

ユーザモデルの記述のためのオントロジー構築に関する考察

官上 大輔[†] 河合 由起子[†] 田中 克己^{†,††}

[†] 情報通信研究機構けいはんな情報通信融合研究センター
^{††} 京都大学大学院情報学研究科 社会情報学専攻

abstract: 近年、ユーザ適応技術への要求が強くなっている。そこで我々はウェブシステムに適応に必要な機能を提供するためのフレームワークである A3 (Adaptation Anywhere & Anytime) を提案している。このフレームワークでは、ユーザプロフィールとしてユーザオントロジーを利用している。ユーザオントロジーは、ユーザがウェブから得た情報 (リソース) を、その属性値などに基いて分類した知識体系として定義される。しかし、あるリソースについてユーザが全ての属性情報を知っている、あるいは分類に利用しているということは考えにくく、どの属性を分類に用いて良いか、といった点を考慮する必要がある。本稿では、リソースとその属性、分類の際に用いられる属性とそうでない属性との差異などについて考察する。

Ontology Construction for describing a user model

Daisuke KANJO[†] Yukiko KAWAI[†] Katsumi TANAKA^{†,††}

[†] Keihanna Human Info-Communication Research Center, National Institute of Information and Communications Technology
^{††} Dept. of Social Informatics, Graduate School of Informatics, Kyoto University

abstract: We proposed a framework for a user adaptation, called A3 (Adaptation Anywhere & Anytime). This framework provides functionalities for a user adaptation. In this framework, "user ontology" is used as a user profile. A user ontology is a classified tree consisting of web resources and its categories. In this paper, we describe a user ontology in detail.

1 はじめに

近年、ネットワークの発展や blog をはじめとする情報提供手段の一般化によって、ユーザが手に入れられる情報は著しく増加した。しかし、入手できる情報が増加したのにもない、本当に必要としている情報の発見や提示は、情報を得ようとするユーザと、情報を提供しようとするシステムの双方にとって困難なものとなった。また、携帯端末や RFID などの機器の発達には、ユーザが様々な環境にある任意の端末からネットワークへのアクセスを可能にした。そこで、ユーザが要求する情報を適切な表現で任意の端末へと提示する、ユーザ適応技術の実現が期待されるようになっていく。

このような背景のもと我々は、適応システム構築のためフレームワークである、A3 (Adaptation Anywhere & Anytime) フレームワークを提案している [16]。こ

のフレームワークは、適応のために必要な機能をウェブシステムに提供する。適応システムは、出力する情報をユーザに適したものにするため、当該のユーザの知識や嗜好を表したプロフィールを必要とする。A3 フレームワークでは、ユーザプロフィールを動的に構築し、かつ、複数のウェブシステムで共有することで、従来の適応システムが持っていたプロフィール獲得のためのユーザの負荷や適応精度の差異といった問題を解決している。

A3 フレームワークでは、ユーザオントロジーと呼ばれるオントロジーによってユーザプロフィールを表現している。ユーザオントロジーは、ユーザがウェブシステムから獲得した各種の情報 (リソース) に関する情報を分類して整理した知識体系として定義され、ユーザとシステムの間インタラクションに基いて動的に獲得される。ユーザオントロジー獲得の際には、リソース

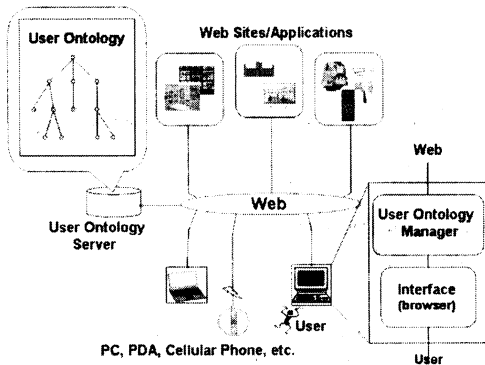


図 1: A3 フレームワーク

が示す対象オブジェクトの属性などが利用される。例えばリソースが書籍を示すものである場合、タイトルや著者といった書籍に関する属性情報が用いられる。現状では、付与されているあらゆる属性情報を用いて分類を行っている。

