

任意のデジタルコンテンツに対するアノテーション プラットフォーム

梶 克彦^{†1}

長尾 確^{†2}

^{†1} 名古屋大学 大学院 情報科学研究科

^{†2} 名古屋大学 情報メディア教育センター

E-mail: kaji@nagao.nuie.nagoya-u.ac.jp

あらまし 現在のアノテーションに関する研究のほとんどがコンテンツの種類に依存したものであるため、複数の種類のコンテンツを同時に扱った応用を実現することが困難である。そこで我々は、コンテンツの種類に依存せず容易にアノテーションを生成・利用できるアノテーションプラットフォーム Annphony (アンフォニー) を構築している。本プラットフォームでは、コンテンツの構造と、詳細部分に対する個人の解釈・嗜好に関するアノテーションを扱うために、任意のコンテンツの文節化を実現する ElementPointer を提案し、そのプロセッサを実装する。また RDF を一部拡張することにより柔軟なリレーション記述を可能にし、アノテーション定義の共有・利用を実現する。さらにアノテーションやその定義を扱うための機能について述べる。

An Annotation Platform for Any Kinds of Digital Contents

Katsuhiko Kaji^{†1}

Katashi Nagao^{†2}

^{†1} Graduate School of Information Science, Nagoya University

^{†2} Center for Information Media Studies, Nagoya University

Abstract There are several researches on annotation, but most of them have strong restrictions on types of contents. On the ground of difference of annotation format, we can not deal with different types of contents and their annotations simultaneously. Therefore we have been developing an integrated annotation platform that is applicable to any kinds of digital contents. First, we propose a format of annotation data and its schema to deal with content structures and individual interpretations. Next, we describe several functions of the platform to create/share/retrieve annotations and their schemas.

1 はじめに

Web 上に存在する大量のデジタルコンテンツを有効利用するため、テキストや音楽などのコンテンツに対してメタ情報 (アノテーション) を付与し利用する研究 [6][16] が盛んに行われている。これらの研究から、コンテンツのセマンティクスを考慮し、それぞれのユーザにあったコンテンツ処理を行うことの重要性と、そのためにはコンテンツの構造に加えコンテンツの詳細部分にまで踏み込んだ個人の解釈・嗜好情報が必要であることがわかる。

しかし、現在までに提案されているデジタルコンテン

ツに対するメタデータの表現形式の多くは、コンテンツの構造や多様の解釈を記述することができなかつたり、コンテンツの種類に依存しているため互換性に乏しい。そのため異種コンテンツを同時に利用し、セマンティクスと個人性を考慮した応用研究を進めるにあたっての障害となっている。

また我々は、アノテーションが様々なサービス間で共有されることで、それらを組み合わせた新しい応用を実現できると考えている。アノテーション共有によって実現される応用は、基となったサービスにとってもアクセスやユーザの増加を見込むことができるため、アノテー

ションを広く共有することを支援する必要性が生まれるろう。

本稿では、まず任意のコンテンツに対するアノテーションを実現するために必要な要素について述べ、次に我々が構築しているアノテーションプラットフォーム Annphony(アンフォニー)におけるアノテーションとそのスキーマの形式について述べる。さらに Annphony においてアノテーションやスキーマの利用・共有を支援するための機能について述べる。

2 任意のコンテンツへのアノテーションを実現するために

現在デジタルコンテンツに対するアノテーションの記述形式がいくつか提案されている。例えばテキストに対する構文・意味情報を詳細に記述するための GDA(Global Document Annotation)[4] や、テキストコーパスに対するアノテーション形式である CES(Corpus Encoding Standard)[8] が提案されている。一方 Flickr[1] では、独自のアノテーション形式により画像の矩形部分に対するコメント付与を実現している。

これらの形式はコンテンツの種類に閉じており、例えば Flickr において記述されたコメントに対して GDA による詳細な言語構造を記述といった、あるコンテンツのアノテーションを異種コンテンツに適用することが困難である。このように、現在提案されているコンテンツの内部構造にまで踏み込んだアノテーション形式には、互換性が乏しいといえる。

一方で、コンテンツ全般に対するアノテーションの形式としてリソース間の関係をグラフ構造で記述する RDF(Resource Description Framework)[12] が提案されている。RDF では URI(Uniform Resource Identifier)[3] を持つ対象をリソースと呼び、そのリソースに対するメタ情報を主語・述語・目的語の組で表現する。つまり、どのようなリソースであってもそれを URI で表現することができれば、そのリソースに対するアノテーションを記述することが可能である。

