キーワード検索が可能なXMLデータに対するファセット探索

駒水　孝裕† 　天笠　俊之†,†† 　北川　博之†,††

† 筑波大学大学院システム情報工学研究科
†† 筑波大学計算科学研究センター
〒 305-8573 茨城県つくば市天王台 1-1-1
E-mail: †taka-coma@kde.cs.tsukuba.ac.jp, ††{amagasa,kitagawa}@cs.tsukuba.ac.jp

あらまし　XML データを検索する際には，パス式に基づく問合せとキーワードを用いた問合せが一般的である．これらには，1) データの構造を把握している必要がある，2) 構造的な問合せをキーワードでは記述できない，などの問題がある．これらの問題に対し，我々はこれまでに XML データに対するファセット検索手法を提案してきた．ファセット検索は近年注目される探索的検索手法の一つである．我々は先行研究において，ファセット検索は XML データの探索的な問合せに対し有効であることを示した．しかし，その半面，明示的な問合せ要求に対しては効率的でないことが明らかとなった．そこで，本稿では明示的な問合せに広く用いられてきたキーワード検索をファセット検索に導入することを提案し，被験者実験によりその有効性を示す．

キーワード　XML, ファセット検索，キーワード検索，XML 情報検索

A Keyword Search-enabled Faceted Exploration over XML Data

Takahiro KOMAMIZU†, Toshiyuki AMAGASA†,††, and Hiroyuki KITAGAWA†,††

† Graduate School of Systems and Information Engineering
†† Center for Computational Sciences
University of Tsukuba
1-1-1 Tennodai，Tsukuba，Ibaraki 305-8573，Japan
E-mail: †taka-coma@kde.cs.tsukuba.ac.jp, ††{amagasa,kitagawa}@cs.tsukuba.ac.jp

1. はじめに

XML (Extensible Markup Language) [4] は近年，標準的なデータフォーマットとして広く普及している．そのため，様々なデータが XML 形式で記述されるとともに．多様な XML データに対する問合せ要求が高まっている．XML データに対する問合せとして代表的なものに，1) パス式に基づく問合せ (XPath [2], XQuery [3]) と 2) キーワードを用いた問合せが存在する．1) パスに基づく問合せでは，XML の持つ木構造のルートから所望するノードまでの道筋を記述することでそのノードを検索結果として得る．2) キーワードを用いた問合せ [9], [11], [14] では，キーワードの集合を入力とし，キーワード集合内の単語がマッチする尤もらしい部分木をその結果として返す．

これらの手法には，パス式を記述するために XML データの構造をあらかじめ把握している必要がある，という問題点がある．これに対し，我々は [5], [21] で行われている QCDML に対するファセット検索の研究を基に，先行研究 [18], [19] で XML データに対してファセット検索を適用する手法を提案した．また，我々は先行研究 [20] において [18], [19] で提案した XML データに対するファセット検索手法のプロトタイプシステムを用いた被験者実験を行った．この被験者実験により探索的な問合せ要求に対して我々の手法が有効であることを示した．しかしながら，先行研究の手法ではオブジェクトを特定するような条件が明示された問合せ要求に対しては有効性を欠くということが明らかとなった．

本稿では，XML に対するファセット検索を，明示的な問合せ要求に対しても効率的に検索できる手法を提案する．具体的にはファセット検索の過程でキーワード検索を可能にするキーワード選択演算を導入する．

本稿の構成は次のようになる．まず，第 2 節では基本事項であるファセット検索について説明し，本稿で用いる例示データを紹介する．第 3 節で先行研究の紹介と実験で明らかになった問題点について述べる．続く第 4 節で先行研究における問題に対する改善手法を提案し，第 5 節の被験者実験により本手法の

