

漫画表現用知的トランスコーダを用いた 人間関係ネットワークのブラウジング環境

坂本 竜基[†] 田中 郁^{††} 小暮 潔[†]

本稿では、FOAF によって示された人間関係ネットワークを表やグラフに比べて柔軟な表現力を持つ漫画表現を利用したインターフェースによってブラウジングするシステムを提案する。FOAF は人物プロフィールを RDF/XML 形式で公開するためのフォーマットであり、これをブラウジングするためのツールが数多く提案されているが、これらは FOAF が持つ複雑な語彙を表組で提示しているにすぎない。漫画による FOAF の表現は、既存ツールでは表現しきれない多様な表現を可能にする利点がある一方、漫画表現がハイパーリンクをたどるような作業形態を支援するインターフェースとして機能することはまだ確認されていない。本稿では、個人の FOAF を漫画へと変換するトランスコーダおよび複数の漫画間をハイパーリンク接続するシステムを説明し、評価実験を通じて漫画表現されたインターフェースを持つ本システムが人間関係ネットワークのブラウザとして機能することを示す。

A Social Network Browsing System through Intelligent Transcoder Representing FOAF in Comic Strip

RYUUKI SAKAMOTO,[†] KAORU TANAKA^{††} and KIYOSHI KOGURE[†]

This paper proposes a browsing system for the social network composed in the FOAF framework. The system utilizes comic strip expression which has more superior capacity of representation than diagram with the table of HTML. FOAF is a format to be proposed to exhibit a person's profile in RDF/XML form as a part of the Semantic Web project. Many tools have been developed for browsing the network, however they display people profiles using abstruse vocabulary of FOAF into simple tables. On the contrary, the proposed system displays these structures as comic strip expressions. This system has confirmed experimentally to work as the browser of human social networks.

1. はじめに

セマンティック Web¹⁾ の提唱に呼応して、Web コンテンツに対するメタデータを配布する機運が高まっている。ここでのメタデータとは Web における各種リソースの意味や構造を記述した二次情報であり、Web との親和性を考慮して RDF/XML 形式²⁾ で記述されることが多い。一般的にいうとメタデータとは計算機ネットワーク上のリソースに対して付与されるものであるのに対し、ネットワーク上の直接的なリソースではない「人物」に対してメタデータを付与する FOAF³⁾ と呼ばれる枠組みが提案されている。人物に対するメタデータとは、その人物が持つ総合的のプロ

フィール情報のことである。FOAF では、この情報を RDF/XML 形式を用いて表現する。FOAF におけるプロフィールは、名前、所属等の個人に帰属する情報（以下、個人プロフィールと呼ぶ）と、ある人物がどの人物を知っているかという人間関係の情報（以下、関係プロフィールと呼ぶ）の 2 種類に大別される。このうち関係プロフィールは知人の個人プロフィールや FOAF ファイルを指示することによって表現され、この指示関係による人間関係のネットワークを想定することができる。このような FOAF ファイル群によって形成される人物関係ネットワークは、ネットワークを通じた友人関係の拡張、趣味や共通の話題の人物を集めたオンラインコミュニティ⁴⁾ やソーシャルネットワークワーキングサービス⁵⁾ (SNS)、KnowWho 検索⁶⁾ の基盤データ形式としての活用が期待される。

FOAF が SNS 等の基盤データとして利用された場合、そのデータを扱うのは計算機である。FOAF は他のサービス固有のデータ等と組み合わせられ、人間は加

[†] ATR 知識科学研究所

ATR Knowledge Science Laboratories

^{††} 北陸先端科学技術大学院大学知識科学研究科

School of Knowledge Science, Japan Advanced Institute of Science and Technology

工された形態に間接的に接することになる。一方で、FOAF ファイルは単体でも個人および関係プロフィールを表現しているため、これを人間が直接閲覧する利用形態も考えられる。この場合、関係プロフィールを中心に人間関係ネットワークを巨視的に把握する形態と1人1人の個人プロフィールを順々に閲覧していく微視的な形態が考えられる。このうち、後者は Web ページを1つ1つ、ハイパーリンクをたどりながら拾い読みをする Web ブラウジングと同様に、個人プロフィールや関係プロフィールを逐次的に閲覧していく作業形態である。以下では、この行為を人間関係ネットワークのブラウジングと呼び、これを支援するツールを人間関係ネットワークのブラウザと呼ぶ。

既存の人間関係ネットワークのブラウジング用ツールは、単にプロフィール情報のテーブル表現(以下、表組と呼ぶ)が多く人物プロフィールと関係プロフィールを人間にとって直感的に把握可能な形式として提示しているとはいえない。たとえば、FOAF の人物プロフィール部分を XSLT⁷⁾ を用いて HTML のテーブルに変換すると語彙が項目名となって表示されることになるが、元来機械可読を目標とした RDF/XML の語彙は人間にとって可読性が低いことは明らかである。また、表組は形式が一定であるため、多くの異なる要素に対して内容に応じて柔軟に表現を変えるといった表現能力に乏しい⁸⁾。

本稿では、機械可読である FOAF ファイルを漫画表現を用いて人間にとって直感的な表現にコンテンツ変換する知的トランスコーダと、この表現を用いた人間関係ネットワークのブラウザを提案する。ここでの漫画表現とは、単に絵と文字が組み合わせられた「コマ」の羅列ではなく、コマの連なりがストーリーとして矛盾なく成立するものを指す。以下、提案ブラウザを漫画 FOAF ビューワと呼ぶ。

漫画表現は絵と文字を自由に配置、組み合わせる物語の表現を行うことが可能なため、静的な絵や文字の項目を割り当てる表組やグラフに比べ FOAF が内包する多様な語彙とそれらの関係の表現に対して柔軟に対応することが可能である。しかし、漫画表現は、線条性や現示性を駆使して物語を語る手法であり個人プロフィールは表現可能であるものの、関係プロフィールの表現やブラウザとして個々の FOAF ファイルを逐次的に読み込み、提示する手法については明らかに

されていない。

これに対し本稿では、ある漫画において登場する関係プロフィールが指示する人物の FOAF ファイルをユーザの要求に応じて読み込み、その人物が中心人物となる漫画へと切り替えることによってブラウジングしていく手法を示す。この切替えの指示は知的トランスコーダが漫画のキャラクターにリンクアンカを設定することで実現する。このような漫画とリンクを組み合わせたブラウザが FOAF によって形成される人間関係ネットワークのブラウザとして成立する否かを学術会議における運用実験を通して検証する。

以下、2章において FOAF の概要と人間関係ネットワークについて説明し、3章で提案する漫画表現を用いたブラウジング環境の基本アイデアを述べる。次に、4章で既存のツールや先行研究の説明を通して漫画 FOAF ビューワの新規性を明らかにし、5章において漫画 FOAF ビューワの仕組みを説明した後、6章で漫画 FOAF ビューワを実際に学術会議における人間関係ネットワークを示した FOAF に適用した運用例を示す。また、この運用例において得られたデータとアンケート結果も分析、考察する。最後に7章で結論を述べる。