しかし、一般的に人が物事を分類する際にあらゆる属性情報を用いているとは考えにくい。書籍の場合、著者や出版社といった情報を利用していると考えられる一方、価格などを分類に用いていると考えることは直観に反している。また A3 フレームワークでは、ユーザに提示する情報を決定するためにユーザオントロジーを用いるが、それをどのように用いるか、という点が問題となる。ユーザが存在しているコンテキストなどによって考慮する属性などは異なると考えられるため、ユーザオントロジーから、どのような情報（プロフィール）を獲得するか、ということが重要となる。

そこで本研究では、まず、リソースとリソースを持つ属性について考察を行い、オントロジー構築に利用できる属性やそうでない属性について検討する。その後、実際にオントロジーを構築するに際しての問題を明らかにするとともに、それに対する対処について考察する。

2 A3 フレームワーク概要

図 1 に A3 フレームワークの構成を示す。ユーザは通常の PC や PDA、あるいは携帯電話などの端末から A3 フレームワーク上に構築されているウェブサイトやウェブアプリケーションなど、各種のウェブシステム

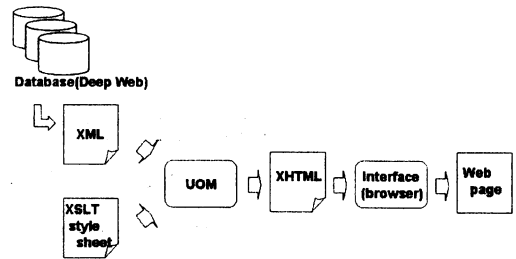


図 2: XSLT による適応過程

を利用する。ウェブシステムからは、ユーザの要求を満たし、かつ、より関心の強いと思われるリソースを含むコンテンツがウェブページなどの表現形式でユーザに提示される。ここでリソースとは、ショッピングサイトの商品やニュースサイトのニュース記事などの特定のオブジェクトを表す情報を指すものとする。

A3 フレームワークは、ウェブシステムの構築を容易にするため、XSLT[8] とその拡張要素を用いた構築手段を提供している。XSLT は、XSLT スタイルシートに従って、XML ドキュメントを異なったフォーマットのドキュメントへと変換する手段の一つであり、変換の際に様々な機能を実現する。例えば、`xsl:sort` 要素は、`apply-templates` 要素にマッチする XML ノードをアルファベットの昇順、もしくは降順にソートする。同様に、A3 フレームワークでは、いくつかの拡張要素によって、変換の際に適応的な機能を実現する。例えば、リソースの適応的な選択を実現するための拡張要素に `a3:sort` 要素がある。XML ノードでリソースを表現した場合、`a3:sort` 要素は `xsl:sort` 要素と同様に `xsl:apply-templates` 要素にマッチするリソースを、よりユーザに関心を持ちそうなリソースが上位に来るようにソートする。また、ユーザオントロジーの構築を実現するための拡張要素として `a3:add_resource` 要素と `a3:resource_template` 要素がある。前者は任意の HTML の `input` 要素に対し、その要素を介した入力が生じた際に、リソースをユーザオントロジーに加えることでユーザオントロジーを構築することを示す要素であり、後者は加えられるリソースを定義する要素である。

