

メタデータを用いたモバイル向け P2P 検索システムの設計と実装

角野 宏光[†] 石川 憲洋[†] 小俣 栄治[†]
ヨハン イェルム^{††} 宮津 和弘^{†††} 村上 慎吾^{†††}

近年、携帯電話などのモバイル端末における音楽、静止画、動画などのマルチメディアコンテンツ利用が急速に普及しており、モバイル端末を利用したマルチメディアコンテンツの簡単かつ効果的な検索に対する要求が高まっている。しかし、現状ではモバイル端末の表示、入力インタフェースおよび回線容量の制限などの特徴を考慮した利便性の高いコンテンツ検索サービスが提供されているとはいえない。そこで、本稿ではメタデータを用いてモバイル端末に対して精度の高い検索サービスを提供し、さらにモバイル端末や PC などの様々な検索ノードがオーバレイ・ネットワークを構成し相互に連携して検索を行うモバイル向け P2P コンテンツ検索システムを設計した。本稿では、本システムの構成技術としてモバイル向けコンテンツのメタデータ定義、メタデータ検索言語の設計、レイテンジ方式の提案について述べる。さらに、本検索システムのプロトタイプについて紹介する。

Design and Implementation of a Mobile P2P Content Search System Using Metadata

HIROMITSU SUMINO,[†] NORIHIRO ISHIKAWA,[†] EIJI OMATA,[†]
JOHAN HJELM,^{††} KAZUHIRO MIYATSU^{†††} and SHINGO MURAKAMI^{†††}

As the use of multimedia content by mobile phones continues to spread, we can expect a growing need for services that allow users to search for and exchange photos and other content stored on their mobile phones. Performing a search from a mobile phone requires that target content be discovered with minimal input and interaction, however there was no service which provides an enjoyable search environment for mobile-phone users. With this in mind, we have developed a multimedia-content search system for mobile phones that uses metadata and runs on a P2P network. To implement such metadata searching, we first defined metadata in relation to content for mobile phones, and designed a query language. In addition, we proposed the application of fuzzy logic as a scheme for returning appropriate results that reflect user intent even if the user should specify vague search conditions. These technologies have been implemented on a prototype system.

1. はじめに

近年、携帯電話などのモバイル端末は著しい発展をとげており、モバイル端末からのインターネット利用ならびに音楽、静止画、動画などのマルチメディアコンテンツ利用が急速に広まっている。また最近では、マルチメディアコンテンツを検索し利用する手段として Gnutella¹⁾ のような P2P ネットワークを利用したファイル交換サービスが人気を集めている。このよう

な P2P サービスは、サーバを用いずにユーザ同士で P2P ネットワークを構成して各ユーザの保持するコンテンツの検索を行うものである。近年ではカメラ付きの携帯電話なども普及しており、将来は携帯電話で保持する写真などのコンテンツを交換することも想定される。このようなモバイル向けコンテンツの普及にとともに、モバイル端末からの簡単かつ効果的なマルチメディアコンテンツ検索に対する要求が高まっている。

インターネットの世界では Google²⁾ や Yahoo³⁾ などのキーワード入力によるテキスト検索が普及しているが、これらの検索サービスを利用する場合、ユーザが試行錯誤でキーワード入力と検索結果の参照を繰り返して検索結果の絞り込みを行う必要がある。携帯電話などのモバイル端末では、表示、入力インタフェースおよび回線容量の制限があるため、デスクトップ PC

[†] 株式会社 NTT ドコモ
NTT DoCoMo, Inc.

^{††} エリクソンリサーチ
Ericsson Research

^{†††} 株式会社日本エリクソン
Nippon Ericsson K.K.

などで利用される一般的なコンテンツ検索のように大量の検索結果から満足する検索結果を絞り込む作業がユーザにとって大きな負担となる。そのため、モバイル端末向けの検索では以下の要求条件を考慮する必要がある。

- インタラクションの削減
なるべく少ないインタラクションでユーザの検索条件に適合した検索結果を表示し、ユーザの負担を軽減することが要求される。
- 通信量の削減
現状のモバイル通信は、固定通信に比べ回線速度が遅く、また利用コストが高いため検索で利用する通信量を軽減することが要求される。
- 入力インタフェースの簡易化
携帯電話のようなモバイル端末は、デスクトップ PCなどに比べると入力インタフェースの操作性が劣るため、できるだけ少ない入力で検索条件を与えることが要求される。

上記の要求条件を考慮した利便性の高いモバイル端末向け検索サービスの実現が期待されている。

一方、インターネットの分野においても、爆発的に増加するコンテンツに対する効率的な検索手段の提供が重要な課題であり、様々な研究開発が行われている。なかでも、近年 W3C (World Wide Web Consortium) により提案されたセマンティック Web⁴⁾ は、次世代 Web 技術の候補として注目を浴び、欧米を中心に活発に研究が進められている。セマンティック Web は、Web サイト上のコンテンツにメタデータを付加してソフトウェアで自動処理を行う技術である。セマンティック Web 技術を用いた検索システムでは、コンテンツの内容を説明するきめ細かなメタデータを用いることで、ユーザの要求に近い検索結果を得ることが可能となる。しかし、メタデータを用いたモバイル向けコンテンツ検索において、検索の精度を向上させるためには、以下の項目に関してさらに検討が必要である。

(1) メタデータ定義

メタデータを用いた検索では、コンテンツの特徴を的確にメタデータに表現する必要がある。従来の代表的なメタデータとして、マルチメディアコンテンツのメタデータ定義である MPEG-7 や、図書情報を記述する DublinCore⁵⁾ などがあるが、モバイル向けの検索について考慮されたものはない。そこで、モバイル向けコンテンツの特徴を考慮したメタデータの定義が必要である。

(2) 検索言語

P2P ネットワーク上で利用する検索言語は、P2P ネットワーク上のデータベースを相互に操作するための機能を備える必要がある。また、モバイル向け検索の要求条件であるインタラクションの削減を実現するためには、検索条件でユーザの意図をできるだけ的確に表現し、検索結果の絞り込みにかかる作業を減らすことが重要である。しかしながら、現状ではメタデータを検索するための十分な機能や記述能力を備えた検索言語がない。

