2008－DD－66 (2)
2008／6／6

XMLデータを対象としたファセット検索インタフェースの生成

天笠俊之† 石井理修† 吉江友照†
建部修見† 佐藤三久†

大規模なデータ群に対して，探索的な検索手段を提供する利用者インタフェースとして，ファセット検索がある．通常，ファセット検索インタフェースは，データの性質などをよく調べた上で人手で作成されるが，XMLデータの場合，データの構造が複雑であったり，考慮すべき属性の組合せが膨大になるといった問題がある．本稿では，XMLデータ，特に科学データを記述したXMLデータ群に対してファセット検索インタフェースを生成する手法について論じる．

Semi-Automated Generation of Faceted Navigation Interfaces for XML Data

TOSHIYUKI AMAGASA,† NORIYOSHI ISHII,† TOMOTERU YOSHIE,†
OSAMU TATEBE† and MITSUHISA SATO†

A faceted search navigation allows a user to search for his/her desired information in an exploratory way. In general, when constructing a faceted navigation interface, we have to conduct careful studies on the data to be queried in order to obtain a better interface design. However, doing it on XML data incurs special considerations due to the complexity of XML data. In this paper we discuss a semi-automated generation of faceted navigation interfaces on XML data, in particular, scientific XML data.

1. はじめに

計算機の高性能化と大容量化，高速な広域ネットワークの整備により，世界中でデータやプログラム，あるいは計算機資源の交換や交流が可能となった．このような計算機技術の発展は，多くの分野に多大な影響を及ぼしている．このような，大量のデータを背景に構築される応用は，しばしばデータ集約型 (data intensive) と呼ばれ，近年注目されている[1]．

データ集約型の応用においては，膨大なデータが存在するため，これから行なうタスクに必要なデータを，効率よく取得できることが，効率や利便性の面から重要である．これに対し，キーワード等の検索条件を与えることによって所望のデータを絞り込む従来型の検索では，

- 厳密な条件を指定するには，あらかじめデータの詳細を知っておく必要がある（逆に言うと，データの詳細を知らない限り，条件を書くことができない）
- 膨大なデータ群に含まれる全てのデータの詳細を知ることは困難である
- 所望のデータを厳密に表現するような条件を示すことが困難である，あるいは不可能であることが少なくない
- 欲しいデータが明確に定まっていない

といった場合に不都合が生じる．このため，個別の検索を単に繰り返すだけではなく，システムによって情報探索の試行錯誤の過程をサポートすることが重要である．

他方，XML (Extensible Markup Language)[2] は，データフォーマットの事実上の標準として，構造化文書，各種データファイル，ネットワーク上のプロトコルなど，さまざまな応用で用いられている．このように，多くの情報がXML形式で生成され，ネットワークを通じて流通している現在では，XMLで形式の情報源から所望のデータを検索することは，ますます重要になりつつある．

XMLデータから情報を抽出する手段としては，

† 筑波大学 計算科学研究センター
Center for Computational Sciences, University of Tsukuba

図 1　ファセット検索の例 (Flamenco Search).

XPath, XQuery, XSLT などを使った検索，あるいは情報検索技術を応用した XML データのキーワード検索の二つの方法が主流である．しかしながら，XML データの増加に伴い，上で述べたような発見的な検索手段が必要になると思われる．そこで，本稿では，検索，キーワード検索に続く第 3 の手段として探索的検索 (exploratory search)[3] の一つである，ファセット検索 (faceted navigation) の適用を検討する．

ファセット検索は，検索対象オブジェクトの集合を効率よく探索するための手法である．オブジェクトは，あらかじめファセットと呼ばれるいくつかの独立したカテゴリ毎に分類されている．各カテゴリ（ファセット）において，オブジェクトは着目する属性の値毎にグルーピングされており，その値がリスト表示されている．利用者はファセットに含まれる具体的な値を選択することで，オブジェクトの絞り込みを行い，探索を行う．

XML データに対してファセット検索を適用する場合，以下のような問題点が考えられる．

- XML は半構造データであり，構造化されている部分と構造化されていない部分が混在する場合がある．このようなデータに対して，検索対象オブジェクトとファセットを定義する必要がある．
- ファセットの取る値が，単純値だけではなく，（構造を持った）部分 XML データである場合がある．
- ファセットとして，要素，属性が持つ値ではなく，要素名や属性名など，XML の構造に関する情報を許す必要がある．