2. FOAF による人間関係ネットワークとブラウジング

本章では、図1を基に本研究が対象とする FOAF および FOAF が構成する人間関係ネットワークについて概説する。FOAF は、foaf:Person という人物を表すクラスに個人プロフィールとして氏名、メールアドレス、所属等を記述する。一般的に、RDF/XML ではクラスを一意に識別するためにリソースの URI を主語として与えるが、foaf:Person は計算機ネットワークの構成要素ではない人間に対するメタデータであるため主体をリソース URI として記述できない。よって、

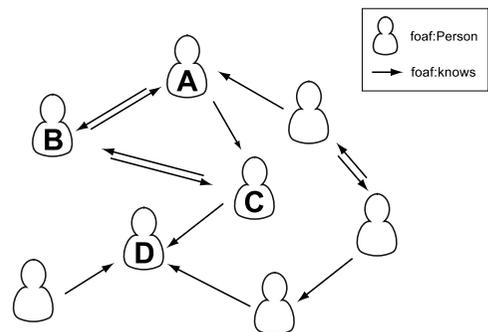


図1 FOAF による人間関係ネットワークの概念図
Fig.1 An image of social network on FOAF.

線条性とは、文章表現に代表されるように表現の部分部分を集積することによってその関係を一連の流れとして把握させる性質を指す。また、現示性とは、写真や絵画のように全体を一望で把握できる表現上の性質を指す⁹⁾。

FOAF では、リソース URI の代替として、foaf:mbox が foaf:mbox_sha1sum の値として与えられるメールアドレスを識別子として代替する案が提唱されている。

一方で、FOAF には関係プロフィールのうち知人関係を記述するための foaf:knows という語彙が用意されている。これは、foaf:Person で示される 2 人の人物メタデータを「知人である」というプロパティで関連付けるものである。ここで、ある特定の人物の FOAF メタデータの中に foaf:knows の子要素として foaf:Person があれば、この FOAF が示す人物と foaf:Person で示される人物の間に知人関係があることが示される。foaf:knows の子要素の foaf:Person のプロパティには、その foaf:Person が示す人物が所有する FOAF ファイルへのポインタを与えることが可能であり、このポインタによって複数の FOAF ファイルをつなぐ人間関係ネットワークが形成される。

3. 漫画表現による人間関係ネットワークのブラウジング環境

本章では、FOAF で構成された人間関係ネットワークをブラウジングするための漫画 FOAF ビューワの動作の概略を図 2 の例で説明する。図 2 は図 1 の人間関係ネットワークが複数の漫画上でどのように表現されるのかを示すイメージ図である。漫画 FOAF ビューワは人物 A の FOAF ファイルを読み込むと、FOAF のコンテンツ変換を行い、図 2 左に示す、人物 A が主人公の漫画を生成する。すなわち、人物 A の foaf:Person における人物プロフィールに基づいた漫画表現をするとともに、関係プロフィールである foaf:knows が指し示す人物 B と C をそれらの foaf:Person の情報に基づき脇役として登場させる。加えて、脇役に関しては、それぞれがルートの foaf:Person である FOAF ファイルへのハイパーリンクのアンカを設定する。

上述のアンカにより、ユーザは漫画からこの主人公 A の持つ個人プロフィールと関係プロフィールを理解したうえで、どの脇役のリンクをたどるかを決定することができる。ここで、ユーザが人物 B のキャラクターのリンクアンカをマウスクリックすると、漫画 FOAF ビューワは選択された脇役 B が所有する FOAF ファイルを読み込み、現在表示されている漫画から、人物 B を主人公とする漫画に切り替える（図 2 中央）。この漫画では、人物 A と人物 C が脇役として登場しており、ユーザは同じく漫画表現されたプロフィールと関係する人物を把握し、次に閲覧したいキャラクターを選別する。この操作を繰り返すことによって、ユーザは人間関係ネットワークの各ノードにおいて漫画表現

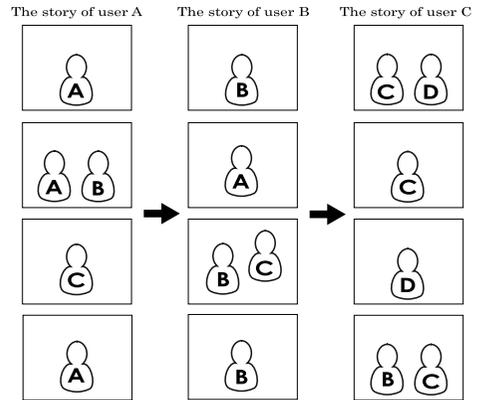


図 2 漫画による人間関係ネットワークをブラウジングするイメージ図

Fig. 2 An image of browsing on the social network using interface on comics style.

された個人プロフィールと関係プロフィールを把握しつつ、人間関係ネットワークをブラウジングしていくことが可能となる。以上の機能を実現するためには、機械可読な XML で表現された FOAF を人間可読な漫画表現された画像へとコンテンツ変換を行う機能と、画像上のキャラクターにリンクアンカを設定する機能が必要となり、これを漫画 FOAF ビューワの機能要求と定義する。

4. 関連研究

公開されている FOAF ファイルを直接読み込み、それを人間可読な表現に変換し、提示する人間関係ネットワークのブラウジング用ツールには表組を利用したものが多く、たとえば、FoaF Explorer¹⁰⁾ や FOAF: Web View に代表されるように、XSLT によって XML である FOAF を HTML に変換する仕組みのツールは多く提案されている。しかし、本来エージェント用データである RDF/XML で表現された FOAF を表組しても語彙や内容に本質的な変化はない。つまり、既存ツールが採用している表組では、リテラル等の末端の情報を提示する段階において何も工夫しなければ難解な語彙をそのまま提示することになるが、この場合、語彙の意味をあらかじめユーザに周知させるか、もしくは十分に一般化されていなければ正確な情報伝達は困難である。これに対して、語彙を表示する際に絵画や写真、アイコン等といった記号を用いて情報の意味を直感的に伝達する方法が考えられる。

このような絵画や写真、アイコンは文字や表組、グラフに比べて視覚心像（イメージ）に近い表現が可能であるとされている⁸⁾。漫画表現はこの絵画や写真、アイコンと文字が複合的に組み合わせられたコマの線条的表現である。画像と文字の組合せは表組においても同様であるが、漫画表現は表組では困難な因果関係、物語、起承転結をはじめとする複雑かつ多様な事柄を直感的に表現する能力を有している^{9),11),12)}。