(3) レイティング

検索で複数の結果が得られた場合、モバイル端末の表示能力に応じた適切な件数の結果を表示するために、コンテンツのレイティングを行いユーザの要望に近いコンテンツを優先的に表示することが重要である。また、ユーザの意図する検索条件が 1 つのキーワードで表現できるものでなく、ある程度の幅を持つ場合、従来の一般的なメタデータ検索では条件を変えて何度も検索を行う必要があるが、モバイル端末向けの検索では、このようなあいまいな条件で検索する際にユーザの負担を軽減させる工夫が必要である。

本研究では、上記項目に関して検討を行い、モバイル端末に対して効率的なコンテンツ検索を提供するための検索システムを提案する。本稿では、まず、モバイル向けコンテンツを表現するためのメタデータのボキャブラリの定義、ユーザの意図を検索条件としての的確に表現するためのメタデータ検索言語の設計、ユーザの要求に応じて適切な検索結果を提示するためのレイティング方法の提案について述べる。次にこれらの技術を用いた検索システムのプロトタイプの実装および評価について述べ、最後にまとめる。

2. メタデータの設計

P2P ネットワークのような分散型のネットワークにおいて、ノード間でメタデータ問合せを行うためには、ノード間で相互利用可能なメタデータ形式が必要となる。そこで、本研究ではメタデータの記述形式として RDF (Resource Description Framework)⁶⁾ を用いることとした。RDF はセマンティック Web の標準的なメタデータ記述形式として W3C により標準化が進められており、今後広く普及する可能性のある有望な記述形式である。また、RDF は、独自に定義したメタデータとともに既存のメタデータを流用することが可能であるため、様々なシステムにおいてデータの相互利用が可能となる。さらに、RDF ではオントロジ適用の枠組みが考慮されているため、将来の検索

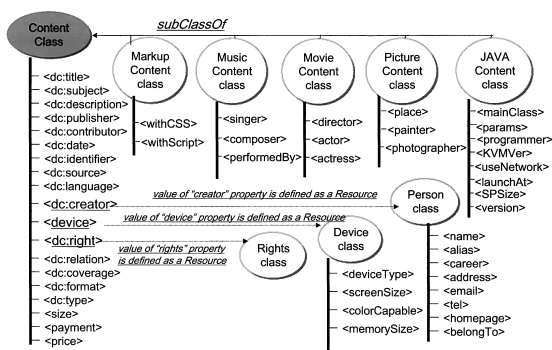


図 1 メタデータ要素定義

Fig.1 Metadata definition for mobile multimedia contents.

```

<?xml version="1.0"?>
<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#"
xmlns:DE="http://www.....com/./schema#"
xmlns:dc="http://purl.org/dc/elements/1.1/">
<rdf:Description about="http://.../othellojam">
<rdf:type rdf:resource="http://.../JAVAContent"/>
<dc:title>Othello</dc:title>
<dc:subject>game, puzzle, easy</dc:subject>
<dc:description>puzzle game</dc:description>
<dc:date>Thu,01Feb2001 05:15:32</dc:date>
<dc:creator rdf:resource="http://.../Creator/s/001"/>
<DE:device rdf:resource="http://.../Device/x503"/>
<DE:size>7556</DE:size>
<DE:KVMVer>CLDC-1.0</DE:KVMVer>
<DE:price>100Yen</DE:price>
</rdf:Description>
<rdf:Description about="http://.../Creator/s/001">
<rdf:type rdf:resource="http://.../schema#Person"/>
<DE:name>DoCoMo</DE:name>
<DE:address>Hikarinooka Yokosuka</DE:address>
<DE:email>hohgehoge@nttdocomo.co.jp</DE:email>
</rdf:Description>
<rdf:Description about="http://.../Device/x503">
<rdf:type rdf:resource="http://.../schema#Device"/>
<DE:deviceType>x503</DE:deviceType>
</rdf:Description>
</rdf:RDF>
    
```

図 2 メタデータ記述例 (i アプリ)

Fig. 2 An example of metadata (iappli).

の高度化が期待できる。たとえば，“ピア・ツー・ピア”のような長い文字を“P2P”のような簡潔な表現へ自動的に置き換えることなどが考えられる。このような RDF の特徴を生かすことにより，モバイル環境に適した検索の実現が期待できる。

本研究では，RDF を用いて現在携帯電話向けインターネットサービスで用いられている HTML コンテンツ，JAVA コンテンツ，音楽コンテンツ，画像コンテンツに関してメタデータのボキャブラリを定義した。図 1 にボキャブラリの定義を示す。本設計では，できるだけ標準的なメタデータ要素を用いることとし，基本的な項目については現在メタデータ標準として最も有名な DublinCore の項目を用いた。これにより，DublinCore で定義された既存のコンテンツのメタデータをモバイル端末においてそのまま利用することが可能となる。DublinCore のメタデータ定義は Title，Date，Creator などの 15 項目の基本要素が定義されており，これらの基本要素はコンテンツの種別によらず定義が可能のため，すべてのコンテンツに共通の要素として用いることとした。また，モバイル端末は端末の種類に応じて閲覧可能なコンテンツのサイズやファイル形式に制限がある。特に JAVA コンテンツなどは，端末のバージョンにより利用可能なコンテンツが異なる。そこで，本稿で設計したメタデータ定義では，端末で利用可能なファイルサイズ (filesize) や対応する端末の種類 (device) に関する情報をメタデータで記述し，検索の際に利用できないコンテンツを検索結果として返さないようにした。また，これらコンテンツの種別に特有の項目について，種別ごとに要素を定義した。たとえば，NTT ドコモが提供している i モードサービス⁷⁾ の JAVA アプリケーションである i アプリコンテンツに対しては ADF (Application Descriptor File) の項目 (deviceType, KVMVer