XSLT による適応過程を図 2 に示す。ウェブシステムからは、

- XML ドキュメント：ユーザに提示可能な全てのリソースを含む。

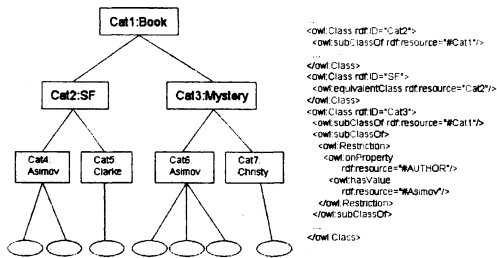


図 3: 例：ユーザオントロジー

○ XSLT スタイルシート：リソースの特定や実行する適応機能を記述する。

の 2 つのドキュメントが User Ontology Manager (UOM) へと送られる。

UOM はユーザの利用する各端末に存在し、リソースのソートやユーザオントロジーの構築を実際に行う。構築されたユーザオントロジーは、ユーザがウェブシステムの利用を終えると User Ontology Server (UOS) へと転送される。その後、ユーザが再び、任意のウェブシステムの利用を開始すると、そのユーザのユーザオントロジーが利用している端末の UOM へと送られ、適応のために用いられる。このように、あるシステムの利用によって構築されたユーザオントロジーが、他のシステムを利用する際にも利用される。したがって、同じユーザオントロジーに基づいて適応が行われ、従来の適応システムが持っていたシステムごとに適応精度が異なるといった問題が解消される。また、オントロジーの操作は全て UOM で同じ方策に基いて行われるため、一貫性が保持されるとともに、ユーザのプライバシーについても守られる。適応的なウェブシステムの構築のために作成者が行う作業は、XML ドキュメントと XSLT スタイルシートの 2 つのドキュメントの作成だけである。作成者は、リソースの選択やユーザオントロジーの構築などのアルゴリズムを考案することなく適応システムを構築できる。

3 ユーザオントロジー

上で述べたように A3 フレームワークでは、ユーザプロファイルは“ユーザオントロジー”によって記述される。図 3 にユーザオントロジーの例を示す。ユーザオントロジーは、リソース (図 3 楕円) とそのカテゴリ (図 3 四角) よりなるリソースに関する分類木と

して定義され、RDF/S[1, 3] および OWL[10] を用いて記述される (図 3 右)。

リソースは 2 章で述べたように特定のオブジェクトを指し示す情報であり、ユーザに提示する情報を選択する際に選択対象として機能する。具体的には、ショッピングサイトにおける商品の情報やニュースサイトにおける個々の記事などであり、名称や作成者などの属性情報を含んでいる。リソースとその属性は、ユーザオントロジーと同様に RDF/S, OWL を用いて記述され、URI によって一意に特定可能であると仮定する。全てのリソースは特定のカテゴリにカテゴリ化される。

カテゴリは、そこにカテゴリ化されるリソースが持つべき属性を制約として持つ。各カテゴリは固有の制約を持ち、かつ、上位カテゴリの制約を継承する。各カテゴリにはその制約を満たすリソースがカテゴリ化される。図 3 の場合、カテゴリ *Cat4* は「作者=Asimov」という固有の制約と、上位カテゴリ *Cat1* と *Cat2* の「対象=Book」、および「ジャンル=SF」という制約を持ち、カテゴリ化されているリソース *r1, r2* は、これらの制約を満たす。

3.1 ユーザオントロジーの構築

ユーザオントロジーの構築は、ユーザがリソースに関する知識を得たと判断できる際に、そのリソースを当該ユーザのユーザオントロジーへと加えることで行われる。XSLT を用いたシステム実装の場合、ユーザからリソースに関する何らかの入力が行われた際にユーザが知識を持っていると判断し、そのリソースをユーザオントロジーに加える。どの入力かユーザによる知識の獲得を示すものかの判断は、ウェブシステムの作成者、すなわち XSLT スタイルシートの作成者に委ねられる。

リソースが追加されると、以下の 3 つの手続きに従ってユーザオントロジーが構築される。

1. 既存のカテゴリへのリソースの追加

対象リソースが制約を満たすカテゴリの中で、最もルートカテゴリから距離の大きなカテゴリにカテゴリ化される。図 4 の (a) に例を示す。リソース *r1* は *Cat1* の制約を満たし、その下位カテゴリである *Cat2* ~ *Cat7* の制約は満たさない。結果、リソースはカテゴリ *Cat1* にカテゴリ化される。

2. 新規カテゴリの作成とリソースの追加

同じカテゴリにカテゴリ化されているリソースのうち、同じ属性を持つリソースの数が閾値 *t* を越える

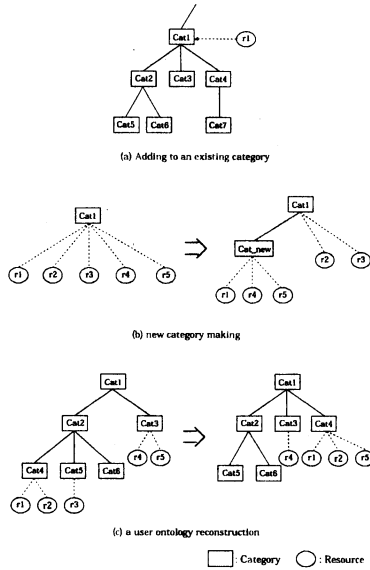


図 4: ユーザオントロジーの構築

場合、その属性を制約とする新たなカテゴリを作成し、そこにリソースをカテゴリライズする。図 4 の (b) に、 $t=3$ の場合の例を示す。

3. カテゴリの再構築

(1), (2) の操作の後、図 4(c) のような親カテゴリの親カテゴリにカテゴリを移動可能かどうかを検討する。この操作を行った場合と行わない場合のそれぞれについて次式を計算し、操作を行った場合の方が値が大きくなる場合、操作を実行する。

$$\sum_{k=1}^m \left(\frac{1}{2}\right)^{x_k-1} * n_k * \left(\frac{n_q}{n_p}\right) * d_{tmp_k} \quad (1)$$