表組のように個々の FOAF ファイルの内容を表現する様式ではなく、人間関係ネットワークの全体的な構造を巨視的に表現するツールも多く提案されている。たとえば、foafnaut! や FOAFCorp は、foaf:Person をノードとして関係プロフィールをエッジとする連結系⁸⁾のグラフ描画¹³⁾を行っている。このようなツールは本稿が対象とする微視的なブラウジングを本来の目的としていないが、たとえば foafnaut! では FOAF ファイルを 1 つ 1 つ逐次的に読み込んでいくことも可能である。しかし、個々の個人プロフィールを表示する段階においては文字情報やアイコンを表示することになるため、人間関係ネットワークのブラウザ用の表現という観点からは表組と同様の表現能力しか持たない。

FOAF People Map は、上記のような FOAF を変換するツールと異なり、FOAF における現在地情報から世界地図上の該当位置にその FOAF を指すポインタを置くアプローチをとる興味深い試みではある。しかし、人間関係ネットワークを表現しない点で本研究とは目的が異なる。

既存ツールは、FOAF を HTML や画像ファイルへとコンテンツ変換を行うトランスコーディング処理を行っているといえる。このような単純な変換に対して長尾は、デジタルコンテンツに付与されたメタデータ（文献では、アノテーションと呼ばれている）を基に知識処理を行うことで、高度なコンテンツ変換を実現するセマンティック・トランスコーディングを提案している¹⁴⁾。漫画 FOAF ビューワも FOAF から漫画表現へとコンテンツ変換を行う際にメタデータに基づくストーリー導出等の知識処理を行っているため、セマンティック・トランスコーディングの一形態ととらえることができる。

人間関係ネットワークとは直接関係しないが、グラフ表現が多い古典的な情報視覚化研究とは別のアプローチとして漫画表現の線条性もしくは現示性に着

目した研究も行われてきている。これらは Video インデックス¹⁵⁾、チャットシステム¹⁶⁾、コンテキストウェアシステム¹⁷⁾における個別のデータの視覚化に漫画表現、もしくはコマ割手法を適用している。しかし、本研究が対象とする人間関係ネットワークのブラウジング用に利用されることは想定されていない。

5. 漫画 FOAF ビューワ

本章では、3 章において動作イメージを説明した漫画表現による人間関係ネットワークのブラウザの実現方法を主にデータ処理の観点から説明する。まず、必要となるデータを定義し、これを処理するシステムの構成を述べる。次に FOAF を漫画のストーリーへと変換するトランスコーディングを中心にサーバ側の処理を説明し、最後に、実際に漫画としてレンダリングするクライアント側の処理を説明する。

5.1 パーツデータおよびメタデータ

漫画 FOAF ビューワは、漫画の構成要素を粒度の小さいものから順にパーツ、コマ、シーン、ストーリーという階層構造でとらえる。一般的に、漫画を構成する基本要素はコマとすることが多い^{9),11),12)}、漫画 FOAF ビューワではさらにコマを構成する図的要素に分解して考え、これをパーツと呼ぶ。シーンは、1 つ以上のコマのリストを指し、後述するトランスコーダにおける操作単位となる。ストーリーはシーンのリストであり、トランスコーダの最終的な出力となる。

漫画 FOAF ビューワは、この構成要素のうちパーツをパーツデータ、コマをコマ情報、シーンをシーン情報という各種データによって管理する。また、これに加え、シーン条件、制約条件という 2 種類のデータも利用する。パーツデータが最終的に漫画をレンダリングする際に表示されるのに対して、コマ情報、シーン情報、シーン条件、制約条件の 4 種類のデータはレンダリングに直接関係しない情報であるため、今後これらをメタデータと呼ぶ。

漫画 FOAF ビューワを利用するためには、対象となる FOAF 群に即したパーツデータとメタデータを準備する必要がある。このメタデータとパーツデータの間を関係を表す概念図を図 3 に示す。

後述する実験用プロトタイプでは、実装のために、オブジェクト記述形式として簡便な JSON (JavaScript Object Notation) に準拠した形式を採用した。図 4 にその記述形式を示し、以下、各データの説明とプロトタイプシステム上で動作するサンプルデータを示す。

パーツデータはたとえばキャラクタ、台詞、背景等 1 つのコマを構成する図的要素であり 1 つのコマに対

<http://jibbering.com/foaf/>

<http://rdfweb.org/foafcorp/intro.html>

<http://jibbering.com/2002/8/foaf-people-map.svg>

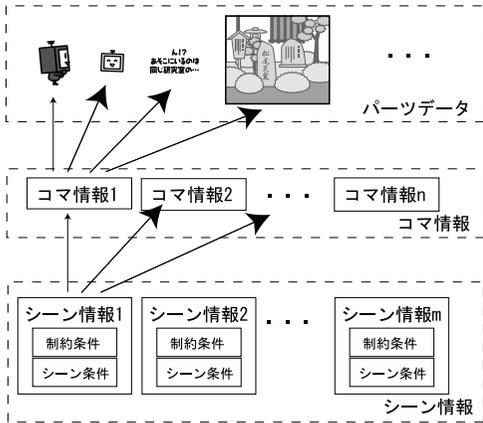


図3 パーツデータ, コマ情報, シーン情報, シーン条件, 制約条件の概念図

Fig. 3 A diagram of Part data, Frame data, Scene data, Scene condition, Constraint condition.

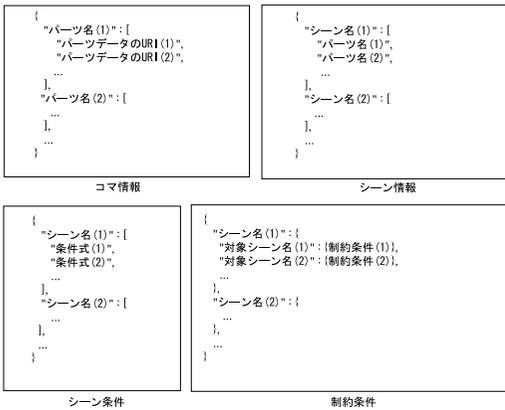


図4 各メタデータの表現形式

Fig. 4 Formats of metadata.

して1つ以上準備される。このパーツデータのプロトタイプにおける実装用のフォーマットはSVG¹⁸⁾である。図5はパーツデータのサンプルであり、このうちのいくつかを重ね合わせると図6左に示されたコマを構成することができる。

パーツデータのSVGファイルには、最終的なレンダリングの際にリンクアンカのための透明な絵として出力されるgノード(SVGにおけるグラフィックオブジェクトの単位)が含まれることがあり(図6), これがある場合はgノードのIDとリンク先となるFOAFのURIを導出するための条件もコマ情報に記される。なお、このリンクアンカの領域指定のためのgノードは、接頭語(「clipping:」)をつけておくことで一般のgノードと区別する。また、後に述べるコマに登場するキャラクタを特徴づける色に関する情報も記述される。

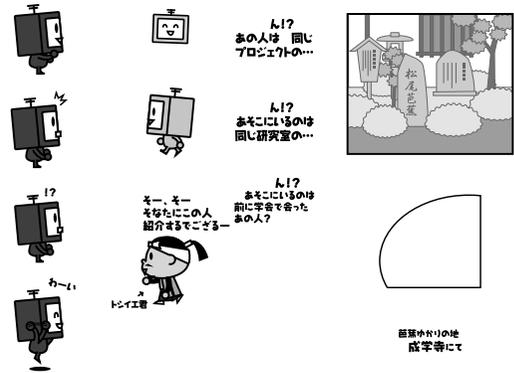
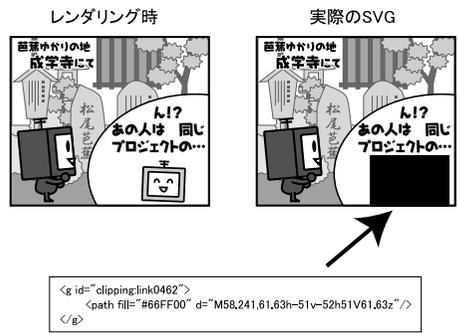


図5 パーツデータのサンプル

Fig. 5 Samples of parts data.