など)を追加した。ADF は i アプリコンテンツのメタデータを記述したファイルであり，i アプリコンテンツをダウンロードする際に i モード端末により読み込まれるものである。すべての i アプリコンテンツに対して必ず ADF が存在するため，この ADF を利用して RDF メタデータを生成することも可能である。さらに，携帯電話で利用されるマルチメディアコンテンツには有料のコンテンツが多く存在するため，どの程度の金額であるかなどコンテンツの料金を意識した検索が要望される。そこでメタデータにおいてコンテンツの課金情報に関する要素 (price, payment) を定義し，ユーザの検索条件において料金を基に検索することを可能とした。本定義を用いた i アプリコンテンツのメタデータ記述例を図 2 に示す。

RDF の文書を記述する際に使うリソースやプロパティのデータ型 (データ構造) の定義を行うものとして W3C において RDF Schema が制定されている⁸⁾。RDF Schema 仕様では，いくつかの特徴を持ったリソースを分類するための抽象的なリソース (クラス) を定義しており，クラス間の関係を記述するルールやプロパティのとりうる値やそのプロパティを持ちうるクラスの制約条件などを定義している。クラス間の包含関係は subclassOf で表され，プロパティ間の包含関係は subPropertyOf で表される。図 3 に RDF スキーマを用いて定義した本メタデータのボキャブラリの関係について示す。本メタデータでは，JavaContentfClass と MusicContentClass を ContentClass のサブクラスとして定義することで，JavaContentClass および MusicContentClass のあるインスタンス (実例) を ContentClass のインスタンス (実例) として扱うことが可能である。また，programmer および composer

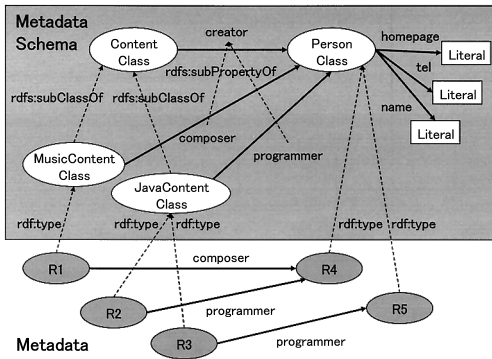


図 3 メタデータの構成
Fig.3 Structure of metadata.

のプロパティを creator のサブプロパティと定義することで, programmer および composer のプロパティは creator プロパティと同様に扱うことが可能である. このような処理は推論とよばれ, RDF Schema の特徴の 1 つを利用したものである.

3. RDF 検索言語 xRQL の設計

3.1 要求条件

RDF のメタデータを用いた検索を実現するためには, 検索ノードにおいて RDF を処理するためのデータベースを用いる必要がある. RDF を扱うデータベースとして RSSDB⁹⁾, rdfDB¹⁰⁾ などいくつかの研究機関から実装例が報告されているが, 現状ではこれらのデータベースにおいて標準的に利用可能な RDF 検索言語が存在せず, W3C の RDF Data Access Working Group で標準化作業を行っている. そこで, まず分散型の RDF メタデータ検索を実現するための, RDF 検索言語の要求条件について検討した. 以下に要求条件を述べる.

(1) RDF メタデータの操作

ノード間で相互にメタデータの登録や更新を行うために, 検索言語は基本的な RDF メタデータの検索機能に加え, RDF メタデータの作成, 変更, 削除の機能をサポートする必要がある. これらの操作は RDF のメタデータを作成したいユーザにとって有効な機能である.

(2) RDF データモデルへの対応

RDF 検索言語は RDF のデータモデル(リソース, プロパティ, 値)に対応し, データモデルを忠実に表現可能とすることが望ましい. また, RDF Schema で定義されるオブジェクト指向のモデルと同様のクラスやプロパティの階層化, 継承の概念に対応することが望まれる. 検索言語の表現能力が乏しい場合は Query

を複数回発行する必要があるが, RDF データモデルを表現することにより, 1 つの Query においてユーザの意図を的確に表現することが可能となるため, 検索のシーケンスを減らし, ユーザの負担やネットワークへの付加を減らすことが可能となる.

(3) 可読性

分散された様々なメタデータを簡単かつ効率的に扱うために, 人間にとって読みやすくマシンにおいても処理しやすい形式が求められる.

(4) 推論

RDF スキーマで定義される語彙の階層化に対応した処理を行うために, 検索言語においてクラスやプロパティの関係を推論して処理する機能を含むことが望ましい.

(5) スキーマ検索機能

様々なスキーマで定義されたメタデータを利用するためには, RDF のデータ構造を定義するスキーマについて把握する必要がある. そこで, RDF の検索言語は RDF のメタデータのスキーマに関しても同様に検索可能とする必要がある. これらのスキーマは RDF Schema を用いて定義され, 検索のクエリで用いられるプロパティやクラス間の関連についての情報を与える.

(6) 検索結果フォーマット定義

RDF の検索結果を取得したユーザにとって, 検索結果を簡単に処理するために, ユーザにとって利用しやすい XML 形式を定義できるよう検索結果のフォーマットを指定する機能が要求される. 結果自体をソフトウェアで自動処理しやすい XML 形式で返すことで, P2P ネットワークのような分散型のネットワーク上で結果を中継するノードにおいても検索結果を利用することが可能となる.

3.2 設計方針

我々は上記の要求条件を満たす RDF 検索言語として, xRQL (XML-based RDF Query Language) を設計した. xRQL は XML 形式を用いるため, メタデータの処理に用いる一般的な XML のツールで処理可能である. xRQL の機能としては, メタデータを扱うための 4 つの基本的な操作(検索, 作成, 削除, 変更)とスキーマの操作を定義した. 以下には, xRQL の設計方針について図 3 のメタデータの例を用いて説明する.