コンテンツの構造情報を扱うためには、そのコンテンツの詳細な部分へのアノテーションを実現しなければならない。コンテンツの任意の箇所を指し示す方式としては、XML の特定のノードを指す XPointer[13] や、ビデオ・オーディオなどの連続メディアの任意の箇所を指し示す URI time interval specification [7] など、いくつかの形式が提案されているが、全てのメディアを網羅できておらず、また例えば「ある楽曲の、特定の時間区分に含まれる特定の楽器パート」といった、提案されている形式では指し示すことのできない例が多く存在する。

そのため、任意のコンテンツの、任意の箇所を指し示すことのできる柔軟な表現形式が必要である。

音楽や絵画などの芸術作品には、鑑賞者によって様々な解釈が存在する場合が多い。また楽曲の構造化を行う GTTM(Generative Theory of Tonal Music)[9] では、楽曲の構造を一意に決定することができない場合がある。そのため複解釈に基づく構造が存在する可能性がある。このような解釈の情報を扱うためには、そのアノテーションをさらに詳しく説明したり補足する必要がある。しかし RDF や GDA では、アノテーションに対してアノテーションを関連付ける柔軟なリレーションを記述することができない。

また構造や解釈といった様々なアノテーションを定義する際には、既存の定義を利用・拡張することで、アノテーションの再利用性が高まると考えられる。例えば異なるコンテンツ間で共通のプロパティを定義し、各コンテンツのアノテーションでそのプロパティを利用することで、コンテンツの種類にこだわらずそのプロパティを利用することができるだろう。そのためアノテーションのスキーマの存在が必須となる。

Flickr ではアノテーションを利用するための Flickr API が用意されており、また RDF に関しては Protégé[2] や RDF Gravity[11] など、RDF の記述・検索・可視化を行うアプリケーションが提供されている。様々な種類のアノテーションを幅広く収集し、利用できるようにするためには、これらのようにアノテーションを容易に扱うことができる環境を提供する必要がある。

3 Annphony

我々は任意のコンテンツに対するアノテーションを扱うアノテーションプラットフォーム Annphony(アンフォニー)を構築している。本章では、本プラットフォームで扱うデータの形式と、プラットフォームとしての機能について述べる。

3.1 アノテーションとスキーマの形式

前章で述べたように、任意のデジタルコンテンツの構造や個人の解釈に関するアノテーションを記述するためには、1. 任意のコンテンツのセグメントを指し示し、2. 指し示したセグメントに対し柔軟なリレーションを記述できなければならない。また、3. 既存のアノテーション定義を効率的に利用し、柔軟に新たな定義を記述できる必要がある。Annphony ではこの3点を以下に述べる手法で解決する。

3.1.1 コンテンツのセグメンテーション

任意のコンテンツの詳細に対するセグメンテーションを行うための表現形式として ElementPointer を提案する。ElementPointer は以下の形式をとる。

```
[Content URI]#epointer([Schema URI]([arg1,arg2..]))
```

コンテンツの URI に続き、#epointer 以降にコンテンツのどの部分を指し示すかを記述する。スキーマ URI 以降に続く引数の順序とそれぞれが表す意味は、RDF[12] のスキーマ記述言語である RDFS(RDF Schema)[15] により記述される。ElementPointer のスキーマについては 3.1.3 節において詳しく述べる。以下に ElementPointer の具体例を示す。

```
http://domain1/picture.jpg#epointer  
(http://domain2/dim.rdfs#dim(10px,20px,30px,40px))
```

この例では、絵画・地図など、2次元で表されるメディアに対する ElementPointer をスキーマ URI において 4つの引数 (X・Y 座標、幅、高さ) により定義し、そのスキーマを用いて具体的な部分を表している。実際には URI としての妥当性を保つため、スキーマの URI とそれに続く引数の部分を URL エンコーディングする。計算機は ElementPointer の URI と、その中に記述されているスキーマの URI を参照することで、どのコンテンツのどの部分を指し示しているかを解釈する。

また ElementPointer では、同一のコンテンツ内に限り、複数のセグメントをグループ化することを許している。下記のように List type にグループの種類を指定することで、複数のセグメントの集合を表す。

```
[Content URI]#epointer([List type]([Schema URI]([arg1,arg2...]),([arg1,arg2...])))
```