リンクアンカ用のノード

図6 SVG ノードIDの指定によるリンクアンカ領域の設定

Fig. 6 Setting link anchors by annotated node IDs of SVG.

コマ情報は、コマを構成するために必要な情報が各コマごとに記述されている。これは、1つ以上のパーツデータを指すURIのリストとコマの名前の対で構成され、パーツデータへのメタデータとなる。コマ情報に基づいてパーツデータを取得しリストの順に下から重ね合わせると1つのコマを形成することができる。たとえば、あるコマが“frameA”という名前で、その構成要素となるパーツデータのURIが重ね合わせ順の画面奥側から順に“./background.svg”, “./characterA.svg”だとするとコマ情報は以下のように記述される

```
"frameA": [
  "./background.svg",
  "./characterA.svg"
]
```

シーン情報は、コマ情報における1つ以上のコマ名のリストであり、コマ名のリストとシーン名の対で保持される。たとえば、あるシーンが“sceneA”であり、そのシーンは“frameA”, “frameB”というコマで構

成されていることを表現するシーン情報は以下のように記述する．

```
"sceneA": [
  "frameA",
  "frameB"
]
```

シーン条件と制約条件は、どちらもシーン情報に対するメタデータである．前者は各シーンについて、対象 FOAF がどのような状態時に利用されるのかという情報が記されており、后者はシーン間の矛盾や制約関係および各シーンの役割に関する制約が記されている．

シーン条件は、シーン名と対象 FOAF 中のノードに対する条件式の対で記述される．条件式は、XPath で特定された対象ノードに関する値の判定式をブール値で返す判定式で与えられ、たとえば、“sceneA”が、ルートノードの foaf:Person における foaf:name が sakamoto であるとき真値となるように設定するシーン条件は、以下のように記述する．

```
"sceneA": [
  "foaf:Person/foaf:name=="sakamoto"
]
```

制約条件における制約の種類は 2 種類存在する．一方は、シーン間の共存関係を制約する条件であり、たとえば、あるシーン “sceneA” は別のシーン “sceneB” との共起が許されない、直後に存在することが許されない、直前に存在することが許されない等といった制約関係として記述される．他方は、そのシーンがストーリーの先頭になりうるシーンである、もしくは末尾に位置してよいシーンであるというシーンの位置に関する制約である．両者とも制約関係は “condition”, “position”, “allowed”, “forbidden”, “both”, “role”, “tail” 等といった予約語によって指示され、たとえば “sceneA” は “sceneB” の後ろに存在してよいが、“sceneC” とは共存不可であり、なおかつストーリーの末尾として成立する制約条件は以下のように記述する．

```
"sceneA": {
  "sceneB": {
    "condition": "allowed",
    "position": "following"
  },
  "sceneC": {
    "condition": "forbidden",
    "position": "both"
  },
  "role": "tail"
}
```

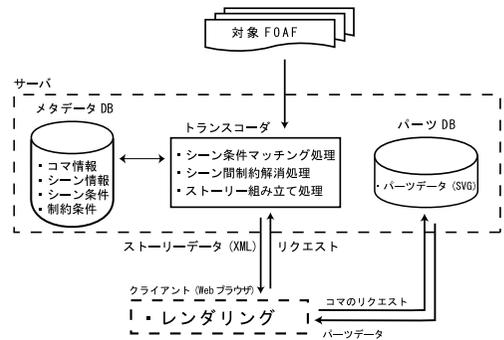


図 7 システム構成図

Fig. 7 The system architecture.

}

5.2 システム構成

システム構成図 (図 7) を基に全体の構成を説明する．漫画 FOAF ビューワは、構成上、大きくサーバ側とクライアント側に分かれる．サーバ側は、クライアントからのリクエストをトリガとして、源泉の FOAF から任意の foaf:Person を主人公とする漫画のストーリーを示す XML にトランスコードする．クライアント側は、サーバ側から返答されたストーリーの XML をもとに、実際のパーツデータを取得してレンダリング、表示を行う．

サーバ側は、パーツ DB、メタデータ DB、トランスコーダの各モジュールに分かれる．このうち、トランスコーダ以外は、対象 FOAF や表現したい漫画ごとに用意する必要がある．以下、それぞれのモジュールについて説明する．

パーツ DB SVG ファイルのパーツデータを保存管理する役割を担う DB であり、個々の SVG ファイルの URI を指定することによりクライアント側から直接アクセス可能とする CGI を持つ．

メタデータ DB 各種メタデータを保存しておく DB である．DB は、コマ情報、シーン情報、シーン条件、制約条件の 4 種類のテーブルで構成される．これらのメタデータは、既存のトランスコーディングに関する研究^{14),19)}におけるアノテーションと同じ役割を持つ．

トランスコーダ トランスコーディング対象の FOAF とメタデータからストーリーを組み立てるモジュールである．ストーリーとは、シーンのリストである．このリストに含まれるシーンどうしが話として矛盾しないよう、制約条件を基に制約を充足させる．ストーリーは使用されるパーツデータ群と配置位置が記された XML 形式で出力される．また、本モジュールは、クライアント側からの漫画

描画リクエストを受け付けるインタフェースの役割も請け負う。

クライアント側の役割は、ユーザインタフェースの提供、サーバへのリクエスト、パーツ DB からのパーツ取得、レンダリング、リンクアンカの設定を行うことにある。最終的にレンダリングされる画像の形式は SVG のままでも問題ないが、6 章で述べる実験用プロトタイプシステムでは、プラグインが導入された一般的な Web ブラウザをクライアントソフトとして利用することができるよう、SVG ファイルを Macromedia 社の Flash 形式に変換して出力した。