3.2.1 RDF メタデータの検索操作

xRQL では RDF のリソースを定義しているスキーマに関する知識を用いて検索の Query を記述することとする. 検索の操作において, まず我々は RDF の

データモデルに従ったパス表現を定義した .xRDF におけるデータのパス表現は RDF のグラフ構造に対応し、さらに人が理解しやすいように、グラフにおけるノードと矢印を順番に並べる形式とした。以下にパスの表現形式を示す。

$L = C1 \{p1\} C2 \{p2\} C3 \dots$

ここで、 $p1, \dots, pn$ および $C1, \dots, Cn$ は RDF データに対応するスキーマで定義されたプロパティおよびクラスである。パスの表現と関連する RDF リソースを結びつけるためには、パス中にリソースに対応した変数を用いる。例として、図 3 のようなスキーマにおいて、あるコンテンツ (Content クラスのリソース x) の作成者 (creator) である人物 (Person クラスのリソース y) をパス表現すると以下ようになる。

`Content:x{creator}Person:y`

ここで、変数 x は Content クラスのインスタンスであることを示す。このパス表現を基に x に対応するリソースを検索する場合、もし Content クラスの下に定義される subclass があれば、その subclass に属するリソースも取得可能とする。たとえば、図 3 に示す RDF スキーマにおいては、MusicContent と JavaContent のクラスに属するインスタンスについても Content クラスのインスタンスとして取得可能である。つまり、本検索言語では前章で述べたような階層化されたデータの推論に対応した検索が可能である。図 4 に xRQL の Query の例を示す。図 4 は、作者が "john Smith" であるコンテンツを問い合わせる例である。operator 要素で操作の指定 (検索の場合は select) を行い、location 要素でパスの指定を行う。condition 要素はデータマッチングに用いられる条件を与える。また、result 要素を用いて結果出力のフォーマット定義を指定する。usingnamespace 要素は操作の対象となるメタデータのスキーマを指定するために用いる。

3.2.2 結果のフォーマット定義

既存の RDF の検索言語は検索結果のフォーマットについて考慮されておらず、結果を取得し自動的に処理するためには、結果を機械で処理しやすい形式に変換するツールを用意する必要があった。そこで、xRQL では XML 形式を基にしたユーザの望む結果のフォーマットをユーザが自由に定義可能となるように設計した。図 4 の検索要求に示されるように、パス表現 `content:x{creator}person:y` における変数 x の値を結果として取得する場合について述べる。結果である x の値を `<content>` 要素の値として記述された形式で検索結果を取得したい場合に、以下のように記述する。

```
<xrql:xrql>
<xrql:operator>select</xrql:operator>
<xrql:location>
Content:x{creator}Person:y
</xrql:location>
<xrql:condition>
<xrql:item>y="john smith"</xrql:item>
</xrql:condition>
<xrql:result>
<content> x </content>
</xrql:result>
<xrql:usingnamespace>
<xrql:item>ns1="www.jp/schema"</xrql:item>
</xrql:usingnamespace>
</xrql:xrql>
```

図 4 検索操作例

Fig. 4 An example of query request.

```
<xrql:result>
<content>"www.content1.co.jp"</content>
<content>"www.content2.co.jp"</content>
<content>"www.content3.co.jp"</content>
</xrql:result>
```

図 5 検索操作例 (図 4) の結果

Fig. 5 An example of query result.

```
<result>
<content> x </content>
</result>
```

結果として複数値が得られた場合には、結果の入った `<content>` 要素が複数返される。図 5 は図 4 の検索要求に対する結果の例である。結果を xHTML 形式で表示した場合や、他のフォーマットに変換してその先のプロセスで利用する場合などにも、XSLT などを用いて簡単かつ効率的に変換することが可能である。

3.2.3 RDF メタデータの作成、削除、変更の操作
xRQL におけるメタデータの作成、削除、変更の操作に関しては、オブジェクト指向のコンセプトを用いて定義した。スキーマで定義されるクラスにリソースは属し、リソースは唯一の ID (URI) で識別されるオブジェクトである。図 6 の例では、オブジェクト R0 は title, creator, device というプロパティを持つオブジェクトであり、creator プロパティの値は別のオブジェクト R4 を示している。

ここで、削除の操作は RDF のレポジトリから RDF のリソースを取り除くために用いる。この操作により RDF のリソースそのものだけでなく、リソースに関連づけられたプロパティを含むステートメントも取り除かれる。図 7 には削除の操作の記述例を示す。operator 要素で delete を設定し、object 要素で操作対象のクラス (Person) を指定している。この場合 Person クラスの `www.person1.co.jp` というオブジェクトが削除される。

作成の操作は RDF レポジトリ内に新たに RDF の

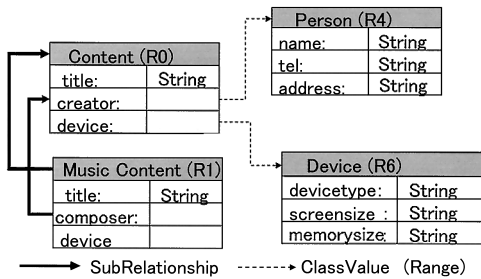


図 6 オブジェクト指向のデータ
Fig. 6 Object oriented data.

```
<xrql:xrql>
<xrql:operator>delete</xrql:operator>
<xrql:object>Person</xrql:object>
<xrql:location>Person.x</xrql:location>
<xrql:condition>
<xrql:item>x="www.person1.co.jp"</xrql:item>
</xrql:condition>
<xrql:usingnamespace>
<xrql:item>ns1="www.jp/schema"</xrql:item>
</xrql:usingnamespace>
</xrql:xrql>
```

図 7 リソース削除操作例
Fig. 7 An example of delete resource.

```
<xrql:xrql>
<xrql:operator>create</xrql:operator>
<xrql:object>Person</xrql:object>
<xrql:objectvalue>
<xrql:objectitem>
<xrql:resource>"www.person4.co.jp"</xrql:resource>
<xrql:propertyvalue>
<xrql:item>name="Richard Lim"</xrql:item>
<xrql:item>tel="010-1122-3344"</xrql:item>
<xrql:item>create="www.content.co.jp/content1"</xrql:item>
</xrql:propertyvalue>
</xrql:objectitem>
</xrql:objectvalue>
<xrql:usingnamespace>
<xrql:item>ns1="www.jp/schema"</xrql:item>
</xrql:usingnamespace>
</xrql:xrql>
```

図 8 リソース作成操作例
Fig. 8 An example of create resource.