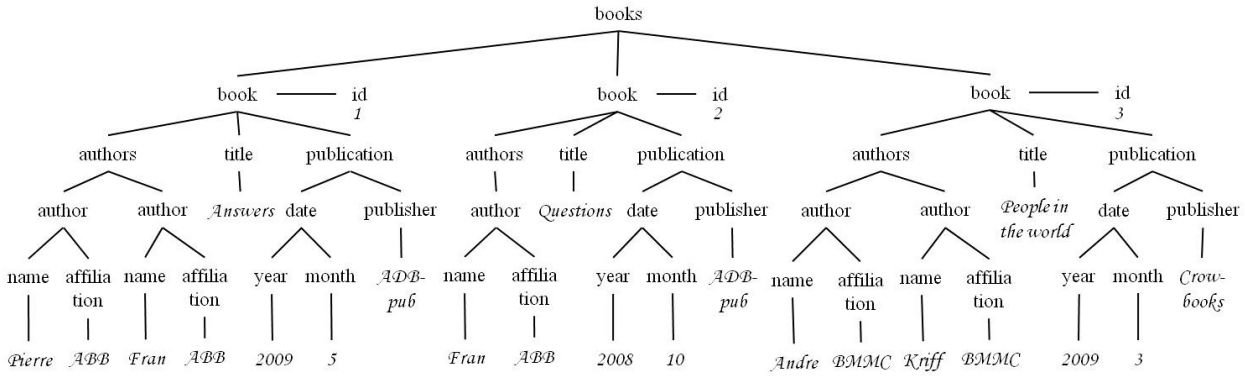

図 1　XML データの例：本のリスト

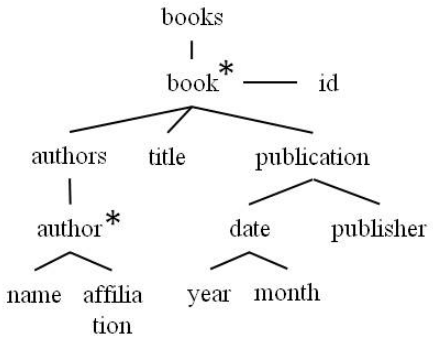

図 2　例 1 のデータの拡張 DataGuide

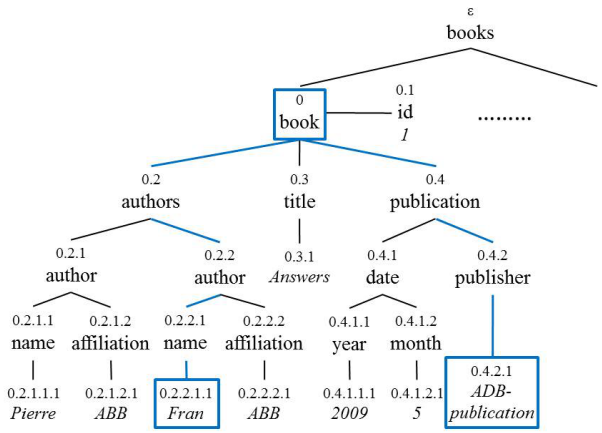

図 3　Dewey 番号と LCA の例

有効性を示す．第 6 節で関連研究について述べ，最後に第 7 節で本稿をまとめる．

2. 準　　備

2.1 ファセット検索

ファセット検索は探索的検索手法の一つで，利用者の検索行動をサポートする手法である．ファセット検索システムのインターフェースは，ファセット (facet) とその値のリストを表示する．ファセットは検索対象のオブジェクト集合を分類するカテゴリである．各値にはオブジェクト集合中に出現する数が付与されており，これによりユーザは現在のオブジェクト集合の全体像を把握することができる．ユーザはファセットの値を選択することで検索対象を絞り込む．逆に，現在得られている結果において目的のデータが見つけられない場合は，すでに選択したファセットを取り除くことで，検索範囲を広げることができる．その結果，選択されたオブジェクトに基づき，選択可能なファセットとそれぞれのファセットが取りうる値が再計算される．この過程を，利用者が望むオブジェクトが検索されるまで繰り返す．

2.2 キーワード検索

本節では XML データに対するキーワード検索について述べる．キーワードの集合を入力として XML データの部分木を検索することを可能にした技術である．XML データに対するキーワード検索は現在も注目されている研究トピックである [9], [11]．

キーワード検索は入力されたキーワード集合にマッチする尤もらしい部分木を返す．この尤もらしい部分木を決める一つの方法として，すべてのキーワードにマッチするノード集合を包含する最少共通祖先である LCA (Lowest Common Ancestor) が用いる方法がある．LCA はルートを持つグラフにおいてあるノード集合が共通で持つ祖先ノードのうちで最もルートから遠い位置にあるノードのことである．XML データにおいて LCA を計算する際には Dewey 番号 [12] が多くの研究で利用されている．Dewey 番号は親の番号に自分の場所の番号を付加する形で表現される．この Dewey 番号を利用して LCA を計算するためには，各 Dewey 番号に共通する最長の接頭辞 (prefix) を見つける．最長の接頭辞が表すノードが LCA となる．

例 1 の一部のデータを例にとる (図 3)．それぞれのノードに付加されている数字が Dewey 番号である．ノード集合 {0.2.2.1.1, 0.4.2.1} の LCA は集合中の Dewey 番号の最長の接頭辞を計算する．この計算により，0 が LCA を表す Dewey 番号とわかり，対応するノードは図中の book ノードである．

LCA を用いて XML データに対してキーワード検索を行う際には，キーワードにマッチするノード集合の LCA をその答えとして返す．従来の研究では，SLCA (Smallest LCA) [16] や VLCA (Valuable LCA) [9]，LCA をもとにした効果的なノードに関する研究 [11] などが提案されている．

2.3 例示データ

本稿では以下のデータを例示データとして本稿での説明に用いる．

［例 1 ］(本のリスト） 図 1 にデータを示した．このデータは本についての情報を取り扱い，著者 (authors)，タイトル (title)，出版 (publication) に関する情報を持つ．

3. 異種 XML データに対するファセット検索

本節では先行研究 [18], [19] で提案した XML データに対するファセット検索手法について紹介し，先行研究 [20] で行った被験者実験とその考察について議論する．