5.3 サーバ側の処理

サーバ側の処理は、FOAF ファイルをストーリーファイルの XML へ変換する一種の XSLT エンジンであるが、知識処理によりシーン間に矛盾のない漫画用ストーリーを導出する点で一般的な XSLT エンジンとは大きく異なる。まず、サーバ側は、クライアント側から HTTP の GET リクエストとして漫画表現を行う FOAF ファイルのリクエストを受付ける。リクエストがあると関連するメタデータをすべてロードした後、対象となるユーザの FOAF ファイルを HTTP で読み込み、ファイルをトランスコーダに渡す。トランスコーダは、FOAF からメタデータ DB を参照してストーリーとなる XML を導出することが目的である。よって、最終的な表示にのみ必要なパーツデータは扱わず、メタデータだけが処理される。トランスコーダは、メタデータ DB からシーン条件および制約条件をロードし、まず (a) 対象 FOAF に対して後述するシーン条件が真になるシーン群を抜き出す処理を行う。次に (b) この群における各シーン間の制約条件を満たしてストーリーとして矛盾しないシーン列をストーリーとして作成する。最後に (c) そのシーンを構成するコマにおけるパーツの URI のセットを XML 表現したストーリー XML をクライアントに返信する。以下、トランスコーダの各処理を説明する。

(a) 適合シーンのコレクション

まず、すべてのシーン条件のうち判定式が真になるシーン条件を持つシーンを収集する。この収集結果の集合を S_0 とする。

(b) シーン条件の制約充足

次にシーン集合 S_0 におけるシーン間の制約を充足して矛盾のないストーリーの集合 S を導出する。制約充足の手順を図 8 に擬似コードとして示す。この擬似コードにおける「#」の行はコメントを表し、`continue(n)` は n 段階外側のブロックに出る `continue` 制御を表す。また、2 項制約 $C(s_i, s_j)$ は、任意のシー

```

T ← {}
foreach( $s_i \in S_0$ )
begin
  #ストーリーの頭にくるシーンのコレクション
  if( $C(s_i, "head")$ )
  begin
    #要素数が 1 のリストとして追加
    T ← T ∪ { $s_i$ }
  end
end
S ← {}
while( $T$  の要素数が 1 以上)
begin
  Ttemp ← {}
  # L はリスト
  foreach( $L \in T$ )
  begin
    foreach( $s_i \in S_0$ )
    begin
      foreach( $s_j \in L$ )
      begin
        #適合しないシーンであれば追加しない
        if(! $C(s_j, s_i)$ ) continue(2)
      end
      #末尾にシーンを追加
      L ← append(L,  $s_i$ )
      if( $C(s_i, "tail")$ )
      begin
        # S に完成したストーリーを追加
        S ← S ∪ {L}
      end
    end
  end
  Ttemp ← Ttemp ∪ {L}
end
T ← Ttemp
end

```

図 8 ストーリー導出のアルゴリズム
Fig. 8 An algorithm for making stories.

ン s_i と他の任意のシーン s_j との共存関係に関する制約条件であり共存が許される場合真を返す関数である。 $C(s_i, "head")$ および $C(s_i, "tail")$ は、 s_i がストーリーの先頭もしくは末尾のシーンとして成立する場合に真になる制約条件を示す。 $\{\}$ および $\langle \rangle$ は、それぞれ集合とリストを示し、関数 `append(L, M)` はリスト L の末尾にリスト M を追加したリストを返す。

図 8 の処理は有限回で終了するが、 S_0 の要素数によっては時間がかかりすぎることもあるので、 S の要素数が 1 以上の場合、時間制限で処理は打ち切られる。この処理が終わると解集合 S を得るが、 S の要素数が 1 以上の場合、たとえば 4 コマになった一番最初の解、等といったあらかじめ与えられたルールにより 1 つのリストを単一な解として保持する。また、もし S の要素数が 0 である場合は、エラーを返す。

(c) ストーリーファイルの出力

S に含まれるシーン情報は、コマの名前が列挙されているため、この名前を基にコマ情報を参照することができる。コマ情報には、各コマが使用するパーツの実ファイルの保存場所が URI で記述されているので、これをシーンとコマの順列に沿って展開していく。この展開されたコマの URI 群とコマ情報をストーリー

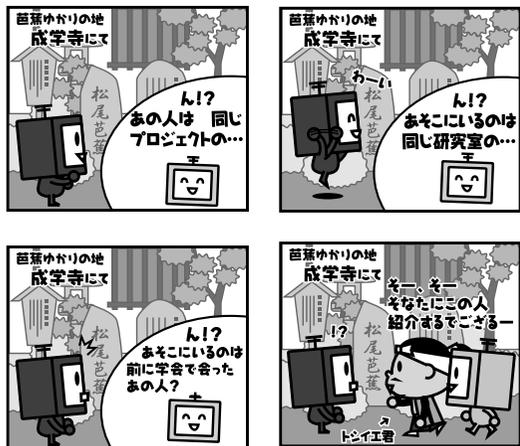


図 9 コマの例

Fig. 9 A sample of rendered frames.

ファイルとして XML で出力する .

5.4 クライアント側の処理

5.3 節の処理が終了すると、サーバ側は出力したストーリーファイルをクライアントのリクエストに対する応答として戻す。クライアント側では、ストーリーファイルを基にパーツ DB からパーツ群を取り出し、コマ情報のリストに従ってパーツを配置、重ね合わせて漫画をレンダリングする。図 9 に図 5 で示したパーツデータを用いて 4 パターンのレンダリングされたコマを示す。ここでは、異なる組合せのパーツが重ね合わせられ、コマとして出力されていることが分かる。

レンダリングの際には、コマ情報にふくまれるリンクの情報に従ってリンクアンカ用の g ノードの領域にリンクアンカを設定すると同時に、その g ノードのレンダリング処理をスキップする。図 10 に後に述べる運用実験においてレンダリングされた漫画の例を示す。漫画は 4 コマで構成されており、線で指示された脇役のキャラクタは、左側の語彙に対応した FOAF ファイルに対してリンクが設定されている。これらのリンクはリンクアンカ用 g ノードで領域指定されており、なおかつその g ノードは表示されていない。

また、レンダリングの際にリンク元でのキャラクタの色とリンク先での色を合わせる目的で、コマ情報に記載された元のキャラクタの色が指定されたキャラクタの g ノードにおける色指定のトリビュートを foaf:Person 固有の色へと置換する。この foaf:Person 固有の色は、読み込みを行った FOAF ファイルの URI 文字列を MD5 で変換、オクテットに再変換して RGB 値とする。この機能は、ユーザがリンクをたどっていく際のキャラクタと foaf:Person との認知的な結び付

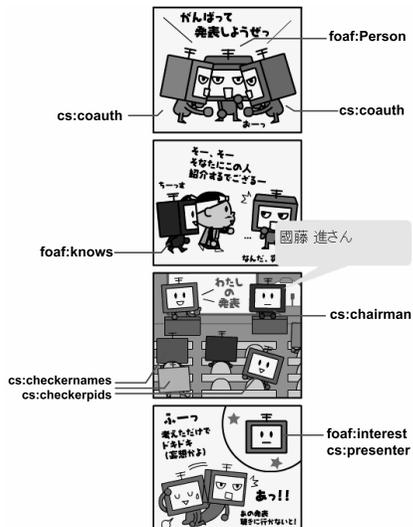


図 10 レンダリング結果とリンクアンカ

Fig. 10 A rendered result and link anchors.