リソースを作成する場合に用いる。図 6 に示すように、それぞれの RDF リソースはプロパティとプロパティ値で構成される 1 つのオブジェクトとして表現される。プロパティ値は他のオブジェクトもしくはリテラルである。RDF リソースが継承するプロパティはその parent クラスで定義される。たとえば、MusicContent オブジェクトの parent クラスである Content クラスより title, creator, device というプロパティを継承する。図 8 の例は Person クラスに属するオブジェクトを作成する操作を示している。

変更の操作はプロパティを変更するために用いられ、プロパティ値として新たにリテラルまたは関連するリソースを与える。この操作は別々に定義された RDF

グラフのリソースをプロパティにより関連づけする場合にも用いられる。

3.2.4 スキーマの操作

xRQL は RDF のスキーマで定義されたクラスやプロパティの関係を検索するために、以下の機能を提供する。

- subclassof : 所望のクラスのサブクラスを取得するために用いられる。たとえば, subclassof (Content) という記述では MusicContent, JavaContent, MovieContent など Content クラスのサブクラスと Content クラス自身を結果として返す。
- subpropertyof : あるプロパティのサブプロパティを取得するために用いられる。たとえば, subpropertyof (creator) は composer, programmer プロパティなどとそれ自身である create プロパティを返す。
- domain および range : あるプロパティのドメインとレンジを取得するために用いられる。たとえば, creator プロパティのドメインとレンジを取得したい場合には, それぞれ domain(creator), range(creator) と記述する。
- typeof : ある RDF のリソースまたはプロパティの属するクラスやプロパティを取得するための定義である。たとえば, 図 3 のデータでは typeof (R1) により MusicContent クラスが返される。

4. ファジーレイティング

4.1 概要

メタデータを用いた検索では、ユーザが入力した検索条件とメタデータに記述されたリソースの値が少しでも異なれば、検索結果として得ることはできないという課題がある。つまり、ユーザの要求とは非常に近い値を持つリソースや、複数の検索条件のうち適合しない条件が含まれるリソースなどは検索結果として得られない。そこで、モバイル端末からの入力に制限があることを考慮し、入力された条件に対してファジー理論を適用し意味的な幅を持たせることで検索条件をよりユーザ意図に近づける方法を提案する。さらに、携帯電話のような小さな画面には多くの検索結果を画面出力することは困難であるため、複数の検索結果が得られた際には結果をレイティングし、的確に絞り込みを行ったうえで、ユーザに提供する手法を提案する。本手法の手順は以下のとおりである。

- (1) ユーザが入力したあいまいな検索条件をもとに、検索を行う際に必要なメンバシップ関数を定義

する。

- (2) 検索条件を定式化する。
- (3) 検索結果のレーティングを行うために、各条件に対する検索対象データの評価値を算出する。
- (4) 算出した評価値を用いて、各検索結果の総合評価値を求める。
- (5) 総合評価値をソートすることにより、ユーザの満足度が高い結果のみを選び出す。

4.2 検索条件のファジー化

以下に検索条件のファジー化について説明する。ユーザが持っているあいまいな要求で、検索範囲がどこからどこまでと明確に定めることは困難である。そこで、その検索範囲となるあいまいな境界を、ファジー理論を用いることにより定式化する。ファジー理論を用いた場合、一致と不一致の間に中間値を持つことができる。この中間値により、検索条件とプロパティ値が完全に一致していなかったとしても、ユーザの要求に近いメタデータに関して、検索結果として選出することが可能となる。そこでまず、ユーザが要求するあいまいな条件を定式化するために、条件ごとにメンバシップ関数を定義する。メンバシップ関数とは、検索条件と検索対象のメタデータが保持するプロパティバリューがどの程度適合しているかを 0~1 の範囲で示す関数である。一般的に三角形・台形・ガウス分布型などのメンバシップ関数が存在する。

あいまいな条件の定式化手法について、音楽コンテンツを検索する場合の具体例を用いて説明する。まず、ユーザがある音楽ファイルを検索する際に以下の検索条件を与える場合について考える。

- 条件 1: ファイルのサイズは 100 KB くらい
- 条件 2: 値段は 100 円以内

ユーザが要求した検索条件には「~くらい」や「~以内」といったあいまいな単語が含まれている。このユーザの入力した条件ごとにメンバシップ関数を定義し、検索条件を定式化する。ここでは条件に「~くらい」が含まれている場合は三角型メンバシップ関数を用い、「~以内」が含まれている場合は台形型メンバシップ関数を用いる。メンバシップ関数を定義する際に、検索条件にあいまいさを付与するため幅を持たせるが、その際の検索条件幅は、ユーザ要求の $\pm\alpha\%$ とする。図 9 に、 $\alpha = 20$ として定義したメンバシップ関数の例を示す。このようにメンバシップ関数を定義し、検索条件を定式化することにより、これまでの RDF メタデータ検索では実現できなかった、あいまいな条件による検索が可能となる。

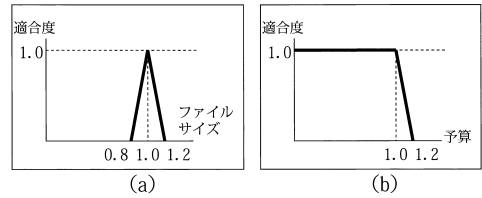


図 9 メンバシップ関数
Fig. 9 Membership function.