本稿では，XML データとして，特に科学データを記述した XML データ群を扱う．物理科学，物質科学，生命科学，環境科学の諸分野においては，大規模シミュレーションや大規模データ解析等を中心とするデータ集約的なアプローチで科学的な発見を行う計算科学が注目されており，そこでは，実験・観測データの標準フォーマットやメタデータフォーマット，あるいは特定分野における統制語彙の表現手段として XML が広く用いられている．具体的には，素粒子分野における ILDG (International Lattice Data Grid)☆ のアンサンブル XML データを対象に，ファセット検索インタフェースの生成を検討する．

本稿の構成は以下のとおりである．第 2 章では，基本的事項としてファセット検索，本研究で用いる XML データである QCDml を説明する．第 3 章で，XML データからのファセット抽出について議論し，第 4 章でシステムの概要を述べる．第 5 章で関連研究を紹介し，第 6 章でまとめと今後の課題について触れる．

2. 基本的事項

2.1 ファセット検索

ある情報群から情報を抽出する場合，キーワード検索等の従来型の検索手法を適用するには，情報群に対する事前の知識が必要である．これに対し，探索的検索手法[3] では，事前の知識なしに効率的なブラウジングが可能になる．ファセット検索[4] は，探索的検索手法の一つである．

ファセット検索において，検索対象となるオブジェクト群は，予めファセットと呼ばれるいくつ

☆ http://www.lqcd.org/ildg/

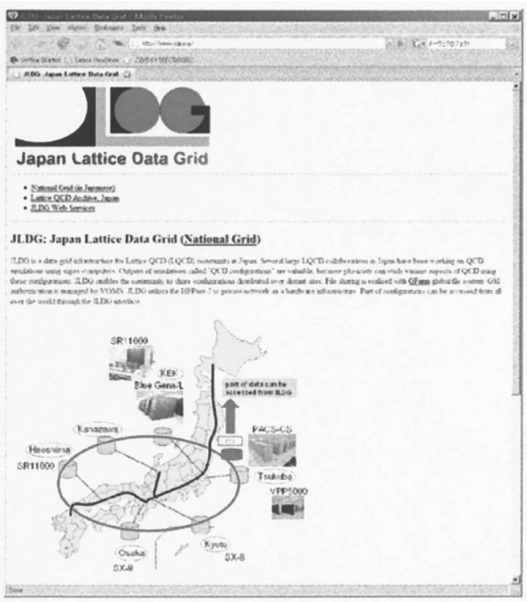


図 2　JLDG (Japan Lattice Data Grid).

かの直交した概念毎に分類されている．ファセットは，検索対象オブジェクトの重要な側面を表している．各ファセットからは，それを代表する値が抽出されており，利用者はそれを選択することで，対象オブジェクトを絞り込んで行く．

図 1 に，The Flamenco Search Interface Project☆によるファセット検索の例を示す．これは，ノーベル賞受賞者をファセット検索で探索している様子である．画面左には，性別，国名，所属，受賞年などのファセットが表示されている．各ファセットには，それが取りうる値の候補と，その値を選択したときに選択されるオブジェクトの個数が示されている．例えば，国名であれば「Argentina (5) / Australia (6) / Austria (12)」といった具合である．右側には，選択されているオブジェクトの一覧が表示される．

ファセット検索の主な利点は次の通りである．

- 各ファセットは直交しており，任意の組み合わせを利用者が指定できる．これは極めて強力であり，10,000 オブジェクトの分類に対して，10 項目から成る四つのファセットがあれば十分対応ができるとされている．
- 絞り込みに用いるファセットの順序は利用者がコントロール可能である．情報の分類・探索の手段として，しばしば階層型の分類手法が用いられるが，これらの手法では階層構造が事前に決められており，それに従った探索しかできないのに比べると自由度が高い．これにより，一つではなく数多くの方向（ファセットの組合せ）から所望の情報にアクセスすることができる．

2.2 QCDml

本稿では，XML データとして ILDG (International Lattice Data Grid) ☆☆ で設計された，格子 QCD 配位データのメタデータフォーマットである QCDml☆☆☆を用いる．ILDG とは，計算素粒子物理学分野における格子 QCD 計算の計算結果である配位データを国際的に共有するためのデータグリッドである．筑波大学計算科学研究センターは，日本の拠点として ILDG に参加するとともに，QCD 配位データのデータベースである LQA を提供している（図 2）．