きを維持することが狙いである。つまり、同じユーザを示すキャラクタはどの漫画上でも同じ色で表現されるため、ユーザは別の登場人物と混同することなく複数の FOAF 間をブラウジングすることが可能となる。

6. 実験

4 章において、漫画 FOAF ビューワは漫画表現を用いている点で既存の表組を表現様式として用いた人間関係ネットワークのブラウザに比べ、複雑かつ多様な事柄を直感的に表現する能力を有していることを論じた。しかし、漫画表現を用いたインタフェースで FOAF ファイル群をブラウジングすることが可能かどうかという点に関して、既存研究等では明らかにされていない。

よって、本章では、学会の参加者で構成される人間関係ネットワークに対して漫画 FOAF ビューワを適用し、参加者に対して運用した運用事例データを示したうえで、漫画 FOAF ビューワが個別の foaf:Person における人物プロフィールと関係プロフィールを表現可能か否かという点、および人間関係ネットワークのブラウジング環境として機能するか否かという点について検証する。

6.1 実験概要

実験は、第 18 回人工知能学会全国大会（以下、JSAI2004 と呼ぶ）の参加者に対して行われた。JSAI2004 は参加者のデータが最初から FOAF 形式で存在しているわけではなかったため、漫画 FOAF ビューワの導入に先がけて、参加者に関する情報を

表 3 JSAI2004 で定義した独自の語彙と意味
Table 3 Defined vocabularies and semantics on JSAI2004.

語彙 (接頭辞 cs:)	内容の説明
usertype	学会への参加形態
hntype	人間関係ネットワークシステムから取得した対象人物との関係
presenter	発表リソースの発表者 (登壇者) の氏名
presenterpid	presenter で示された人物が持つ ID (人物 ID, 学会側が提供)
presen_type	発表形式 (口頭発表, ポスター, 特別セッションでの発表等)
authorpid	発表リソースの著者を示す ID (著者の数だけ記述)
sid	セッションごとの個別の ID (学会側が提供)
chairmanname	セッションの座長に振られた人物 ID
chairmanpid	座長が持つ人物 ID
checkernames	発表をスケジュールに設定した人物の氏名 (群)
checkerpids	checkernames の人物が持つ人物 ID (群)

表 1 学会への参加形態を記述するための語彙と意味
Table 1 Vocabularies and semantics for describing a user's participation type in JSAI2004.

語彙 (cs:usertype)	学会への参加形態
presenter	発表者として参加
chairman	座長として参加
coauthor	共著者として参加

表 2 人間関係ネットワークシステムから得られた情報の表現語彙と意味

Table 2 Vocabularies and semantics from the human network system.

語彙 (cs:hntype)	人間関係
coauth	共著関係
project	プロジェクトメンバ
lab	同じ研究室のメンバ
conference	学会で発表したことがある
no	関連なし

FOAF ファイルとして抽出する作業を行った。FOAF ファイルは、参加者 1 人に対して 1 ファイル作成し、リンク関係を構築することで人間関係ネットワークを構築することにした。この FOAF 化の対象となった参加者は 541 人である。

JSAI2004 では学会参加に対する総合的な支援プロジェクトがあり、様々なアプローチで学会参加を支援するサービスが 7 種類提供されていた²⁰⁾。FOAF の作成には、このうちのスケジューリング支援システム²¹⁾と人間関係ネットワーク支援システム²²⁾のデータを利用した。具体的には、スケジューリング支援システムから個人プロフィールを構成し、人間関係ネットワーク支援システムから関係プロフィールを構成した。

個人プロフィールは、名前、所属、参加形態 (表 1)、スケジュールに設定されている聴講予定の発表、その人物が発表もしくは共著関係にある発表の各情報からなる。このうち名前と所属の情報は、foaf:name と foaf:Organization の語彙に対応するので、これをそのまま利用した。また、聴講スケジュールに登録されている発表は興味がある事柄という意味で foaf:interest で表現した。しかし、本人が登壇する発表もしくは共著者が登壇する発表を聴講スケジュールに登録している人や座長の情報といった既存の語彙では表現しきれない情報は既存の適当な語彙が存在しないため、XML 名前空間において cs という名前空間を宣言して語彙を与えた。表 3 にその一覧と説明を示す。

関係プロフィールとしては、人間関係ネットワークにおける人物関係のうち、関係性の深い人物を foaf:knows が指す foaf:Person として設定することにした。具体的には、人間関係ネットワーク支援システムにおける、ある人物を基準として関連性の深い人物群の上位 n 人の人名を返す Web サービスから、関係のある人物の情報を最大 5 人分獲得して設定した。この際、人間関係ネットワークが判別する 2 人の関係 (表 2) を foaf:knows の目的語となる foaf:Person のプロパティとしてセットした。なお、表 2 における語彙「no (関連なし)」は、プロパティとしては存在するものの、このプロパティに関するシーン条件は設定しなかったため実際のストーリーには反映されなかった。

6.2 パーツデータとメタデータの準備

パーツデータは、キャラクタやセリフ等 64 個用意し、シーンは 9 個用意した。図 5 に示したサンプルは、実験時に用いたパーツデータの一部である。また、メタデータにおけるシーン条件は、主に表 3 のリテラルに対してパーツを割り当てるよう作成し、特に対象参加者の参加形態 (表 1) によってストーリーが大きく異なるよう設定した。パーツデータとメタデータは、漫画作成経験を持つデザイナー 1 人が 1 週間ほどの時間をかけて準備した。

表 4 アクセス数

Table 4 The number of accesses.

集計結果	
アクセス総数 (6/3~6/25)	280 件
スケジューリング支援システムからのアクセス	136 件
人間関係ネットワークシステムからのアクセス	128 件
リンク元が不明なアクセス	16 件
のべ利用人数	118 名

6.3 結果と考察

表 4 に Web サーバのログから得た運用結果データを示す。なお、以下に示すすべてのデータは、筆者を含む学会支援サービス運営者の利用分を除外して集計されている。JSAI2004 は 6 月 2 日から 5 日の 3 日間開催されていたが、本サービスが利用可能になった旨のアナウンスが遅れたため、実質的に利用は 6 月 3 日から可能であり、6 月 25 日までの約 3 週間に 280 件のアクセスが確認された。また、後日学会支援サービス全体でアンケートをとる機会が設けられたため、本サービスでも簡単なアンケート調査を行った。結果、学会支援サービス全体のアンケートに対して 125 人の方が回答し、そのうち 15 人が本サービスを利用したと回答した。