4.3 レーティング

なお、複数の条件を用いた場合には、前述のメンバシップ関数と検索条件から求まる各条件に対するメタデータごとの適合値を算出する。さらに、算出した条件ごとの適合値とファジー測度を用いることにより、ファジー積分を適用する。ファジー積分を用いることにより、複数の条件（属性）から総合評価値を得ることが可能となる。ファジー積分にはいくつかの手法（ショケ積分・菅野積分など）が存在するが、本研究では最も一般的な手法であるショケ積分を用いることにする。ショケ積分は各適合値を測度によって重みづけし、各評価の和を求め総合評価を出す手法である¹¹⁾。ここで A はファジー測度、 t_i は各条件の評価値として、式 (1) にファジー積分（ショケ積分）の一般式を示す。

$$\int_x f d\mu = \sum_{i=1}^M \mu(A_i)(t_i - t_{i-1}) \quad (1)$$

5. プロトタイプ実装

5.1 システム概要

提案システムの有効性を実証するために、JAVA の J2SE1.4.1 を用いて PC 上のアプリケーションとしてプロトタイプシステムを実装した。ここで、P2P ネットワークを構成する P2P プラットフォームは様々なアプリケーションで利用することを目的として、我々が研究を進めている P2P プラットフォームを用いた¹²⁾。

将来的には携帯電話のようなモバイル端末が P2P ノードとしての機能を提供することを想定しているが、現状の携帯電話では P2P 通信機能が提供されていないため、今回は図 10 に示すように P2P ノードの機能をモバイル Proxy に持たせ、モバイル Proxy 経由で携帯電話を参加させる構成で実装を行った。

5.2 検索ノード構成

メタデータを提供する検索ノードの構成について図 11 に示す。検索ノードは、コンテンツデータベース、メタデータデータベース、検索エンジンおよびメッセージ転送を行う P2P プラットフォームで構成される。また、検索ノードはユーザに対して携帯電

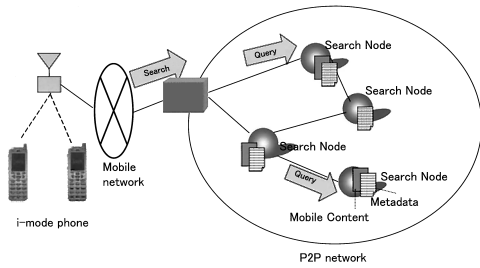


図 10 検索システム概要

Fig. 10 Overview of the search system.

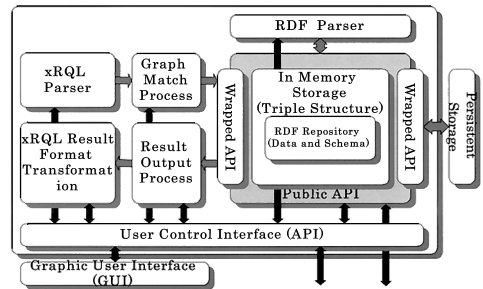


図 12 xRQL サーチエンジン構成

Fig. 12 Structure of the xRQL search engine.

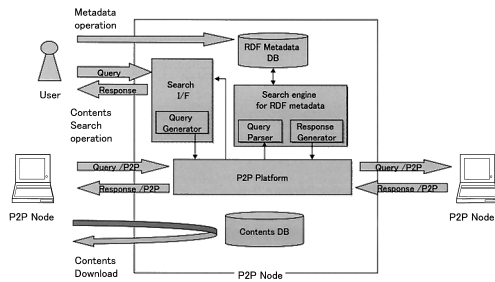


図 11 検索ノード構成

Fig. 11 P2P search node on P2P network.

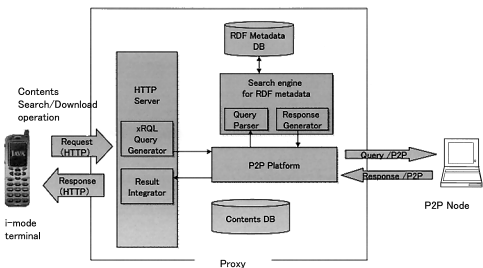


図 13 モバイル Proxy 構成

Fig. 13 Structure of mobile proxy.

話や PC などで広く利用できる Web ベースのインタフェースを提供し、ユーザはブラウザを用いて検索を行う。検索ノードはユーザからの検索要求を受けると、検索条件に記述された情報をもとに RDF メタデータ検索を行い、該当するコンテンツのメタデータを結果としてユーザに返す。また、ユーザへ提供するインタフェースはメタデータ定義に基づいて用意され、かつモバイル端末などの小さな画面でも利用しやすい構成とする。各ノードで保持するコンテンツはコンテンツデータベースに蓄積され、コンテンツの内容を記述したメタデータがメタデータデータベースに蓄積される。ただし、コンテンツを蓄積するノードとメタデータを蓄積するノードは同じノードである必要はない。メタデータにはコンテンツの所在としてノードの URL を記述し、ユーザは検索結果の URL で提示されたノードに対してコンテンツの取得を行う。検索エンジンの構成を図 12 に示す。検索エンジンは RDF パーサ、API、グラフ照合処理、結果出力処理とストレージにより構成される。既存のほとんどの RDF データベースはリレーショナルデータベース上で実装しているが、本システムでは RDF のデータを text ファイルとしてメモリ上に保存する。本研究では携帯電話のようなモバイル端末からのメタデータ検索を想定しており、将来的にモバイル端末への実装の実現を目指し軽量化を図った。

5.3 モバイル Proxy

モバイル Proxy の構成を図 13 に示す。モバイル Proxy は i モード端末に対して HTML/HTTP のインタフェースを提供し、ユーザが検索用の画面より検索条件を入力すると、検索条件のパラメータを抽出し、xRQL の Query を生成する。次に、モバイル Proxy は P2P ネットワーク上の検索ノードに対して P2P プロトコルを用いて Query メッセージを送信し、Query Response メッセージを受信する。P2P から受信した Query Response メッセージから検索結果を表示するための HTML 形式のページを生成して i モード端末に送信する。本プロトタイプでは、複数の検索ノードより到着する複数の QueryResponse メッセージを統合した後に検索結果の通知を行う。

5.4 ユーザインタフェース

図 14 には、モバイル Proxy で用意した i モード端末用のユーザインタフェースを示す。検索用の画面では、携帯電話の端末から利用しやすいように、プロパティを選択しそのプロパティの値を条件として入力することができるようにした。検索を実行する際にコンテンツの種類を選択すると、選択された種類のコンテンツのメタデータがどのスキーマに対応するかを判断し、対応するスキーマのプロパティのうち検索に利用するプロパティをリストボックスとして検索条件の入力ページに表示する。ユーザはリストの中から検索条

