある．よって，他のデータベースとは類似しないデータベース特有のデータ項目名は Description となる可能性が高い．

以上を踏まえて，自動マッピングの手順は以下の通りである．

- 判別したいデータ項目名を，データ項目セット中のデータ項目名と比較し，部分一致した数を求める．
- データ項目セット中のデータ項目名と部分一致した数を，データ項目セット中のデータ項目名の総数で割り，各項目との一致した割合（以降，メタデータスコア）を求める．
- 判別したいデータ項目名に表 1 のルールを適用し、適合したルールによってメタデータスコアの値に加算する．
- 一番メタデータスコアの値の大きかった項目のメタデータ要素を採用する．ただし，どのデータ項目セット中のデータ項目名とも一致せず，どのルールにも適合しなかった場合，そのデータベース特有のデータ項目名であると判断し Description とする．

図 2 はメタデータ要素のマッピングを図で表したものである．全体の処理の流れとしては，図 3 にあるように「文書名」というデータ項目名の場合，データ項目セット Title 内の「古文書名」，「書名」に部分一致している．データ項目セット中のデータ項目名の総数は 3 なので，「文書名」の Title とのメタデータスコアは 2/3 となる．また「文書名」は「～名」で終わっているので，ルールを適用して，Title に値を 1 追加する．よって「文書名」の Title の値は 1+2/3 となり約 1.66 となる．その他のデータ項目名セットでは適合するものはなく，適応されるルールもないので「書名」は Title 以外のメタデータスコアは 0 になり，一番メタデータスコアの大きかった Title をメタデータ要素に採用する．

「ファイル形式」ではデータ項目セット Format 内の「ファイル」と部分一致している．データ項目名の総数は 2 なので Format のメタデータスコアは 1/2 となる．また Description 中の「形式」とも部分一致し，総数は 4 なので Description のメタデータスコアは 1/4 となる．これ以外に適合するデータ項目名もなく，適用されるルールもないので，その他のメタデータスコアは全て 0 になる．Format と Description の値では Format のメタデータスコアの方が大きくその他のメタデータスコアは全て 0 なので，「ファイル形式」には Format をメタデータ要素に採用する．

図 3 はルールの適用の例を図で表したものである．ルールの適用では，図 2 にあるように「作者名」といったデータ項目名ではルール中の「～名で終わっている」に適合するので，対応するメタデータスコアの値 Title に+1, Creator に+0.5, Publisher に+0.5, Contributor に+0.5 を行う．また「者を含んでいる」にも適合するので，Creator に+1, Publisher に+1, Contributor に+1 を行う．「出版番号」は「版を含んでいる」に適合するので，Publisher に+1 を行う．また「～番号で終わっている」に合致するので Identifier に+2 を行う．

5．提案する横断検索システム

本システムでは，図 4 に示すように，まずユーザに横断的に検索したいデータベースと検索語を指定してもらう．次にその指定されたデータベースからデータ項目名を抜き出す．指定されたデータベースが Z39.50 や SRW/SRU などのプロトコルを採用していればデータ項目名は容易に取得することが出来るが，現状ではそのようなデータベースまだ数少なく，ある程度手動で行う必要も出て来る．続いて抜き出したデータ項目名に対してメタデータの自動マッピングを行い，ユーザが指定した検索語とメタデータにマッピングされたデータ項目名から自動的に検索要求を作成する．最後に各データベースに対して検索を行い，検索結果をユーザに返す．