本サービスは、スケジューリング支援システムと人間関係ネットワークシステムの HTML 上から利用することができた。スケジュール支援システムは、運用上ログインした直後にそのユーザの個人化された画面が表示され、そこに自分の FOAF を表示する漫画 FOAF ビューワへのハイパーリンクがある状態であった。一方で、人間関係ネットワークでは、人名の検索機能があり、その結果画面から検索対象人物の FOAF の漫画を作成するリンクアンカが張られていた。前者のリンクからは 136 件、後者からは 128 件のアクセスが確認された。本サービスではユーザ認証をしなかったため正確な利用人数は不明であるものの、セッション情報から得られたのべ利用人数は 118 人であった。これらの情報と IP アドレスの重複から判断すると少なくとも 60 人の利用はあったと推察される。

これらのユーザが漫画間のリンクをたどった平均パス長は 1.36 ($\sigma^2 = 8.77$) であった。これは、平均 2 枚強の漫画を読んだことを示している。このうち、パス長が 0、つまり単一の漫画しか読まなかったユーザがのべ 71 人おり、全体の 60% を占めていた。これら単一の漫画しか読まなかったユーザを除けば、たどったリンクの平均パス長は 3.42 ($\sigma^2 = 15.08$) であり、一度操作方法が分かれば 4~5 枚の漫画をブラウジングしていったことになる。以上の結果から、操作方法

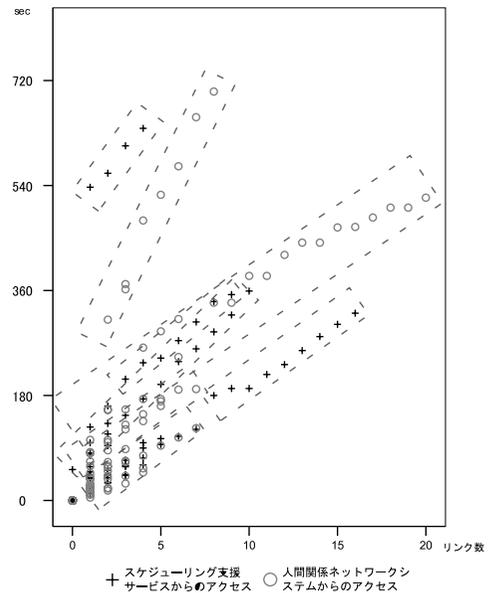


図 11 リンク間の移動回数と経過時間の関係

Fig. 11 The relationship between jumping times and elapsed time.

が理解できたユーザにとって、漫画表現された FOAF のリンク間をブラウジングしていくことは可能であったと考えることができる。一方で、この結果は、対象ユーザのドメインやインタフェースのユーザビリティによって大きく左右されると考えられるうえ、比較対象がないため、この結果によって提案手法の優劣を述べることはできない。ただし、「漫画に現れているリンクに気づきましたか」というアンケート質問の回答に約 3 割が気づかなかったと回答していることから、操作方法やツールの意義が周知されていたとはいえない。さらに、画面を 2 回クリックしなければ漫画間を移動できない仕様であったこと等も含めてインタフェースと運用方法に課題を残す結果となった。

図 11 は、ユーザがリンクを移動した回数と経過時間の関係を示す散布図であり、1 ユーザが 1 回で移動した点列が破線の矩形によって指示されている。これを見ると、移動回数の少ない箇所に点が集中しているものの、リンク回数の多い箇所に注目すると点が直線的に並んでいる箇所が多く、一定のリズムでブラウジングを行った形跡がうかがえる。一方、個々の矩形の傾きの差異は、漫画を読み次の漫画へと移動する時間の個人差を示している。これは、被験者間のストーリーを読解するペースや、レンダリングを行った PC のマシンパワーに起因すると考えられる。

表 5 は、アンカにおけるリンク元とリンク先の foaf:Person の参加形態が判別可能な漫画において、

表 5 リンク元の人物 A とリンク先の人物 B との関係 (A-B 間のアンカの内容) における移動件数とアンカ総数

Table 5 The number of jumpings and anchors on the each relationship of going link and incoming link.

アンカ名	リンク A B の関係	移動件数	総数 (平均・分散)
a1	A が座長のセッションを人物 B がスケジュールに設定	3 件	125 個 (0.64・1.77)
a2	A が座長を務めているセッションの発表者 B	5 件	51 個 (0.52・0.77)
a3	A の発表の座長 B	19 件	64 個 (0.65・0.22)
a4	A の発表を B がスケジュールに設定	15 件	336 個 (1.71・2.90)
a5	A が興味を持つ発表を B がスケジュールに設定	2 件	62 個 (0.81・0.31)
a6	A が興味を持つ発表の発表者 B	20 件	60 個 (0.61・0.24)
a7	A と B は本学会において共著者	15 件	9 個 (0.09・0.17)
a8	A は B を知っている (foaf:knows)	43 件	98 個 (1.00・0.02)

アンカごとの移動件数とそのアンカが 1 つの漫画上に存在した総数である。これによると、知人関係のアンカである a8 をたどったケースが最も多く全体の 35% を占めるが、アンカ自体の総数は全体の 12% とそれほど多くない。逆に、実際の知人関係かどうか不明な foaf:knows 以外のリンクのほうが圧倒的に多く、foaf:knows がリンクの移動先として選択される機会は少なかったはずである。たとえば、a8 は、a4 や a1 よりも総数が少ないにもかかわらず、たどられた件数は両者の合計よりも多い結果となっている。本システムの利用形態から考えて、自分の FOAF ファイルから表示し始めるケースが多かったと考えられるため、少なくとも 1 つ目に表示した漫画における a8 は、ユーザの知人であった可能性が高い。また、foaf:knows の次に多いリンク先は、a3 の「A の発表セッションの座長 B」であり、座長は多くの場合有名人であるため、事実上、「知っている人」つまり一方的な知人であったと考えられる。さらに、移動先の漫画において最頻のストーリーであった a3 を 1 個、a4 を 5 個、a8 を 1 個持つ漫画において、各リンクがたどられるであろう期待値と実際にたどられた数によるカイ自乗検定を行ったところ、有意差が認められた ($\chi^2(2, N = 26) = 14.41, p < .01$)。これら知人を示したアンカに移動先が偏るという結果は、ユーザが何も考えずキャラクタをクリックしていたのではなく、キャラクタを選別してブラウジングを行っていたことを示しており、漫画 FOAF ビューワが漫画表現によってユーザに人物および関係プロフィールを提示できていたことを示唆している。

最後に使用感や印象を問うアンケートを紹介する。まず「サービスは便利だったか」という質問に対して、「便利である」と答えた人は 7 人、「どちらともいえない」は 3 人、「そう思わない」は 4 人という結果となった。次に、「サービスは面白かったか」という質問に対しては「面白かった」が 14 人、「どちらともいえない」が 0 人、「そう思わない」が 1 人という結果となっ

た。最後に「サービスを来年も利用したいか」という質問に対しては「利用したい」と答えた人が 10 人、「どちらともいえない」が 2 人、「そう思わない」が 1 人という結果となった。これらの回答結果は、本サービスが利用者におおむね好感をもって利用されたことを示唆している。以上の結果は、漫画 FOAF ビューワは FOAF が持つ機械可読なプロフィールを人間可読な表現として提示し、操作方法に慣れが必要なものの FOAF による人間関係ネットワークのブラウザとして機能したことを示している。