List type には、RDF においてリストを表現する seq(順序付リスト)、bag(順序無しリスト)、alt (代替表現リスト) のいずれかを記述する。本形式により、代表的なコンテンツの構造化手法であるグルーピングを実現する。

新たに ElementPointer を定義する場合、ElementPointer によって示されたコンテンツの部分を扱うための ElementPointer プロセッサを同時に作成する。さらに後述する ElementPointer スキーマへのアノテーションとして関連付けることで、本プラットフォームにおいて該当するプロセッサを利用することができる。

ElementPointer プロセッサは、特定のインタフェースに基づいてメソッドを実装することによって作成される。具体的には、指定された ElementPointer の存在確認や、実際に指定された部分を取得するメソッド、また

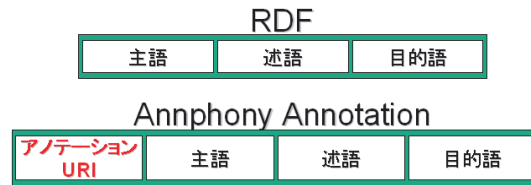


図 1: RDF と Annphony Annotation の違い

```
- <imp:impression rdf:about="http://domain/annotation?key=32">  
- <an:target>  
- <rdf:Seq>  
  <rdf:li rdf:resource="http://domain/content1.obj" />  
  <rdf:li rdf:resource="http://domain/content2.obj" />  
</rdf:Seq>  
</an:target>  
<imp:desc_impimpression>2つ一緒に楽しむととても心地よいです。  
</imp:desc_impimpression>  
<an:date>2005-12-24</an:date>  
<an:time>12:31:11</an:time>  
<an:reliability>0.5714285714285714</an:reliability>  
<an:annotator rdf:resource="http://domain/kaji.xml" />  
</imp:impression>
```

図 2: Annphony におけるアノテーションの例

二つの ElementPointer 間の論理和、論理積、差にあたる部分の取得など、ElementPointer の演算を行うためのメソッドを実装する。

3.1.2 柔軟なリレーション記述

本プラットフォームで扱うアノテーションは ElementPointer で規定したセグメント間の関係や、複数人の解釈を多数のコンテンツに対して記述するため、コンテンツやアノテーション同士の関係は複雑なグラフ構造になると予想される。そこで、アノテーションの基本形式にリソースの関係を有向グラフで表現できる RDF を採用した。しかし、RDF の形式ではそれぞれの解釈に識別子がないため、その解釈に対する解釈を記述することができない。

そこで本プラットフォームでは RDF を一部拡張してアノテーションを記述する。図 1 に示すように、RDF では主語・述語・目的語の Triple で記述するが、Annphony Annotation はそれに加え、そのアノテーションの識別子であるアノテーション URI を持つ Quad で表現する。

図 2 は 2つのリソースに対する印象アノテーションの例である。アノテーションの指し先を表す <an:target> タグと、RDF におけるコレクション表現のタグである <rdf:Seq>、<rdf:Bag>、<rdf:Alt> のいずれかにより複数のリソースに対するアノテーションを記述する。また、アノテーション自体に Annphony が発行する URI を持たせることで、同一の対象に対する複数人の解釈情報の記述を可能にしている。

ElementPointer は、特定のスキーマにおいて定義され

```

- <ep:dim rdf:about="http://domain1/picture.jpg#epointer
  (http://domain2/dim.rdfs#dim(10px,20px,30px,40px))">
  <an:target rdf:resource="http://domain1/picture.jpg" />
  <ep:x>10px</ep:x>
  <ep:y>20px</ep:y>
  <ep:width>30px</ep:width>
  <ep:height>40px</ep:height>
</ep:dim>

```

図 3: アノテーションとして表現される ElementPointer

```

- <rdfs:Class rdf:ID="impression">
  <rdfs:label>印象アノテーション</rdfs:label>
  <rdfs:subClassOf rdf:resource="http://www.nagao.nuie.nagoya-
    u.ac.jp/annphony/annotation.rdfs#annotation" />
</rdfs:Class>
- <rdf:Property rdf:ID="desc_impression">
  <rdfs:label>印象情報(自由記述)</rdfs:label>
  <rdfs:domain rdf:resource="#impression" />
  <rdfs:range
    rdf:resource="http://www.w3.org/2001/XMLSchema#string" />
</rdf:Property>