- m : ユーザオントロジー中のカテゴリの数。各カテゴリを Cat_k とする時、 $1 < k < m$ 。
- x_k : Cat_k と Cat_0 (ルートカテゴリ) の間のカテゴリの数。
- n_k : Cat_k にカテゴリライズされるリソースの数。
- n_p : ユーザオントロジー中の全てのリソースの数。
- n_q : ユーザオントロジー中の Cat_k の制約となっている属性を持つ全てのリソースの数。
- d_{tmp_k} : Cat_k に最後にリソースがカテゴリライズされてからの期間によって決まる 0 から 1 の値。

ユーザの興味の対象の推移を反映する。

n_q/n_p は、 Cat_k の制約で示される属性を持つリソースの重要度を、ユーザオントロジー中における出現頻度によって評価したものである。これは「ユーザは興味のある対象に関する知識をより多く持つ」という仮定に基づく操作である。ユーザオントロジーの再構築は、重要とされる属性を制約として持つカテゴリを、より上位のカテゴリとする操作として捉えられる。

4 ユーザオントロジーのセマンティクス

4.1 ユーザオントロジーから得られる情報

本節では、構築されたオントロジーからどのような情報を得ることができるのかについて述べる。

4.1.1 ユーザの知識の特定

3.1 節で述べたように、ユーザオントロジーは、ユーザがあるリソースについて知識を得たと考えられる時に、そのリソースをユーザオントロジーに追加することで構築される。したがって、ユーザオントロジーに、あるリソースが存在する場合、ユーザはそのリソースに関する知識を持っていると判断できる。さらにユーザオントロジー構築の際には、リソースの属性情報が利用されることから、当該のリソースがどのような属性を持っているかを知っていると判断できる。既知リソースを特定することで冗長性を無くすことや、ユーザへの説明提示に既知の属性情報を利用する、といったことが可能になる。

4.1.2 ユーザの関心の特定

一般に人は関心のある対象について、より多くの知識を持つと考えられる。上記のように、ユーザオントロジー中のリソースをユーザの知識を表すものとして捉えると、より多くのリソースがカテゴリライズされているカテゴリにカテゴリライズされるリソースは、ユーザにとって関心の高いものである可能性が高い。このようなリソースを選択することで、ユーザにとって関心のあるリソース選ぶことができると考えられる。

A3 フレームワークでは、この考え方に基き、各リソースの重みを計算してランキングを行うことが可能であり、そのために XSLT の拡張要素である `a3:sort`

要素を提供している。この要素を用いることで、ウェブシステムの作成者は、容易に適切なリソースを選択することが可能となっている。

4.2 リソースの選択

A3 フレームワークでは、4.1 節で記したユーザオントロジーから得られる 2 種類の情報に基づいて、リソースの選択に関する 2 つの機能を提供している。

- 既知リソースの特定

ウェブシステムから UOM に送られる XML ドキュメントに含まれるリソースの中で、ユーザに既知のリソースを特定し削除する。リソースに与えられる URI に基づいて、ユーザオントロジー中に同じ URI を持つリソースが存在する場合、そのリソースは既知であると判断される。

- リソースの順序付け

ウェブシステムから UOM に送られる XML ドキュメントに含まれるリソースに対するユーザの関心度を重みとして計算し、順序付けを行う。

対象となるリソースがカテゴリ化されるカテゴリを Cat_n 、その上位カテゴリを $Cat_{n-1}, Cat_{n-2}, \dots, Cat_0$ とすると、その重みは次式で与えられる。

$$\sum_{k=1}^n \left(\frac{1}{2}\right)^{k-1} * \left(\frac{n_{qk}}{n_{pk}}\right) * d_{tmpk} \quad (2)$$

n_{pk} はユーザオントロジー中の全リソースの数、 n_{qk} は Cat_k とその下位カテゴリにカテゴリ化される全リソースの数である。 n_{qk}/n_{pk} は、4.1.2 節で記した「より多くのリソースがカテゴリ化されているカテゴリのリソースに関心を持つ」という仮定に基づき、多くのリソースがカテゴリ化されているカテゴリにカテゴリ化されるリソースに、より大きな重みを与える。 d_{tmpk} は Cat_k に最後にリソースがカテゴリ化されてからの期間によって決まる 0 から 1 の値であり、ユーザの興味の対象の推移を反映する。