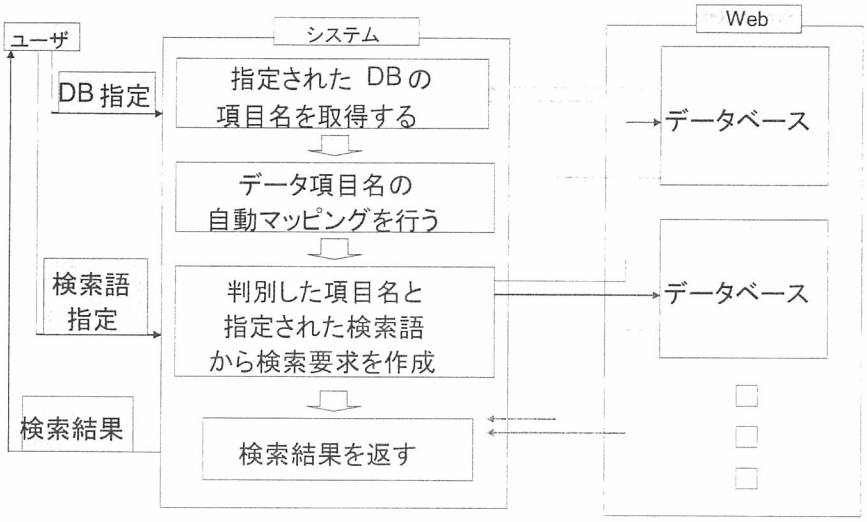

図 4　システム概念図

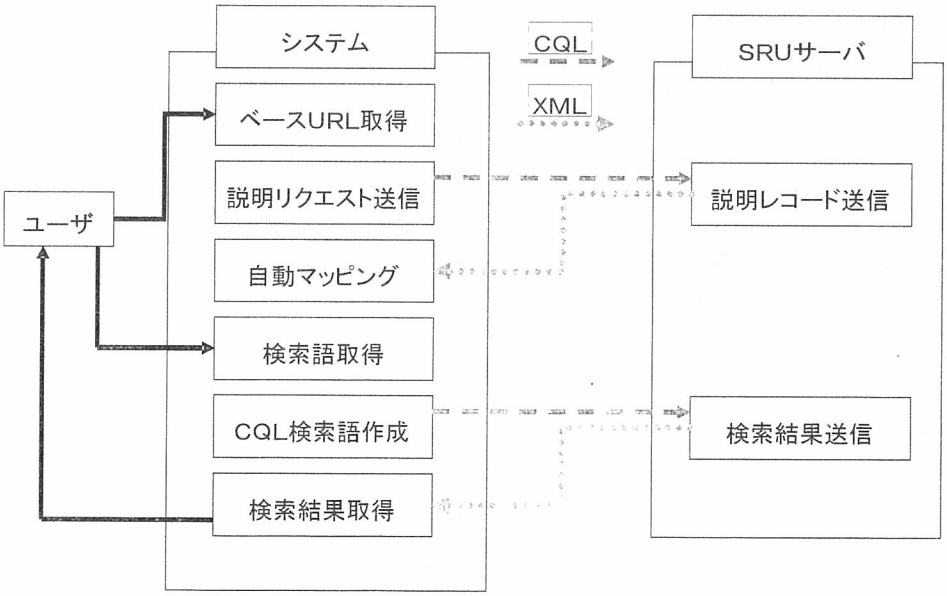

図 5　SRU 横断検索構想図

6. SRW/SRU

SRW（Search/Retrieve Web Service）/SRU (Search/Retrieve via URL) とは，Web 環境で提供・利用されることを前提とした，情報検索のための通信プロトコルである[5]．通信プロトコルに HTTP，検索要求や検索応答に URL や XML を使っており，前身の Z39.50 より Web との親和性が高くなっている．SRU と SRW の違いは、SRU は検索要求に REST フレームワークに基づいた URL を用いる点，SRW は検索要求に SOAP 仕様に基づいた XML を用いる点が異なるだけで他は共通である．URL を打ち込めば使える SRU は、特殊なクライアントがなくても Web ブラウザからも簡単に利用でき，SRW よりも多くのサービスが利用されている．

SRU/SRW では，検索式を CQL(コンテキストクエリー言語;Context Query Language)を使って表現 している．CQL では、検索でよく使われるほとんどの機能(検索項目を指定した検索，論理演算、フレーズ検索、範囲検索、トランケーション，近接演算、結果のソートなど)をサポート している．

SRU/SRW では、3 つのオペレーションに対応する必要がある．

- 検索・返戻オペレーション(searchRetrieve)
  - －検索、返戻（検索結果），ソート済み検索結果の要求・応答のオペレーション
- 説明オペレーション（explain）
  - －サーバ情報の要求・応答のオペレーション
- スキャンオペレーション（scan）
  - －検索語のリストの要求・応答のオペレーション

このうち説明オペレーションを用いればサーバの機能やデフォルト情報などを説明する Explain レコード（XML ドキュメント） を入手することが出来る．この Explain レコードにデータ項目名の一覧が情報として記載されていれば，データ項目名を入手することが出来る．

また SRU では，Explain レコードはベース URL から入手することが可能である．これによってユーザがベース URL の情報を入力するだけで，システム中ではデータベースのデータ項目名を入手することが出来，また URL に検索式を埋め込むことによって容易に検索要求を出すことが出来る．

図 5 に SRU 横断検索の構想図を示す．データ項目名の自動マッピングが出来れば図 5 のように容易に横断検索のシステムが実現できる．

7. 実験・結果

Web 上で公開されている人文系データベース 50 件を無作為に選び，そのデータ項目名に対してメタデータの自動マッピングを行った．50 件の総データ項目名数は 334 である．本実験では，50 件のデータベースに対して，自動マッピング（ルールあり）と自動マッピング（ルールなし）を人手によりマッピングした判別結果と比較した．自動マッピングの判別結果が人手による判別結果と同じなら正解とし，全項目数中の正解数から，正解率を求めた．以下表 2 に自動マッピングの正解率を記し，表 3 にはルールありのマッピングによるメタデータ要素別のマッピング数を記す．表 4 ではマッピング数とマッピング結果のうちの正解数（自動正解数）と手動により求めた正解数（手動正解数）から計算した，再現率・適合率の値を記す．再現率はマッピング結果の内の正解数を手動により求めた正解数で割った値である．適合率はマッピング結果のうちの正解数をマッピング数で割ったものである．表 5 ではメタデータ要素別にマッピングされたデータ項目名の例を幾つか記す．