3.1 諸 定 義

まず，構造情報に基づいて定義するクラス，プロパティと XML データに対して定義するオブジェクト，ファセットの定義を与える．本研究では構造情報として，DataGuide [6] に各ノードが親ノードに対する出現頻度に関する濃度 (cardinality) を持つように拡張したものを用いる．濃度の取りうる値としては，空または * であり，それぞれ 1 回の出現と 0 回または複数回の出現を表す．例 1 に対する拡張 DataGuide は図 2 のようになる．ここで，book ノードと author ノードは親ノードに対し複数回出現するため*がついており，その他のノードは一回しか出現しないのでラベルは付いていない．以下でそれぞれの定義について説明する．

3.1.1 クラスとプロパティ

XML データは木構造を持つことから，特定の検索対象を決めることは必ずしも容易ではない．そこで，ノードの濃度に注目して検索対象を定義する．濃度に * を持つということは複数回出現することを意味しており，このようなノードは情報のまとまりを表していると考えられるため，これを検索対象と考えクラスとして定義する．

このようにして与えられるクラスに対してプロパティを決める．クラスに対するプロパティを次のように定義する．テキストノードを直接持つノードはそのテキストの意味を表現していると考えられる．このようなノードは直近のクラスノードの特徴を表していると考えられるため，これをクラスのプロパティとして定義する．

例 1 におけるクラスとプロパティを表 1 に示した．

表 1 例 1 のクラスとプロパティ

クラス	プロパティ
book	title, year, month, publisher
author	name, affiliation

3.1.2 オブジェクトとファセット

上で定義したクラスをルートとする部分木をオブジェクトと定義する．このオブジェクト集合に対してファセット検索を適用するために，オブジェクト集合中に出現するプロパティをファセットと定義する．例 1 におけるファセットとその値のリストを表 2 に示した．

表 2 例 1 におけるファセット

ファセット	値のリスト
id	1, 2, 3
title	Answers, Questions, People in the world
year	2009, 2008
month	5, 10, 3
publisher	ADB-publication, Crow-books
name	Pierre, Fran, Andre, Kriff
affiliation	ABB, BMMC

3.2 演算の定義

上記の概念に基づいてファセット検索を実現するための演算を定義する．検索する際にはこれらの演算を組み合わせることでオブジェクト集合を絞り込む．

3.2.1 選 択 演 算

選択演算はファセットとその値を指定することでオブジェクト集合の絞込みを行う演算である．選択演算には，ファセット名による選択演算とパス指定に基づく選択演算と存在選択演算の 3 種類があり，それぞれ以下に示すような演算である．

ファセット名による選択演算 ファセット名による選択演算は $\sigma^{name}_{\{(facet\_name,value)\}}(O)$ のようにファセット名とその値を指定することで選択演算を行うものである．この演算では同一名を持つ異なるコンテキストで出現するプロパティも同じファセットとみなして検索を行うという特徴を持つ．

パス指定に基づく選択演算 多様な XML データを扱う場合には同一名の異なるノードがファセットに混在することがある．これらのノードはそれぞれの出現するコンテキストにより値の持つ意味が変わる可能性がある．例えば，year というプロパティが published というノード以下に出現する場合は発売年あるいは発行年を表すのに対し，date-of-birth というノード以下に出現する場合は誕生年を表す，というように year の意味が異なる．そこで，あらかじめ抽出したプロパティのパスをファセットを選択する際に選ぶことを可能にした．パス指定に基づく選択演算は $\sigma^{path}_{\{(facet\_path,value)\}}(O)$ のようにファセットのパスと値を指定することで選択演算を行うものである．

存在選択演算 存在選択演算は $\sigma^{exist}_{\{facet\}}(O)$ のように指定したファセットが出現するすべてのオブジェクトを検索する演算である．

3.2.2 クラス選択演算

クラス選択演算はクラスを指定することでそのクラスのオブジェクト集合を返す演算である．クラス選択演算には，クラス名によるクラス選択演算とパス指定に基づくクラス選択演算の 2 種類があり，それぞれ以下に示すような演算である．

クラス名によるクラス選択演算 クラス名によるクラス選択演算は $\phi^{name}_{\{class\_name\}}(O)$ のようにクラス名を指定することでクラス選択演算を行うものである．この演算は同一名のクラスを区別することなく検索を行うという特徴を持つ．

パス指定に基づくクラス選択演算 前述したパス指定による選択演算と同様にクラスを選択する際にもクラスの出現するコンテキストを選択可能にした，パス指定に基づくクラス選択演算

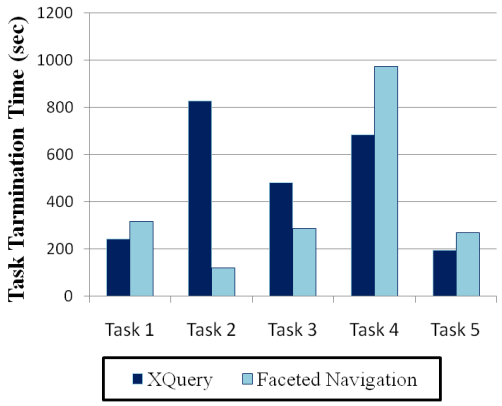

図 4　先行研究における実験 (タスク遂行時間)