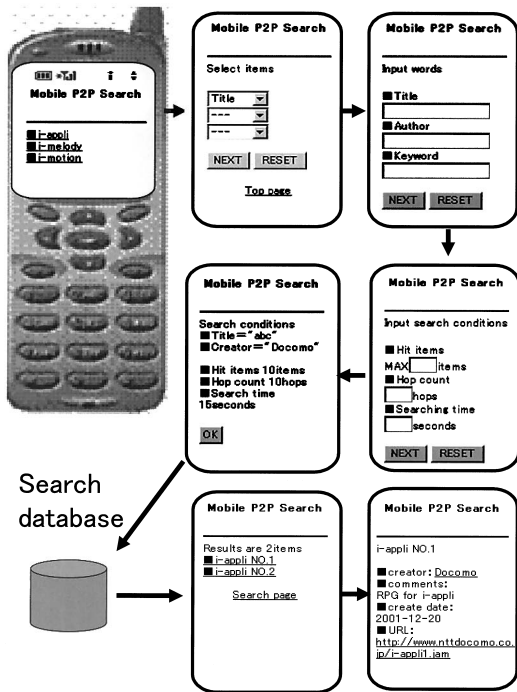


図 14 携帯電話用ユーザインタフェース
Fig.14 User interface for mobile phone.

件として用いるプロパティを選択するだけでよく、プロパティの文字列を入力する必要がなくなる。このように動的に検索画面のページを生成することにより、ユーザの入力を簡易化させ操作性の向上を図った。

6. 評価実験

本稿で提案したファジーレーティングの効果についてプロトタイプを用いて評価実験を行った。本実験ではiモード用iアプリコンテンツ800件分のメタデータを作成し、検索条件として、データサイズ100KBくらい、コンテンツの料金100円以内という条件で検索を実施した。用意したコンテンツのうち80KB~120KBかつ100円以内であるコンテンツが100件あり、その中でユーザの主観評価によって得られた満足するコンテンツを30件を選択してもらった。なお、本実験はP2Pネットワークの特性には関連がないため、PC上で動作させた1つの検索ノードを用いて行い、満足するコンテンツの選定は1人の被験者が行った。検索結果の表示件数に対する再現率の評価結果を図15に示す。ここでは満足するコンテンツ30件のうち、検索結果として何件得られたかの割合を再現率と呼ぶこととする。また、評価の対象として、ランダムに順番をつけたもの(5回の平均値)、および1つの条件をもとに並べ替えたもの(データサイズを条

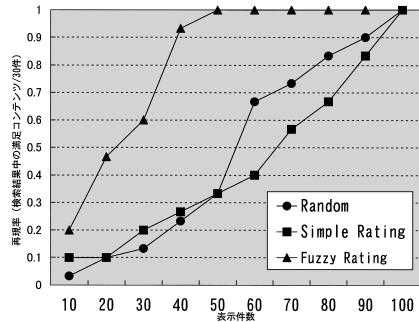


図 15 再現率
Fig.15 Recall.

件とし100KBに近いものから順に並べ替え)を用いた。それぞれ図15にrandom, simple ratingとして示す。ファジーレーティングのメンバシップ関数としては図9に示す関数を用いた。結果に示されるように、ファジーレーティングを行った場合には、検索結果を50件表示すると、満足するコンテンツ30件がすべてが表示されるが、他の方式では満足するコンテンツを30件表示するまでに、100件の結果を表示する必要があった。携帯電話のように表示能力に制限のあるモバイル端末であって、1度に10件程度しか結果を表示できない場合を想定すると、満足するコンテンツを5件得たい場合に、他の手法が3回検索を実施するのに比べ、ファジーレーティングでは1回の検索で済むことになる。ユーザの満足するレーティングを行うことで検索回数や検索結果取得に要する通信量を削減できるため、特にモバイル端末ではこのようなレーティング技術が重要であるといえる。

7. 考察

メタデータを用いた検索は、ユーザの意図を反映した精度の高い検索や、知的検索処理などの高度化が期待できるため、性能の制限があるモバイル端末からの検索に対して有効な手段を与えるものといえる。ただし、メタデータを利用した検索が普及するためには、技術的な課題、運用面での課題などまだまだ多くの課題がある。技術的な面では、本研究で検討を行ったメタデータ定義、メタデータ検索言語、レーティング方法の確立が特に重要であると考えられる。今回紹介したメタデータ定義では、所有者や内容などコンテンツの属性情報を判別可能とするように定義したが、さらに進んで属性値の記述内容を構造化してRDFやオントロジを適用することで、より知的な検索の実現が期待できる。また、今回設計した検索言語xRQLについては、実装による機能の実証も済み、検索言語への

要求条件として提示した機能を満たす実用的な検索言語を実現したといえよう。今後はオントロジ利用を想定したより高度な検索への対応も必要と考えられる。次に、今回提案したファジーレイティングでは、検索条件にユーザの表現したい曖昧さを含ませつつ、条件の適合度に応じて結果をレイティングするため、検索結果を限定しすぎず、かつ多すぎないよう適度に調整することが可能である。そのため、操作性や表示能力に制限のあるモバイル端末にとっては特に有効に働くと考えている。今回はファジー化する値として数値情報への対応しか考慮していないため、単語の意味の曖昧さについての定量化や、ファジーの度合いの設定方法は今後の課題である。このように、メタデータの利用技術の確立ならびに、モバイル端末からの入力、結果の表示の制限を考慮した効率の良い検索手法の実現に向けてはさらなる改善が必要である。

8. 関連研究

P2P ネットワーク上でメタデータを用いた検索を行う代表的なものとして Edutella があげられる¹³⁾。Edutella は SUN の提供する P2P プラットフォーム JXTA¹⁴⁾ を用い、複数の大学間で教育資源を交換するためのプロジェクトである。RDF 形式のメタデータを用いている。保持するコンテンツを分散環境で検索するものであるが、モバイル端末に対しての考慮はされていない。