表 2　自動マッピングの正解率

ルールの有無	正解率
ルールあり	約 81.7%
ルールなし	約 86.8%

表 3　メタデータ別マッピング数

要素名	マッピング数	自動正解数	手動正解数
Title	71	61	65
Creator	46	42	46
Subject	23	22	25
Description	91	67	77
Publisher	21	18	22
Contributor	0	0	0
Date	21	6	8
Type	5	4	5
Format	2	1	1
Identifier	20	19	20
Source	3	3	5
Language	0	0	0
Relation	0	0	1
Coverage	24	22	44
Rights	7	5	7

表4　適合率・再現率

要素名	適合率（%）	再現率（%）
Title	84.7	93.8
Creator	91.3	91.3
Subject	95.7	88.0
Description	73.6	87.0
Publisher	85.7	81.8
Contributor	—	—
Date	28.6	75.0
Type	80.0	80.0
Format	50.0	100.0
Identifier	95.0	95.0
Source	100	60.0
Language	—	—
Relation	—	0.0
Coverage	91.7	50.0
Rights	71.4	71.4

表5　マッピング例

メタデータ要素	正解例	不正解例
Title	誌名，書名	人名，国名
Creator	作者名，著者	
Subject	作品分類，形態	分類コード
Description	注記，備考	所蔵機關
Publisher	出版者，版元	資料出版年月日
Contributor		
Date	日付，月日	出版年，上演年
Type	種別	会議種別コード
Format	形式	袋画像
Identifier	番号，ISBN	
Source	出典，シリーズ	
Language		
Relation		
Coverage	西暦，地名	年記
Rights	所蔵，所蔵機関	

表2から，文字列の部分一致割合のみを用いた場合の正解率は81.7%だが，ルールを追加することにより，約5%の精度の向上が見られた．

表3からCreatorやPublisherは精度が良かったが，Date，CoverageやDescriptionは精度がよくないことが分かる．これはDateとCoverageには概念的に被ってしまう部分があり，時間の概念をカバーするメタデータの要素の検討や，どこにも適合しないときDescriptionにしてしまっていることが課題となっていることがわかる．

表4からDateの適合率が悪く，Coverageの再現率が悪いことがわかる．これは本来Coverageにマッピングされるべきデータ項目名がDateにマッピングされてしまっている数が多いためと思われる．

表5からは「書名」や「著者」，「注記」などのデータ項目名は正しくマッピングされていたが，「所蔵機關」などのデータ項目名は後述のように漢字の表記のゆれに対応できずに正しくマッピングできていなかった．また，正規表現の「～名で終わっている」のルールによって「国名」や「人名」がTitleにマッピングされてしまっている．

8．課題

現状では，単純な文字列の部分一致による判定と正規表現によるルールの適応しか行っていないため，漢字の表記のゆれに対応できていない．文字列の比較方法をさらに検討する必要がある．

またデータ項目セット中のデータ項目名の総数にばらつきがあり，少ないところに一致してしまうと値が大きく出てしまう．そのため正しい判別結果が得られていない．ルールは単純なものしか今回は使っていないが，今後さらに有効なルールを検討する必要がある．

今回は，一つのデータ項目名に対して，一つのメタデータにしかマッピングしていない．だが中には，複数のメタデータに当てはまるデータ項目名があることも考慮しなければならない．例えばDublin CoreではDateとCoverageの二つの項目が時間の概念をカバーしているが，実際にユーザが時間を指定したいときには，その二つに大きな差は見受けられない．一つのデータ項目名に対してDateとCoverage両方ともにマッピングされる可能性があることなども考慮する必要がある．

9．まとめ

現在多数の人文系データベースが公開されている中で，いくつか横断検索の研究がなされている．本研究では，横断検索システムの実現のためにデータ項目名のメタデータへの自動マッピングの手法を提案した．メタデータにはDublin Coreを使用した．

公開されているデータベースでは類似したデータ項目名が使われている．そこで本研究では，様々な

データベースからデータ項目名を集め，また各メタデータの特徴を考慮したルールを作成し,それを用いてメタデータへの自動マッピングを行った．集めてきたデータ項目名とマッピングしたいデータ項目名を比較し，文字列の一致により判別を行うが，表記ゆれや複数項目への対応，新たなルールの検討などが今後の課題である．

参考文献

[1] 山本 泰則，原 正一郎，柴山 守，安達 文夫，合庭 惇，安永 尚志:Dublin Core メタデータと Z39.50 プロトコルにもとづく人文科学系データベースの統合検索に関する実証実験，人文科学とコンピュータシンポジウム論文集，pp.199-205, 2004.

[2] 及川 昭文，藤沢 桜子，洪 政国，山本 啓史:研究支援機能を強化したデータベースシステムの開発，人文科学とコンピュータシンポジウム論文集，pp.213-220, 2007.

[3] 安達 文夫:歴史資料の Dublin Core へのマッピングと統合検索，地域研究コンソーシアム情報資源共有化・地域情報学合同研究会，http://www.cseas.kyotou.ac.jp/jcas/infoshare/Past_seminar.html，(参照 2008-11-16）．

[4] Dublin Core Metadata Initiative. http://dublincore.org/, (参照 2008-09-06).

[5] IRDL Docs http://kaede.nier.go.jp/wiki/?c=index, (参照 2008-09-06).