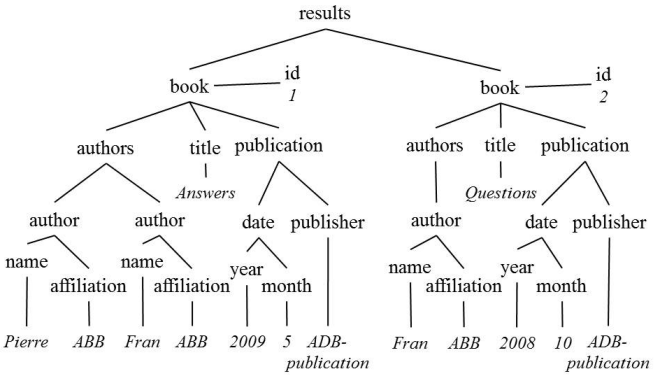

図 5　キーワード選択演算の例: $\rho_{\{"Fran","ADB-publication"\}}(O)$

は $\phi^{path}_{\{class\_path\}}(O)$ のようにクラスのパスで指定することでクラス選択演算を行うものである．

3.3　被験者実験

これまでの定義に基づいてプロトタイプシステムを構築し評価実験を行った [20]．評価実験では 10 名のボランティアによる被験者実験を行った．実験においてタスクの設計として探索的な問合せタスクを 2 個 (Task2, Task3) と明示的な問合せタスクを 3 個 (Task1, Task4, Task5) 用意し，それぞれのタスクの遂行時間の測定とタスク遂行の達成度に関するアンケートにて XQuery との比較評価を行った．本稿での提案手法の実験でも同様のタスクを用いているため，タスクの詳細については第 5. 節で述べる．[20] での実験の結果は，図 4 のようになった．ここでは，タスク遂行時間のみを表示する．

この実験から，XML データに対するファセット検索が探索的な問合せタスクに対して有効であることを示した．その反面，明示的な問合せタスクに対しては XQuery の方が有効であるという結果となった．この原因としては，明示的な問合せにて指定する値をファセット検索のインターフェースから探し出す際に目視にて探さなければならず，その過程に時間がかかっているためである．

4.　提 案 手 法

先行研究 [20] の実験から，明示的な検索要求に対応することによって，ファセット検索インターフェースの利便性が向上することが期待される．特に．この実験の考察から任意の単語をファセットから検索する機構を導入することが有効であると考えられる．任意の単語を検索する方法としては，従来の情報検索で最もよく用いられるキーワード検索が有効であると考えられるため，本稿ではファセット検索にキーワード検索を行う機構をファセット検索に導入することを提案する．

4.1　キーワード選択演算

本研究ではキーワード検索機構をひとつの演算として定義することで，ファセット検索の過程の一環としてキーワード検索を可能にする．キーワード選択演算をキーワード集合を入力としてキーワードにマッチするノードの LCA (Lowest Common Ancestor) を含み，かつ LCA から直近のオブジェクトの集合を返すような演算であると定義する (定義 1)．ここで LCA とはグラフ理論で用いられる概念でルート付きグラフにおいて，ノード $v_i ... v_n$ に共通する祖先のうち最もルートから遠いもののことを指す．

[定義 1](キーワード選択演算)　KW をサイズ n のキーワードの集合とし，各キーワード $keyword_i \in KW$ にマッチするノードをそれぞれ v_i とする．このときオブジェクト集合 O に対するキーワード選択演算 $\psi_{KW}(O)$ を以下のように定義する．$\mathrm{LCA}(v_1 ... v_n)$ はノード $v_1 ... v_n$ の LCA である．また，$a \preceq b$ は a が b の祖先ノードまたは a と b が一致することを表す．

$$\psi_{KW}(O) = \{o \mid o \in O \wedge o \preceq LCA(v_i, ..., v_n)\}$$

この定義により，キーワード検索を任意のオブジェクトに対し行うことができるようになった．これにより，他の演算と同様にキーワード選択演算を利用できるため，任意のタイミングでのキーワードによる絞り込みが可能になった．キーワード選択演算の例を図 5 に示した．この例では入力キーワードは”Fran”と”ADB-publication”である．これらにマッチするノードの LCA は book ノードであり，book はクラスであるのでそれぞれの LCA となった book オブジェクトが 2 つが図のように検索結果として得られる．

4.2　検索プロセス

本節では検索を行う際の処理について説明する．まず，オブジェクトを検索する際のアルゴリズムをアルゴリズム 1 に示した．検索プロセスは入力を現在の検索結果オブジェクト集合 O と検索演算の集合 $Cond$ とし，すべての条件にマッチするオブジェクト集合 O' を出力する．$Cond$ は選択演算の集合 Σ，クラス選択演算の集合 Φ，キーワード選択演算の集合 Ψ からなる．関数 $selection$ はオブジェクト集合 O におけるそれぞれの選択演算 $s \in \Sigma$ により得られるオブジェクト集合の積集合を返す関数である．関数 $c\_selection$ はオブジェクト集合 O におけるそれぞれのクラス選択演算 $cs \in \Phi$ により得られるオブジェクト集合の積集合を返す関数である．また，関数 $k\_selection$ はオブジェクト集合 O におけるキーワードの集合 $T \in \{p.term \mid p \in \Psi\}$ に対するキーワード選択演算により得られるオブジェクトの集合を返す関数である．