7. おわりに

本稿では、漫画表現を用いた FOAF で構成された人間関係ネットワークのブラウジング環境を提案し、RDF/XML である FOAF を漫画表現へとコンテンツ変換するための知的トランスコーディング処理と漫画どうしのハイパーリンクを実現する漫画 FOAF ビューワについて述べた。漫画表現は、既存 FOAF ビューワの表やグラフによる表現に比べて情報の意味を表現しやすい点で有利であるが、ブラウジング用インタフェースとして用いた例はなかった。これに対して本稿では、SVG ファイルの特定ノードをリンクアンカとして設定して漫画間のリンク関係を構築することによって、漫画表現とブラウジングを両立させる方法を示した。また、学会会議における参加者の FOAF を対象に運用した例を示し、提案システムが人間関係ネットワークのブラウザとして機能したことを示した。

本稿で示した運用例では、学会会議の参加形態によってシーン条件を設定したため比較的容易に異なったストーリーを生成することができた。よって、パリエーションに富むプロフィールを一元管理する場所、たとえばソーシャルネットワークサービス等といった、運用例に似た箇所において提案システムは容易に適用可能であると予想され、人間関係ネットワーク表現の一形態として活用されることが期待される。しかし、

実際に Weblog 等で公開している FOAF をブラウジングしようと考えた場合、個人によっては情報量が少ない場合が多くバリエーションを増やすことが困難である。今後、たとえばハブにあたる人物に適当なストーリーを割り当てる等、ネットワーク構造上の立場をシーン条件として設定可能にしたり、Web 上の画像リソースを有効に利用した漫画生成等を行い、分散した人間関係ネットワークの漫画表現にも取り組みたい。

謝辞 実験にご協力いただいた JSAI2004 関係者の皆様、イベント空間情報支援プロジェクトメンバおよびイラストレーターの中尾恵子氏に感謝する。本研究は、情報通信研究機構の研究委託により実施したものである。また、処理エンジンの一部に独立行政法人情報処理推進機構の委託によって開発したものを利用した。

参 考 文 献

- 1) Berners-Lee, T., Hendler, J. and Lassil, O.: *The Semantic Web*, Scientific American (2001).
- 2) Beckett, D.: RDF/XML Syntax Specification (Revised).
<http://www.w3.org/TR/rdf-syntax-grammar/>
- 3) Brickley, D.: Friend of a Friend.
<http://www.foaf-project.org/>
- 4) Ishida, T. (Ed.): *Community Computing and Support Systems*, Springer-Verlag (1998).
- 5) 武田英明, 大向一輝: Weblog の現在と展望—セマンティック Web およびソーシャルネットワーキングの基盤として, 情報処理, Vol.45, No.6, pp.586–593 (2004).
- 6) Igata, N., Tsuda, H., Katayama, Y. and Kozakura, F.: Semantic groupware and its application to KnowWho using RDF, *ISWC2003 Poster*, pp.20–23 (2003).
- 7) Clark, J.: XSL Transformations (XSLT) Version 1.0. <http://www.w3.org/TR/xslt>
- 8) 出原栄一, 吉田武夫, 渥美浩章: 図の体系図的思考とその表現, 日科技連出版社 (1986).
- 9) 呉 智英: 現代マンガの全体像, 史輝出版 (1990).
- 10) Frederiksen, M.: FoaF Explorer.
<http://xml.mfd-consult.dk/foaf/explorer/>
- 11) McCLOUD, S.: *UNDERSTANDING COMICS*, Kitchen Sink Press (1994).
- 12) 四方田犬彦: 漫画原論, 筑摩書房 (1994).
- 13) 杉山公造: グラフ自動描画法とその応用, 計測自動制御学会学術図書, コロナ社 (1993).
- 14) 長尾 確: アノテーションに基づくデジタルコンテンツの高度利用, 情報処理, Vol.42, No.7, pp.668–675 (2001).
- 15) Uchihashi, S., Foote, J., Girgensohn, A. and Boreczky, J.: Video Manga: Generating Semantically Meaningful Video Summaries, *Proc. ACM Multimedia '99*, pp.383–392, ACM Press (1999).
- 16) Kurlander, D., Skelly, T. and Salesin, D.H.: Comic Chat, *Proc. SIGGRAPH96*, pp.225–236, ACM Press (1996).
- 17) 坂本竜基, 角 康之, 中尾恵子, 間瀬健二, 國藤進: コミックダイアリ: 漫画表現を利用した経験や興味の伝達支援, 情報処理学会論文誌, Vol.43, No.12, pp.3582–3595 (2002).
- 18) Ferraiolo, J., Fujisawa, J. and Jackson, D.: Scalable Vector Graphics (SVG) 1.1 Specification. <http://www.w3.org/TR/SVG11/>
- 19) Hori, M., Kondoh, G., Ono, K., ichi Hirose, S. and Singhal, S.: Annotation-based Web content transcoding, *Proc. 9th international World Wide Web conference on Computer networks : the international journal of computer and telecommunications networking*, pp.197–211, North-Holland Publishing Co. (2000).
- 20) 濱崎雅弘, 武田英明, 大向一輝, 沼 晃介, 上松大輝, 市瀬龍太郎: 2004 年度人工知能学会全国大会スケジューリング支援システムの開発と運用, 人工知能学会全国大会 (2005).
- 21) 濱崎雅弘, 武田英明, 大向一輝, 市瀬龍太郎: パーソナルネットワークを利用したコミュニティシステムの提案と分析, 人工知能学会論文誌, Vol.19, No.5, pp.389–398 (2004).
- 22) 浅田洋平, 友部博教, 松尾 豊, 石塚 満: Web 上の情報からの大規模人間関係ネットワークの構築と分析, 人工知能学会全国大会 (2004).

(平成 18 年 5 月 30 日受付)

(平成 18 年 11 月 2 日採録)



坂本 竜基 (正会員)

2003 年北陸先端科学技術大学院大学知識科学研究科博士課程修了。同年 ATR 知能ロボティクス研究所研究員。現在, ATR 知識科学研究所研究員。2005 年より奈良女子大学非常勤講師。博士 (知識科学)。CSCW, インタフェースの研究開発に従事。ACM, 日本創造学会各会員。



田中 郁 (学生会員)

2005 年北陸先端科学技術大学院大学知識科学研究科博士前期課程修了。現在, 同大博士後期課程に在学中。知識共有システムの研究に従事。



小暮 潔（正会員）

1981年慶應義塾大学大学院工学
研究科電気工学専攻修士課程修了。
同年日本電信電話公社に入社。現在
はATR知識科学研究所所長。博士
（工学）。自然言語処理，エージェン
ト，ロボット，知的環境等の研究に従事。電子情報通
信学会，人工知能学会，言語処理学会，日本認知科学
会，日本音響学会各会員。