格子 QCD の配位データはバイナリ形式で表現されるが，それがどのような内容であるかを記述するためのメタデータの標準が必要である．QCDml は，その目的のためにまとめられた仕様であり，どのような物理シミュレーションなのかを記述するアンサンブル XML と，どのようなパラメータなのかを記述するコンフィギュレーション XML の 2 種類から構成される．図 3 にアンサンブル XML の例を示す．

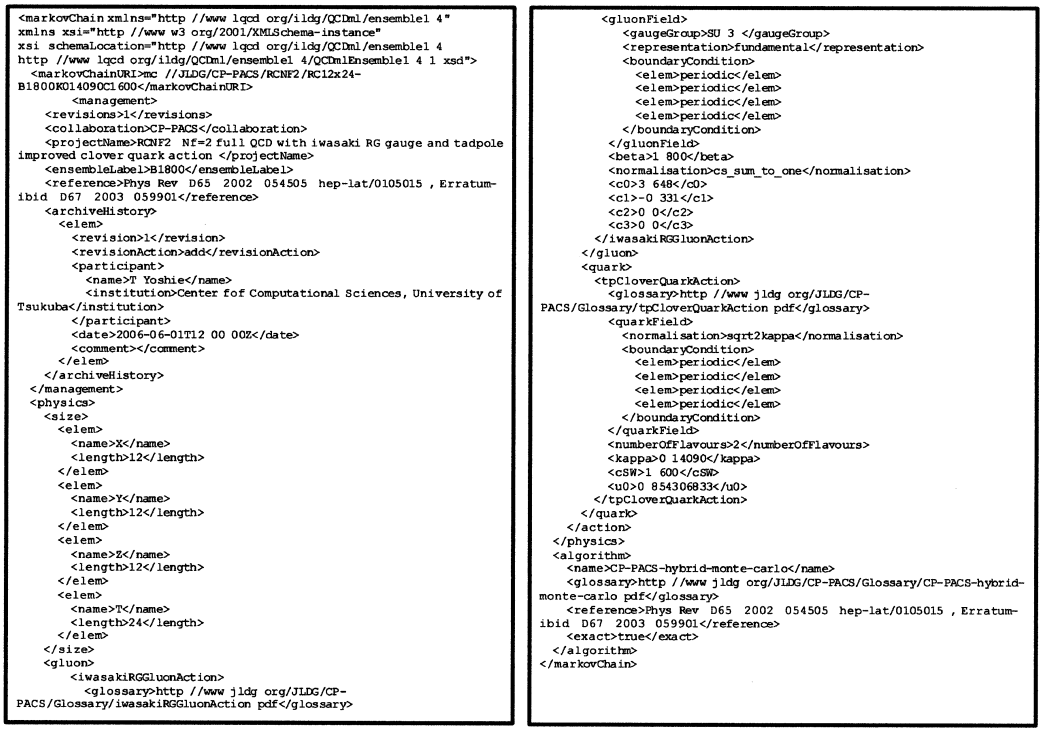
アンサンブルとコンフィギュレーションの間の対応関係は `markovChainURI` と呼ばれる識別子によって記述される．また，バイナリには LFN (Logical File Name) と呼ばれる識別子が割り当てられており，コンフィギュレーションにその値が記述されることで，両者の対応が取られている．利用者は，まずアンサンブル XML を検索し，そこからコンフィギュレーションを経由して，必要なバイナリファイルを取得することになる．