次に，検索されたオブジェクト集合 O' に対してファセット

Algorithm 1 検索アルゴリズム

INPUT: O is a set of current result objects, $Cond := \{\Sigma, \Phi, \Psi\}$
$O' \leftarrow selection(\Sigma, O) \cap c\_selection(\Phi, O) \cap k\_selection(\Psi, O)$
return O'

function $selection(\Sigma, O)$
$O_{selection} \leftarrow O$
for each selection $s \in \Sigma$ **do**
  if s is *name_based selection* **then**
    $O_{selection} \leftarrow \sigma^{name}_{\{(s.name, s.value)\}}(O_{selection})$
  else if s is *path_based selection* **then**
    $O_{selection} \leftarrow \sigma^{path}_{\{(s.path, s.value)\}}(O_{selection})$
  else
    //existential selection
    $O_{selection} \leftarrow \sigma^{exist}_{\{s.name\}}(O_{selection})$
  end if
end for
return $O_{selection}$

function $c\_selection(\Phi, O)$
$O_{c\_selection} \leftarrow O$
for each class selection $cs \in \Phi$ **do**
  if cs is *name_based class selection* **then**
    $O_{c\_selection} \leftarrow \phi^{name}_{cs.name}(O_{c\_selection})$
  else
    //path based class selection
    $O_{c\_selection} \leftarrow \phi^{name}_{cs.path}(O_{c\_selection})$
  end if
end for
return $O_{c\_selection}$

function $k\_selection(\Psi, O)$
$T \leftarrow \{p.term \mid p \in \Psi\}$
return $\psi_T(O)$

の持つ値のリストを計算する (アルゴリズム 2)．各ファセット f に対して関数 $countForEachValue$ を用いて f の持つ値をグルーピングし ($groupingValue$ 関数)，それぞれの値がオブジェクト集合 O' 中に出現する回数を付加し変数 V に格納する．ファセットと V の要素を出現回数の降順にソートしたものペアにして $FacetList$ に追加する．

4.3 プロトタイプシステム

本研究では先行研究 [20] のシステムに対してキーワード選択演算を導入する．図 6 に本システムの概要を示す．以下で各モジュールについて説明する．

Retrievau Interface　ここではファセット検索に必要な情報をユーザに提示し，ユーザの入力を受け付けるなど対話的な処理を行う．ユーザの入力を Retrieval module に受け渡し，その結果を Display module から受け取りユーザに表示，再度問合せの入力を待つ，という動作を繰り返す．

Retrieval module　ここでは Retrieval Interface からの問合せを基に各データベースへの問合せを生成する．データベースへの問合せの結果を Display module に受け渡す．

Algorithm 2 ファセットの値の計算アルゴリズム

INPUT: a set of retrieved objects O'
$FacetList \leftarrow \{\}$
for each facet $f \in F$ **do**
  $V \leftarrow countForEachValue(f, O')$
  $FacetList \leftarrow FacetList \cup (f, sort(V))$
end for
return $FacetList$

function $countForEachValue(f, O')$
$valueCount \leftarrow \{\}$
$valueList \leftarrow groupingValue(f, O')$
for each value $v \in valueList$ **do**
  $valueCount \leftarrow valueCount \cup (v, count(v))$
end for
return $valueCount$

Display module　ここでは Retrieval module から検索結果を受け取り，ユーザに表示する形式に成形し Retrieval Interface に受け渡す．

Facet Database　ここではファセット検索に必要な情報を保管しておくためのデータベースである．Retrieval module からの問合せを処理する．Facet Database に保管されるデータはクラス，プロパティ，オブジェクト，ファセット，オブジェクトの子孫ノードに出現するキーワード，でありそれぞれのスキーマは表 3 のようになる．ここで，ファセットのテーブルはそれぞれのファセットについてその名前のテーブルを作成する．

表 3　Facet Database の各テーブルのスキーマ

テーブル名	スキーマ
Class	(class-id, class-name, path)
Property	(class-id, property-name, path)
Object	(class-id, object-id, path)
facet-name	(value, class-id, object-id)
Keyword	(term, class-id, object-id)

XML Database　XML Database は XML データを管理するためのデータベースである．Retrieval module から受け取った XQuery による問合せを処理し，オブジェクト集合を結果として返す．

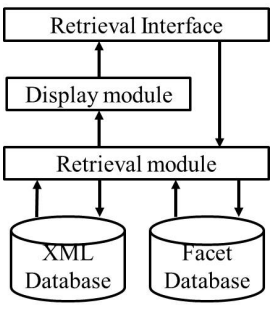

図 6　システムの概要

表 4　探索的な問合せタスク

Task2	あなたは研究室のグループで研究者についての調査を行うことになりました．内容としてはある研究者の最も業績を上げていた時期とその時のトピックを調べるというものです．そこで，検索システムを使い任意の研究者を自身で選び，その研究者が最も活躍した年を一つ 挙げ，その年の中でトピックを表すような論文を 3 本探してください．
Task3	1997 年に学位論文 (mastersthesis, phdthesis のどちらか一方) を執筆した人の国際会議で発表された論文を検索システムを利用して，探してください．