RDF に関連する技術については現状ではまだ実用レベルに達していない。主な課題としては、メタデータ記述、意味および知識（オントロジ）の記述方式、実装などがあり、特に実装に関して現状では実用レベルの RDF のデータベースおよび推論エンジンの実装は存在しない。現在開発されている代表的な RDF リソースの検索言語としては RQL¹⁵⁾、RDQL¹⁶⁾ などがあるが、現状の検索言語は非 XML の形式を用いており、XML 形式と比較すると RDF で記述される複雑な形式を表現するうえで制限がある。また、プレーンなテキスト形式で RDF データのような階層的なモデルを記述するのは難しい。これらの検索言語は RDF データやスキーマに関する検索機能のみを定義しており、RDF のリソースデータに対する作成、削除、変更といった操作はサポートされていない。また、既存の RDF 検索言語は検索結果のフォーマット定義を行わず、このような検索結果のフォーマットはマシンによる自動処理が容易ではない。

9. まとめ

本稿では、セマンティック Web 技術を利用し、メタデータを用いた P2P ネットワーク上でのモバイル向けコンテンツ検索システムを提案した。本システムはコンテンツの内容を記述したメタデータを用いることで、モバイル端末に対して精度の高い検索サービスを提供するとともに、オーバレイ・ネットワークとして構成し、モバイル端末や PC などのように端末能力や利用するネットワーク種別の異なる様々な検索ノードが相互に連携してメタデータ検索を実現している。また、本システムの構成技術として、メタデータ記述、RDF 検索言語、ファジーレイティングについて検討を行った。さらに本研究では、検索システムの有効性評価を行うためにメタデータを用いた i モードコンテンツ検索システムのプロトタイプ実装を行った。今後はプロトタイプを用いて、検索性能やモバイルユーザの利便性について評価を行うとともに、さらなる改良を行う予定である。

参考文献

- 1) Gnutella. <http://www.gnutella.com/>
- 2) Google. <http://www.google.com/>
- 3) Yahoo!. <http://www.yahoo.com/>
- 4) W3C: Semantic Web Activity. <http://www.w3.org/>
- 5) Dublin Core Metadata Initiative. <http://dublincore.org>
- 6) Lassila, O. and Swick, R.R.: Resource Description Framework (RDF) Model and Syntax Specification, W3C Recommendation 22 (Feb. 1999).
- 7) i-mode. http://www.nttdocomo.co.jp/p_s/imode/
- 8) Brickley, D. and Guha, R.V.: Resource Description Framework (RDF) Schema Specification 1.0, W3C Candidate Recommendation 27 (Mar. 2000).
- 9) Alexaki, S. and Tolle, K.: The ICS-FORTH RDF Suite: Managing Volumes RDF Description Bases, WWW conference (2001).
- 10) Guha, R.V.: rdfsDB. <http://web1.guha.com/rdfsdb/>
- 11) 高萩栄一郎：ファジィ積分による多属性の特性値の総合評価法，SOFT 第 10 回ファジィシステムシンポジウム。
- 12) 加藤ほか：モバイル向け P2P ネットワークのアーキテクチャとプロトコルの提案，DICOMO (2002)。
- 13) Nejdil, W., et al.: EDUTELLA: A P2P Net-

working Infrastructure Based on RDF, WWW conference (2002).

- 14) Sun Microsystems: *Project JXTA*. <http://www.jxta.org/>
- 15) Karvounarakis, G., et al.: RQL: A Declarative Query Language for RDF, WWW conference (2002).
- 16) Seaborne, A.: RDQL — A Query Language for RDF. <http://www.w3.org/Submission/RDQL>

(平成 16 年 5 月 20 日受付)

(平成 16 年 11 月 1 日採録)



角野 宏光 (正会員)

1995 年名古屋大学工学部電子情報通信学部卒業。1997 年同大学大学院工学研究科修士課程修了。同年 NTT ドコモ入社。現在、NTT ドコモネットワークマネジメント開発部に所属。モバイルインターネットプロトコルおよびアプリケーションの研究開発に従事。電子情報通信学会会員。



石川 憲洋 (正会員)

1978 年京都大学工学部情報工学科卒業。1980 年同大学大学院工学研究科情報工学専攻修士課程修了。同年日本電信電話公社 (現 NTT) 入社。現在、NTT ドコモネットワークマネジメント開発部に所属。モバイルインターネットの研究開発と国際標準化に従事。博士 (情報学)。電子情報通信学会会員。



小俣 栄治

2002 年武蔵工業大学工学部電子情報工学科卒業。同年 NTT ドコモ入社。現在、NTT ドコモネットワークマネジメント開発部に所属。モバイルインターネットプロトコルおよびアプリケーションの研究開発に従事。



ヨハン イェルム

エリクソン・リサーチにおけるシニア・スペシャリスト。スウェーデン陸軍士官退役の後、ジャーナリストとして 12 年間従事する。その間、雑誌の編集長としてインターネットに深く関わり、EU の On the Move プロジェクト等のコーディネータを務める。また、スウェーデンにおける最初のインターネットに関する本を出版し、スウェーデンのインターネットに初期の頃から深く関わる。エリクソンに入社後は、マサチューセッツ工科大学の客員研究員として在籍し、CC/PP 部会等の議長を務めて W3C フェローとなる。現在は Open Mobile Alliance のアーキテクチャー・グループの副議長も務める。



宮津 和弘

1990 年東京工業大学工学部電子物理工学科卒業。1992 年イリノイ大学大学院修士課程修了 (M.S.E.E.)。日本モトローラ株式会社入社後は、デジタル信号処理および移動体通信の研究に従事。1997 年日本エリクソン株式会社入社し、モバイル端末企画に携わる。1998 年以降は、日本における Bluetooth 技術および製品の責任者を務める。著書としては『Bluetooth 技術解説ガイド』(リックテレコム) 等がある。現在、エリクソン・リサーチ・ジャパンにおいて、モバイル・ユビキタスに関する研究に従事。



村上 慎吾 (正会員)

1998 年筑波大学第 3 学群情報学類卒業。2000 年同大学大学院工学研究科修士取得後退学。同年日本エリクソン株式会社入社。以来、エリクソン・リサーチ・ジャパンにおいて、モバイルインターネットやアプリケーションに関する研究に従事。