本研究では，QCDml のアンサンブル XML を対象にファセット検索インタフェースを構築す

☆ http://flamenco.berkeley.edu/

☆☆ http://www.lqcd.org/ildg/

☆☆☆ http://www.lqcd.org/ildg/QCDml/



```
<markovChain xmlns="http //www lqcd org/ildg/QCDml/ensemble1 4"
xmlns xsi="http //www w3 org/2001/XMLSchema-instance"
xsi schemaLocation="http //www lqcd org/ildg/QCDml/ensemble1 4
http //www lqcd org/ildg/QCDml/ensemble1 4/QCDmlEnsemble1 4 1 xsd">
  <markovChainURI>mc //JLDG/CP-PACS/RCNF2/RC12x24-
B1800K014090C1600</markovChainURI>
      <management>
    <revisions>1</revisions>
    <collaboration>CP-PACS</collaboration>
    <projectName>RCNF2  Nf=2 full QCD with iwasaki RG gauge and tadpole
improved clover quark action </projectName>
    <ensembleLabel>B1800</ensembleLabel>
    <reference>Phys Rev  D65  2002  054505  hep-lat/0105015 , Erratum-
ibid  D67  2003  059901</reference>
    <archiveHistory>
      <elem>
        <revision>1</revision>
        <revisionAction>add</revisionAction>
        <participant>
          <name>T Yoshie</name>
          <institution>Center fof Computational Sciences, University of
Tsukuba</institution>
        </participant>
        <date>2006-06-01T12 00 00Z</date>
        <comment></comment>
      </elem>
    </archiveHistory>
  </management>
  <physics>
    <size>
      <elem>
        <name>X</name>
        <length>12</length>
      </elem>
      <elem>
        <name>Y</name>
        <length>12</length>
      </elem>
      <elem>
        <name>Z</name>
        <length>12</length>
      </elem>
      <elem>
        <name>T</name>
        <length>24</length>
      </elem>
    </size>
    <gluon>
        <iwasakiRGGluonAction>
          <glossary>http //www jldg org/JLDG/CP-
PACS/Glossary/iwasakiRGGluonAction pdf</glossary>
```

```
          <gluonField>
            <gaugeGroup>SU 3 </gaugeGroup>
            <representation>fundamental</representation>
            <boundaryCondition>
              <elem>periodic</elem>
              <elem>periodic</elem>
              <elem>periodic</elem>
              <elem>periodic</elem>
            </boundaryCondition>
          </gluonField>
          <beta>1 800</beta>
          <normalisation>cs_sum_to_one</normalisation>
          <c0>3 648</c0>
          <c1>-0 331</c1>
          <c2>0 0</c2>
          <c3>0 0</c3>
        </iwasakiRGGluonAction>
      </gluon>
      <quark>
        <tpCloverQuarkAction>
          <glossary>http //www jldg org/JLDG/CP-
PACS/Glossary/tpCloverQuarkAction pdf</glossary>
          <quarkField>
            <normalisation>sqrt2kappa</normalisation>
            <boundaryCondition>
              <elem>periodic</elem>
              <elem>periodic</elem>
              <elem>periodic</elem>
              <elem>periodic</elem>
            </boundaryCondition>
          </quarkField>
          <numberOfFlavours>2</numberOfFlavours>
          <kappa>0 14090</kappa>
          <cSW>1 600</cSW>
          <u0>0 854306833</u0>
        </tpCloverQuarkAction>
      </quark>
    </action>
  </physics>
  <algorithm>
    <name>CP-PACS-hybrid-monte-carlo</name>
    <glossary>http //www jldg org/JLDG/CP-PACS/Glossary/CP-PACS-hybrid-
monte-carlo pdf</glossary>
    <reference>Phys Rev  D65  2002  054505  hep-lat/0105015 , Erratum-
ibid  D67  2003  059901</reference>
    <exact>true</exact>
  </algorithm>
</markovChain>
```

図 3　QCDml のアンサンブル XML の例 (Ensemble-RC12x24-B1800K014090C1600.xml).

ることを考える．アンサンブル XML は，1 ファイルが一つの markovChainURI に対応するため，必要な条件を見たすアンサンブル XML ファイルを特定し，その markovChainURI を取得することが検索の目的となる．

3. XML データを対象にしたファセット検索

近年，多くの情報が XML 形式で流通している．このような情報に対して探索的な検索を可能にするため，XML データにファセット検索を適用することを考える．

3.1 XML データからのファセット抽出

3.1.1 問　題　点

通常のオブジェクトを対象とした場合，検索対象となるオブジェクトは予め与えられていることが前提である．また，各オブジェクトの持つファセット（属性）も明確に示される場合が多い．これに対し XML の場合，その半構造性から次のような問題が生じる．

(1) 検索対象の粒度がまちまちである．ものによっては，文書（ファイル）単位で検索したいかもしれないし，特定の要素を検索したいことも多い．QCDml の場合，検索対象は markovChainURI で特定される，文書全体，すなわち一つのファイルになる．

(2) データの構造を規定するスキーマの種類によって，規則性の強い構造を持つ場合と，ゆるい制約しか持たない構造，あるいはその両者を持つことがある．前者の場合であれば，通常のオブジェクトに類する形で扱うことができるが，後者の場合，オブジェクトを明確に規定できない可能性がある．QCDml の場合，規則性の強い構造を持っているので，この点は非常に扱いやすい．