表 5　明示的な問合せタスク

Task1	あなたは先日，研究室の同僚にデータベースを研究している Michael J. Franklin 教授を知っているか尋ねられました．あなたは名前こそ知っていましたが具体的なトピックや論文についてはあまり知りません．そこで，検索システムを用いて Michael J. Franklin 教授が行っている研究トピックを三つ探してください．
Task4	あなたは今学期「システム言語論 (仮称) 」を受講していることを想定して下さい．次回の授業から OCaml を利用することになりました．そこで宿題として ”Using, Understanding, and Unraveling the OCaml Language. From Practice to Theory and Vice Versa.” という論文を読むことになりました．この論文を検索システムを用いて検索してください．
Task5	あなたの研究チームは今度のセミナーで研究室のこれまでの年毎の代表的な論文を紹介することになりました．あなたの担当は 2002 年分です．また，事前の下調べで 2002 年は，IDEAS という学会に投稿していることが分かっています．しかし，論文のタイトルと北川博之教授以外の著者の情報がわかりません．この検索システムを用いて必要な情報を収集してください．

5. 被験者実験

本節では，提案手法の有効性を検証するための被験者実験について紹介する．探索的検索を評価する上で評価のためのタスクの設計は重要である．本稿では先行研究 [20] と同様に [8] の考え方を基にタスクの設計を行った. [8] では探索的検索を評価する際に考慮すべき点として以下の項目を挙げている．

- 不確定であること (uncertainty)
- 不明確であること (ambiguity)
- 発見的であること (discovery)
- 不慣れなド メインであること (unfamiliarity)
- 受け入れやすい状況を想定すること (situation)

我々は実験に伴い，上記特性を満たすタスクを探索的問合せタスクとし，上記の特徴のうち，不確定性と不明確であること，発見的であることの 3 点を除いたものを明示的問合せタスクとし，合計 5 つのタスクを設計した (表 4，5)．うち二つ (Task2, Task3）は探索的問合せタスク，それ以外のタスク (Task1, Task4, Task5) については明示的問合せタスクとした．

Task4 は Task2 に比べて明示的な問合せである．なぜなら，Task2 が不特定の著者の論文 3 本という指定なのに対し，Task4 ではタイトルを指定された 1 本の論文を探すタスクなためである．

実験での評価項目としてはタスク遂行までにかかった時間 (タスク遂行時間), タスク遂行の達成度に関する 2 項目のアンケート (タスク遂行に対する自信度，タスク遂行の容易さ) である．比較対象としては，構造的検索を行うことができる XQuery と従来研究の XML データに対するファセット検索を用いる．実験には我々の研究室のボランティア 10 名の協力のもとで行った．この 10 名はいずれもデータ工学の分野について学んでおり，10 名のうち XML に関しての知識が十分な者は 1 人だったため，事前に 1 週間，XML および XQuery について勉強を行った．実験結果を図 7，8，9 にそれぞれ，タスク遂行時間，タスク遂行に対する自信度，タスク遂行の容易さを示した．各図の左の濃い青の棒グラフは XQuery を，真ん中の青い棒グラフは従来のファセット検索を右の薄い青の棒グラフは本稿で提案しているキーワード検索機構を導入したファセット検索を示す．

タスク遂行時間 (図 7) を見ると，本稿で提案したキーワード検索機構を導入したファセット検索が他に比べてタスク遂行までにかかる時間が短くなっていることがわかる．この要因としては 2 つ考えられる，1) 明示的な問合せ (Task1, Task4, Task5) に対応できる手法になったこと，2) 探したいファセットを目視で探す必要がなくなったこと．1) は先行研究のシステムの弱点を克服したことを示しており，本手法が有効であることを示している．また，2) は先行研究ですでに有効であったファセット検索をキーワード検索を導入することで効率の向上を図れたことを示している．加えて，タスク遂行における達成度に関するアンケート (図 8，9) では自信度，容易さともに向上しており，キーワード選択演算がユーザの検索効率を向上していることが観察できる．

6. 関 連 研 究

本節では，本研究に関連する研究を紹介する．

ファセット検索を用いた検索に関する研究 [1], [7], [15]　Oren らは [1] においてファセット検索を RDF データに適用しており，本稿とは扱うデータが異なる．また，Oren らはファセット検索におけるファセットの順番の重要性について述べ，ファセットの順序付けに対する指標を提案し，その有用性を検証した．ファセットの順序付けに関しては先行研究である [17] で [1] の手法を拡張したものを用いていたが，計算にかかるコストが高く速度面に問題があると結論付けていた．そのため，ファセットの順序付け手法関しては別の議論が必要であると考えられる．