```

図 4: Annphony におけるアノテーション定義の例

た複数のプロパティから構成されるため、アノテーションと同様の構造を持っている。また、ElementPointer をアノテータが手入力することは困難である。そこで、本プラットフォームでは ElementPointer をアノテーションと同様の形式で扱い、URI の生成を支援する。図 3 は、前節の ElementPointer の例がアノテーションに変換されたものである。通常のアノテーションと同様に、スキーマを指定し、各プロパティを入力することで自動的に ElementPointer の URI が生成される。

3.1.3 アノテーションスキーマ

本プラットフォームでは RDFS によるアノテーション定義をサポートする。日付、後述するアノテータ URI や信頼度など基本的なアノテーションを定義する基本アノテーションをルートとし、既存のアノテーションを継承して新しいアノテーションを定義する。アノテーション定義には、そのアノテーションの説明や、利用可能なプロパティとそのデータ型が記述される。図 4 では、印象情報を文字列型 (string) で記述するためのアノテーションを定義している。基本アノテーションを継承しているため、そこで定義されている日付やアノテータ URI などのプロパティも同時に記述できる。

データ型は XML Schema[14] で定義されている integer や string などのプリミティブ型に加えて、XML Schema において正規表現や列挙による型、単位付きの型などを記述する SimpleType により定義された型を利用することができる。図 5 では、数字の列の最後に "px" という単位が付くデータ型を定義している。

新しい定義やデータ型は本プラットフォームに登録することで利用可能になる。アノテーション定義は容易に一覧表示・検索ができ、任意のユーザはそれらを継承し

```

- <xsd:simpleType name="pxType">
  - <xsd:restriction base="xsd:string">
    <xsd:pattern value="\d+px" />
  </xsd:restriction>
</xsd:simpleType>

```

図 5: データ型定義の例

た新しいアノテーションを定義することができる。アノテーションを継承して定義することで、異種コンテンツの間でアノテーションのプロパティを共有でき、効率のよいアノテーションの活用を実現できる。

前節で述べた ElementPointer の定義も、アノテーションの定義と同様に RDFS で記述されるが、それに加えそれぞれのプロパティについて、その出現順序を明示する。これは ElementPointer の URI で記述される引数の順序を確定するためである。

コンテンツの詳細箇所を指し示す手段は、XPointer や URI time interval specification の他にも存在するが、幅広い分野のユーザが必ずしもそれらを熟知しているわけではない。ElementPointer の定義の形式をアノテーション定義と同様の形式にすることで、コンテンツの詳細箇所を指し示す手段についても容易に検索し、利用・新規定義をすることができる。

3.2 Annphony プラットフォームの機能

幅広くアノテーションを共有・利用するためには、アノテーションを利用する環境を整備しなければならない。本プラットフォームでは、アノテーションやその定義を容易に共有・検索・作成・利用する機能、またコンテンツの構造・解釈を扱うために必要となる機能を提供する。さらに、アノテーションが潜在的に抱える問題に対処するための機能を備えている。以下に本プラットフォームの主要な機能について述べる。

3.2.1 アノテーション定義へのアノテーション

アノテーションの定義が登録される際、定義の作成者はその定義へのアノテーションを同時に記述する。具体的には定義者情報、自然言語による利用例などの説明、「Jazz」、「ニュース」、「判例」など複数のタグによる適用可能なコンテンツの列挙を行う。

どのような場面においてその定義が利用可能であるかという情報は、定義の作成者本人が全て網羅することは困難である場合が多い。そこで本プラットフォームでは、定義者以外の複数のユーザがそのスキーマに対して利用例やタグなどのアノテーションを行うことを許している。また ElementPointer の定義へのアノテーションの場合、上記のアノテーションに加え 3.1.1 節で述べた

ElementPointer プロセッサを指定する。

また本プラットフォームはそれらのアノテーションに基づくアノテーション定義の検索機能を備えている。アノテーション定義を発見したいユーザは、キーワードでの検索や、適用するコンテンツのタグ情報から絞り込むことで容易に目的の定義を発見し、利用・拡張定義することができる。

3.2.2 アノテーション定義に基づくアノテーションの検索

幅広いコンテンツへのアノテーションを扱う環境では、動的にアノテーションの定義が増加することが予想されるが、アノテーションの定義が追加される度に、その定義に基づくアノテーションを利用するようアプリケーションを対応させることは非常に高コストになる。