(3) 各オブジェクトについて，ファセットに基づく分類を行うため，ファセットに関する値（ファセット値と呼ぶ）を抽出する必要があるが，上で述べたのと同様の理由でファ

セット値を明確に規定することが難しい場合がある．例えば，出現するXML要素やXML属性が，親要素の種類に応じて変化するようなXMLが考えられる．この場合，出現する親要素の種類に応じて異なる種類のファセット値を扱う必要があるため，問題が複雑になる．例えば，QCDmlの場合，Gluon Actionが重要な一つのファセットになる．それは，

```
<beta>1.800</beta>
<normalisation>
  cs_sum_to_one
</normalisation>
<c0>3.648</c0>
<c1>-0.331</c1>
<c2>0.0</c2>
<c3>0.0</c3>
```

という一連のパラメータを取るが，このパラメータの組み合わせは親要素である`iwasakiRGGluonAction`に依存している．別のアクションが親要素の場合，出現するパラメータ集合も異なるものになる．

(4) 各ファセットについて，それが文字列や数値などの単純値を取る場合と，部分XMLデータとして構造を持つ場合がある．後者の場合，部分XMLデータの値をどのように扱うかを検討する必要がある．QCDmlの場合，格子サイズ：

```
<size>
  <elem>
    <name>X</name>
    <length>12</length>
  </elem>
  <elem>
    <name>Y</name>
    <length>12</length>
  </elem>
  <elem>
    <name>Z</name>
    <length>12</length>
  </elem>
  <elem>
    <name>T</name>
    <length>24</length>
  </elem>
</size>
```

がそのような例にあたる．これらを扱うには，

- 配下のテキスト値を単に連結して表示する (X12Y12Z12T24)
- ファセットに応じたルールを与え変換する (12/12/12/24)

といった対応が考えられ，適切な対応を行なう必要がある．

(5) ファセットが取る値として，XML要素のテキスト値だけではなく，要素名や属性名などを取る場合がある．QCDml の場合，`/markovChain/action/gluon`直下にある`iwasakiRGGluonAction`要素の要素名自身が重要な情報を含んでおり，同じ位置にある他の要素も含めて適切に集約処理する必要がある．

3.1.2 抽出手順

ここまでの議論を踏まえて，XMLデータからのファセット抽出について検討する．まず前提として，検索対象オブジェクトはXPath式あるいはXQuery問合せで特定可能な要素であるとする．

いったんオブジェクト（に対応する要素の）集合が特定できれば，基本的にはその配下の全ての子要素とその属性がファセット値の候補となる．ファセット検索インタフェースを構築するプロセスにおいて，可能なファセットをこのような形でリストアップし，インタフェースの設計者に必要な項目を選択させることで，インタフェースの概要を決定することができる．

上で述べた各問題については，以下のような対応が考えられる．なお今回はQCDmlを念頭に置いているため，(1), (2) については対象としない．

(3) 検索対象となる全てのオブジェクトから，可能性のあるあらゆるファセットとファセット値の組合せを抽出，それを設計者に選択させることで対応可能である．

(4) インタフェースの詳細を詰める段階で，部分XMLデータからファセット値の写像方式を選択させる．基本的には，

- テキスト値のみを連結する
- 要素名とテキスト値を連接する

– 設計者が用意する変換スクリプト (XSLT や XQuery 等) を適用する

から選択させれば，実際の応用で生じるほとんどのケースに対応することができると考えられる．

(5) これも，(4) と同様に，ファセット抽出の段階で可能性のある全ての値を選択しておき，インタフェースの詳細を決める段階でどのように値に写像するかを決めることで対応できる．

3.2 ファセット値の抽出

ファセットを決定すると，次は各ファセットで取りうる値を XML データから抽出し，集約処理を行なう必要がある．各オブジェクトは XQuery (XPath) で特定されるので，そこからの相対パスでファセット値に到達することができる．

ファセット値がテキストや数値などの単純な値の場合は，それをそのまま取得する．部分 XML データである場合は，上で設計者によって示された方法に従って値への写像を行ない，それをファセット値とする．

3.3 ファセット値の集約処理

各ファセットについて，インタフェースに提示するための集約処理を行う必要がある．具体的には，各ファセットにおいて代表的な値を選択し，それぞれの代表値の出現頻度をカウントする．