Koren らは [7] にて，ペタバイト級の大規模ファイルシステムの検索にファセット検索を用いることを提案していた．近年ペタバイト級のファイルストレージを用いることが増つつある

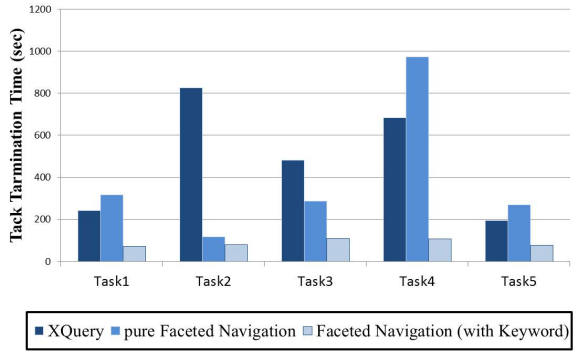

図 7　タスク遂行時間

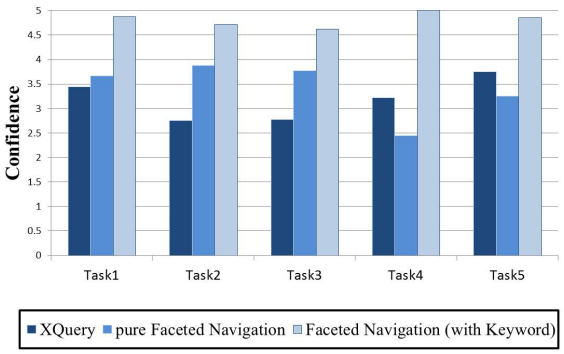

図 8　タスク遂行に対する自信度

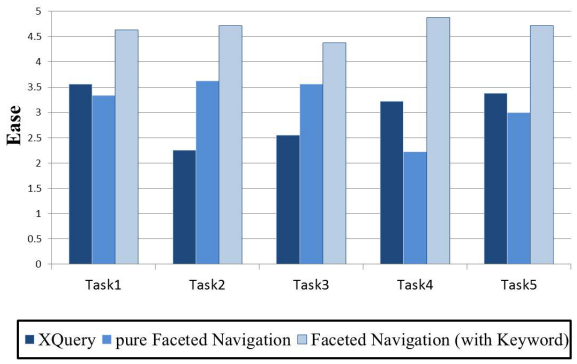

図 9　タスク遂行の簡易さ

ことに着目し，[7] ではファイルシステムで検索する際に利用者がはっきりとした意図で適切な問合せを表現することはできないと述べている．この点への改善策としてファセット検索を適用していた．このことから，大規模データの検索に対する手法としてファセット検索が有用であることがうかがえる．ファイルシステムにおける検索対象はすべてのファイルであるが，本稿では XML データを対象にしているために検索対象が異種のものになるという点で扱うデータが異なっている．

[15] では Tzitzikas らによって *FleXplorer* 提案された．構造データと非構造データに対するファセット検索のミドルウェアとしてのフレームワークが提案されており，検索の高速化を目標としている．またファセットの階層構造を扱うための手法として階層構造を記憶するためのテーブルを用いている．本研究とは階層構造を考慮するという点は共通であるが，その手法としてテーブルを用いるという点で本研究とは異なる．また，この研究ではファセットとキーの抽出については触れられておらず，この点に関しても本研究とは異なる．

XML データに対するインタラクティブな検索に関する研究 [10] Li らは [10] にて XML データに対するインタラクティブなキーワード検索を提案した．Li らはキーワード推薦の技術を XML データ検索に応用し，入力したキーワードのコンテキスト入力を支援する XQSuggest というシステムを構築した．[10] は XML データに対するインタラクティブな検索に関する数少ない研究の一つである．本研究ではファセット検索とキーワード入力支援という点で異なる．

7.　まとめと今後の課題

本稿では，先行研究の手法に対し，キーワード検索機構を導入する手法を提案した．キーワード検索を演算として扱うことでファセット検索の過程にキーワード検索を行うことを可能にした．また，本手法の有効性を被験者実験により確認した．今後の課題としては，パフォーマンスの問題に着手する予定である．この問題はタスクの遂行時間にシステムのレスポンス速度が関わっている問題で，ファセット検索全般に見られる問題である．主要なアプローチとしては，ファセットの取捨選択が行われてきている [13] が，ユーザの欲するようなファセットを必ずしも表示できるとは限らないため，別な方法の検討が必要であると我々は考える．