本プラットフォームでは、アノテーション定義の継承関係を考慮し、着目した定義の継承元や継承先のアノテーションについても検索対象とすることができる。また、ある特定のプロパティに着目し、そのプロパティが利用されている定義に基づくアノテーション全てを検索対象にするなど、アプリケーションが、動的に追加されたアノテーション定義に基づくアノテーションを利用できる検索機能を備えている。

3.2.3 コンテンツの構造を扱うために

様々なコンテンツの構造化に関する研究 [4][9] が行われているが、それらに多く見られる手法として、コンテンツのグループ化、または階層化が挙げられる。本プラットフォームでは ElementPointer の採用により、コンテンツの任意の箇所のグループ化に対応している。また、コンテンツやアノテーションを一つのノードとみなし、それらの関係をツリー形式で記述し、そのツリーを利用する機能を備える。さらに、アノテーションの基本構造として RDF を採用しているため、他のコンテンツとの間の関係をグラフ構造で記述することもできる。

3.2.4 解釈を扱うために

アノデータに関する情報は、コンテンツの変換や推薦など、個人の嗜好や解釈を踏まえた応用を実現する際に必須となる。本プラットフォームではそれぞれのアノデータに URI を持たせ、そのプロファイルのアノテーションとして管理する機能を提供する。

また、あるコンテンツに対し複数の解釈が存在する場合、それぞれの解釈が正確であるか、または信頼できるものであるかを判断する必要がある。そこで各アノテーションとアノデータに「信頼度」プロパティを付与

する。信頼度の算出に関しては、文献 [17] において定義された手法を採用した。

アプリケーションはアノテーションの信頼度に基づき、複数のアノテーションの中から最も信頼性の高いアノテーションを採用したり、信頼度が閾値以上である全てのアノテーションを採用するなど、複数の解釈の扱いを決定することができる。

3.2.5 アノテーションが持つ問題点への対処

MP3 形式の音楽ファイルにはアーティストやタイトルなどのメタデータをそのファイル自身に埋め込むことができ、Exif(Exchangeable Image File Format) 形式の画像には、撮影日時やその画像についての情報を埋め込むことができる。このようなコンテンツの内部にアノテーションを埋め込む方式の場合、コンテンツへのアノテーションの付与や利用のためにはコンテンツとアノテーションを同時に管理しなければならない。しかし様々な種類のコンテンツを対象としたアノテーションを扱う場合、必ずしもコンテンツとアノテーションを同時に管理できるとは限らない。そのため本プラットフォームではコンテンツとアノテーションを切り離れた管理方法を採用している。

コンテンツとそのアノテーションが分離されている場合、コンテンツの改変・移動・削除によって、そのコンテンツに関連付けられたアノテーションが Orphan Annotation(指し先の無いアノテーション) や Misleading Annotation(誤解を生むアノテーション) になってしまう危険性が指摘されている [5]。そこで本プラットフォームではこれらの問題の発生を抑えるために以下の機能を提供する。

Orphan Annotation への対処のため、指し先が存在しないアノテーションを登録することを許可しない。データベースへの登録時に、アノテーションの指し先のリソースへアクセスを試み、存在を確認した後に登録される。指し先が ElementPointer によって記述された URI の場合は、前述の ElementPointer プロセッサによって該当するコンテンツのセグメントの存在を確認する。

Misleading Annotation は、コンテンツやアノテーションが編集されることで現れ、そのリソースを指し示しているアノテーションが意味をなさなくなったり、別の意味になってしまうものである。本プラットフォームでは、アノテーションが編集・再登録された場合、そのアノテーションに関連するアノテーションを取得し、それらの確認・編集を管理者に促す。

また一定時間ごとに登録されたアノテーションをクローリングしてコンテンツの編集の有無を確認し、編集

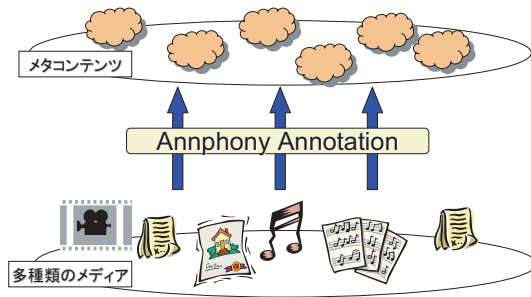


図 6: コンテンツからメタコンテンツへ

や削除が確認されたコンテンツに関連するアノテーションに関して同様にそれらの確認・編集を促す。

4 Annphony がもたらす効果

4.1 メタコンテンツとしての処理

様々な種類のコンテンツに対して付与された、統一した形式による内部構造及び個人の解釈情報を利用することで、図 6 のようにそれぞれのコンテンツをメディアの種類によらないメタコンテンツとして捉えることが可能になる。今後はメタコンテンツの処理に関する研究が盛んに行われることが期待される。