4. 提案システムの概要

XML データに対するファセット検索を実現するためのシステム構成について述べる．図 4 にその概略を示す．システムは，

- Web サーバ
- ファセット検索モジュール
- XML データベース
- ファセットデータベース

から構成される．検索対象の XML データは予め XML データベースに格納されており，そこから前述の方法でファセットと関連する値を抽出する．得られた値はファセットデータベースに格納する．これは抽出されたデータの形式にもよるが，XML データベースかリレーショナルデータベースのいずれかである．

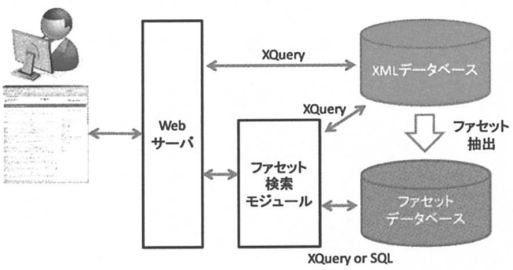


図 4 システムの概要．

Web サーバは，ページの構成に従って初期ページを表示し，利用者からの入力を待つ．利用者の操作はファセット検索モジュールに転送される．

ファセット検索モジュールは，ファセットデータベースと XML データベースに対して適切に問合せを行い，ファセット検索の過程をサポートする．具体的には，利用者からの指示を絞り込みの条件に読み替え，ファセットデータベース (XML データベース) への問合せを絞り込み条件に従って書き換え，データベースに問合せを行う．得られた結果から，最新の結果を生成し，Web サーバに返却する．

5. 関 連 研 究

構造化データを対象としたファセット検索に関する研究としては，Flamenco[4], mSpace[5], Ontogator[6] などがある．その他にも多くの Web サイトでファセット検索をサポートしている[7]．

上記以外のデータを対象とした例として，Oren 等はグラフ構造を持つ RDF データを対象にしたファセット検索を提案している[8]．グラフ構造を有する RDF データを対象とするため，RDF に対する各種操作 ((inverse) basic selection, (inverse) existential, (inverse) non-existential, (inverse) join, intersection) を形式的に与えるとともに，結果のランキング手法を検討している．XML を対象とした我々の手法とは，ベースとなるモデルが，XML は木構造であるのに対して，RDF はグラフ構造を有するという点が異なる．

6. ま と め

本稿では，XML データを対象にしたファセッ

ト検索インタフェースの生成手法について議論した．具体的には，素粒子分野における ILDG (International Lattice Data Grid) のアンサンブル XML データを念頭に，XML データにファセット検索を適用する際の問題点とその対応について検討した．また，XML データのファセット検索を実装するためのシステム構成について触れた．

今後はシステムの実装を行いながら，さらに問題点を明らかにして行く予定である．また，他の研究で行われているような，選択演算以外の問合せセマンティクスの検討や，処理の効率化についても検討したい．

謝辞 本研究の一部は科学研究費補助金 若手研究 (B)(#19700083) によるものである．ここに記して謝意を示す．

参 考 文 献

1) Gorton, I., Greenfield, P., Szalay, A. and Williams, R.: Data-Intensive Computing in the 21st Century, *IEEE Computer*, Vol. 41, No. 4, pp. 30–32 (2008).
2) W3C: Extensible Markup Language (XML) 1.0 (Fourth Edition), `http://www.w3.org/TR/xml/` (2006). Recommendation.
3) White, R. W., Kules, B., Drucker, S. M. and m.c. schraefel: Supporting exploratory search: Introduction, *Communications of the ACM*, Vol. 49, No. 4 (2006).
4) Yee, P., Swearingen, K., Li, K. and Hearst, M.: Faceted Metadata for Image Search and Browsing, *Proc. ACM CHI 2003*, pp. 401–408 (2003).
5) m.c. schraefel, Wilson, M., Russell, A. and Smith, D. A.: mSpace: improving information access to multimedia domains with multimodal exploratory search, *Communications of the ACM*, Vol. 49, No. 4, pp. 47–49 (2006).
6) Mäkelä, E., Hyvönen, E. and Saarela, S.: Ontogator —A Semantic View-Based Search Engine Service for Web Applications, *The 5th Int'l Semantic Web Conference (ISWC 2006)*, pp. 847–860 (2006).
7) Ley, M.: DBLP Bibliography, `http://www.informatik.uni-trier.de/~ley/db/`.
8) Oren, E., Delbru, R. and Decker, S.: Extending faceted navigation for RDF data, *The 5th Int'l Semantic Web Conference (ISWC 2006)*, pp. 559–572 (2006).