謝辞　本研究の一部は科学研究費補助金特定領域研究 (#21013004) による．ここに記して謝意を示す．

文　　献

[1] Resource Description Framework. http://www.w3.org/RDF/.
[2] XML Path Language. http://www.w3.org/TR/xpath/.
[3] XML Query Language. http://www.w3.org/TR/xquery/.
[4] XML1.0. http://www.w3.org/TR/REC-xml/.
[5] Toshiyuki Amagasa, Noriyoshi Ishii, Tomoteru Yoshie, Osamu Tatebe, Mitsuhisa Sato, and Hiroyuki Kitagawa. A Faceted-Navigation System for QCDml Ensemble XML Data. In Fatos Xhafa, Leonard Barolli, Hiroaki Nishino, and Markus Aleksy, editors, *3PGCIC*, pp. 132–139. IEEE Computer Society, 2010.
[6] Roy Goldman and Jennifer Widom. DataGuides: Enabling Query Formulation and Optimization in Semistructured Databases. In Matthias Jarke, Michael J. Carey, Klaus R. Dittrich, Frederick H. Lochovsky, Pericles Loucopoulos, and Manfred A. Jeusfeld, editors, *VLDB*, pp. 436–445. Morgan Kaufmann, 1997.
[7] Jonathan Koren, Andrew Leung, Yi Zhang, Carlos Maltzahn, Sasha Ames, and Ethan L. Miller. Searching and navigating petabyte-scale file systems based on facets. In Garth A. Gibson, editor, *PDSW*, pp. 21–25. ACM Press, 2007.
[8] Bill Kules, Robert Capra, Matthew Banta, and Tito Sierra. What do exploratory searchers look at in a faceted search interface? In Fred Heath, Mary Lynn Rice-Lively, and Richard Furuta, editors, *JCDL*, pp. 313–322. ACM, 2009.
[9] Guoliang Li, Jianhua Feng, Jianyong Wang, and Lizhu Zhou. Effective Keyword Search for Valuable LCAs over XML Documents. In Mário J. Silva, Alberto H. F. Laender, Ricardo A. Baeza-Yates, Deborah L. McGuinness, Bjørn

Olstad, Øystein Haug Olsen, and André O. Falcão, editors, *CIKM*, pp. 31–40. ACM, 2007.

[10] Jiang Li and Junhu Wang. XQSuggest: An Interactive XML Keyword Search System. In Sourav S. Bhowmick, Josef Küng, and Roland Wagner, editors, *DEXA*, Vol. 5690 of *Lecture Notes in Computer Science*, pp. 340–347. Springer, 2009.

[11] Ziyang Liu and Yi Chen. Identifying meaningful return information for XML keyword search. In Chee Yong Chan, Beng Chin Ooi, and Aoying Zhou, editors, *SIGMOD Conference*, pp. 329–340. ACM, 2007.

[12] Jiaheng Lu, Tok Wang Ling, Chee Yong Chan, and Ting Chen. From Region Encoding To Extended Dewey: On Efficient Processing of XML Twig Pattern Matching. In Klemens Böhm, Christian S. Jensen, Laura M. Haas, Martin L. Kersten, Per-Åke Larson, and Beng Chin Ooi, editors, *VLDB*, pp. 193–204. ACM, 2005.

[13] Senjuti Basu Roy, Haidong Wang, Gautam Das, Ullas Nambiar, and Mukesh K. Mohania. Minimum-effort driven dynamic faceted search in structured databases. In James G. Shanahan, Sihem Amer-Yahia, Ioana Manolescu, Yi Zhang, David A. Evans, Aleksander Kolcz, Key-Sun Choi, and Abdur Chowdhury, editors, *CIKM*, pp. 13–22. ACM, 2008.

[14] Arash Termehchy and Marianne Winslett. Effective, design-independent XML keyword search. In David Wai-Lok Cheung, Il-Yeol Song, Wesley W. Chu, Xiaohua Hu, and Jimmy J. Lin, editors, *CIKM*, pp. 107–116. ACM, 2009.

[15] Yannis Tzitzikas, Nikos Armenatzoglou, and Panagiotis Papadakos. FleXplorer: A Framework for Providing Faceted and Dynamic Taxonomy-Based Information Exploration. In *DEXA Workshops*, pp. 392–396. IEEE Computer Society, 2008.

[16] Yu Xu and Yannis Papakonstantinou. Efficient Keyword Search for Smallest LCAs in XML Databases. In Fatma Özcan, editor, *SIGMOD Conference*, pp. 537–538. ACM, 2005.

[17] 駒水孝裕, 天笠俊之, 北川博之. XML データに対するファセットナビゲーションのためのフレームワーク FoX の提案. 第 1 回データ工学と情報マネジメントに関するフォーラム (DEIM), pp. B7–6, 2009.

[18] 駒水孝裕, 天笠俊之, 北川博之. 異種 XML データに対するファセット検索手法の提案. 情報処理学会研究報告「デジタルドキュメント (DD)」, Vol. 2009, No. 73, pp. 1–8, 2009.

[19] 駒水孝裕, 天笠俊之, 北川博之. 異種 XML データに対するファセット検索における多様な検索. 第 2 回データ工学と情報マネジメントに関するフォーラム (DEIM), pp. C7–5, 2010.

[20] 駒水孝裕, 天笠俊之, 北川博之. XML データに対するファセット検索のユーザビリティ評価. 情報処理学会 第 73 回全国大会, pp. 4N–3, 2011.

[21] 天笠俊之, 石井理修, 吉江友照, 建部修見, 佐藤三久. XML データを対象としたファセット検索インターフェースの生成. 情報処理学会研究報告「デジタルドキュメント (DD)」, Vol. 2008, No. 53, pp. 7–13, 2008.