具体的な応用例としては、例えばビデオのテロップに対するアノテーションに、テキスト処理のために定義されたアノテーション定義を利用するなど、異種コンテンツ間でアノテーションを共有し、互いに利用したり、音楽プレイリストを拡張し、音楽に限らない複合メディアのプレイリストを作成・配信するといった、従来困難であった複数のコンテンツのセマンティクスと個人嗜好を考慮した統合メディア処理が実現されるだろう。

4.2 オントロジー構築へのアプローチ

アノテーションは、人間がコンテンツに対し詳しい意味記述を行うものである。そのため大量のアノテーションを用いてその解析を行うことにより、リソースの分類体系やその関係、推論のためのルールなどを定義するオントロジーの構築を実現できる可能性を秘めている。

トピックマップ [10] ではリソース全体についてのオントロジー構築を目指しているが、コンテンツの任意の箇所を分節化する ElementPointer は、より詳細なオントロジーの構築に役立つと期待できる。また図 6 に示されるように、アノテーションがコンテンツの種類に依存しない共通の形式で記述されるため、Web における様々なコンテンツにまたがる意味関係を記述されたアノテーションが増加することになるだろう。そのため大量のアノテーションを利用して、リソース全体に関するオントロジーを構築することが期待される。

5 おわりに

本稿では、任意のデジタルコンテンツに対しコンテンツの構造と個人の解釈に関するアノテーションを作成・利用することのできるアノテーションプラットフォーム Annphony について、アノテーションとそのスキーマの形式と、それらを管理するための機能に関して述べた。

今後はネットワーク上に Annphony が複数存在する分散環境における協調機能やアノテーションのアクセス・編集ポリシーについて検討すると共に、実際にアノテーションを付与した異種コンテンツによる応用を実現していく予定である。

文 献

- [1] Flickr. <http://www.flickr.com/>, 2005.
- [2] Protégé. <http://protege.stanford.edu/>, 2005.
- [3] Berners-Lee, T., Fielding, R. and L. Masinter. Uniform Resource Identifiers (URI): Generic Syntax. *RFC 2396*, 1998.
- [4] Hashida, K. Global Document Annotation. <http://i-content.org/GDA/>, 2002.
- [5] Ralph R. S. Marja-Riitta K., Eric P. Annotea: An Open RDF Infrastructure for Shared Web Annotations. *Proc. Tenth International World Wide Web Conference*, 2001.
- [6] Katashi Nagao. *Digital content annotation and transcoding*. Artech House Publishers, London, 2003.
- [7] Pfeiffer, S., Parker, C. and A. Pang. Specifying time intervals in URI queries and fragments of time-based Web resources. http://www.annotex.net/TR/URI_fragments.txt, 2003.
- [8] MULTTEXT Project. Corpus Encoding Standard. <http://www.cs.vassar.edu/CES/>, 2000.
- [9] Fred Lerdahl Ray Jackendoff. *A Generative Theory of Tonal Music*. MIT Press, 1996.
- [10] SC34/WG3. Topic Maps . *ISO/IEC 13240*, 2002.
- [11] Rupert W. Sunil G. RDF Gravity (RDF Graph Visualization Tool). <http://semweb.salzburgresearch.at/apps/rdf-gravity/index.html>, 2004.
- [12] W3C. Resource Description Framework. <http://www.w3.org/RDF/>, 1999.
- [13] W3C. XML Pointer Language. <http://www.w3.org/TR/WD-xptr>, 2001.
- [14] W3C. XML Schema. <http://www.w3.org/XML/Schema>, 2001.
- [15] W3C. RDF Schema. <http://www.w3.org/TR/rdf-schema/>, 2004.
- [16] 梶克彦, 平田圭二, 長尾確. コミュニケーションメディアとしてのプレイリストを目指して. *情報科学技術レターズ vol.4*, pp. 115–118, 2005.
- [17] 山本大介, 長尾 確. 閲覧者によるオンラインビデオコンテンツへのアノテーションとその応用. *人工知能学会論文誌*, Vol.20, No.1, pp. 67–75, 2005.