4.3 考察

提案する手法によって実際にユーザオントロジーを構築し、構築されたユーザオントロジーについて考察を行った。以下にその結果について記す。

4.3.1 ユーザオントロジーの構築

ユーザオントロジーの構築手法や、ユーザオントロジーから得られる情報の妥当性について検討するために、実際に書籍を対象としてユーザオントロジーを作成した。利用したデータは、Amazon (<http://www.amazon.co.jp/>) で一人のユーザが書評を書いている書籍データで、そのデータを用いて当該ユーザのユーザオントロジーを構築した。また、書籍の属性データとしてはタイトル、著者、出版社、版型、価格、ジャンル、出版日、ISBN の 8 つの属性データを仮定した。

構築されたユーザオントロジーの例を以下に記す。括弧内の数字は、そのカテゴリにカテゴリ化されている書籍の数である。

```
top(book)
|--genre:文芸 (0)
|  |--type:ハードカバー (10)
|  |  |--pub:新潮社 (7)
|  |  |--pub:白水社 (8)
|  |  |--pub:角川書店 (3)
|  |
|  |--type:文庫 (8)
|  |  |--pub:早川書房 (6)
|  |  |--pub:新潮社 (4)
|  |
|  |--type:新書 (1)
|  |  |--白水社 (4)
|  |
|--type:ハードカバー (6)
```

次に表 1 に、3 人の評者のユーザオントロジーについて、(a) 書評を書いている書籍の数、(b) 構築されたオントロジーのカテゴリの数、(c) ルートカテゴリから末端のカテゴリまでの距離の最大値、(d) 同じカテゴリにカテゴリ化されるリソースの最大の数、(e) 同じく最小の数を示す。

表 1: 構築されたユーザオントロジー

	(a)	(b)	(c)	(d)	(e)
評者 A	26	12	4	9	0
評者 B	27	10	4	4	1
評者 C	57	11	3	10	0

4.3.2 検証

本節では、4.3.1節で記したようにして構築したユーザオントロジーについて、作成されたカテゴリとその制約となっている属性などの観点から検証を行う。

1. リソースの属性とカテゴリ

上記の例から分かるようにカテゴリの制約となった属性はジャンル、版型、出版社である。他のユーザの例では、著者と価格に関するカテゴリが作成されている例があった。一方、ISBN、タイトル、出版日については作成されていない。

個々の属性について見てみると、ISBNは書籍に対して一意に決定される属性値であり、カテゴリが作成されないことは妥当である。また、直観的にも反することはない。タイトルは定義的に一意性が保証されているものではないが、特定の書籍に対して付与されるものであることを考えるとほぼ一意性があり、ISBNと同様にカテゴリとして機能しないことは妥当であると思われる。

ジャンル、出版社、版型、著者についてカテゴリが作られることは直観に反さず、妥当であると考えられる。一般的な書店でもこれらの属性による分類が多く採用されている。一方、価格、出版日に基いて分類するのは例外的と思われる。今回の例では出版日に関するカテゴリは今回の例では作成されていないが、作成される可能性はある。

また、表1を参照するとリソースの数は、作成されるカテゴリの数などにあまり影響のないことが理解できる。ただし、特定のサイトの書評として公開しているというコンテキストが影響して、評者自身が特定のテーマを持っている場合も考えられる。例えば評者Aは26冊中18冊がコンピュータに関するものなど、特定の傾向を示していると思われる例もある。

2. 知識体系としての妥当性

上で述べたようにタイトルに関するカテゴリが作成される例は珍しい。しかし、一般的には、ユーザが任意の書籍について有している知識として第一に考えられる情報の一つである。同様に著者とジャンルについては知識を持っている場合が多いと考えられる。

ユーザオントロジーの木構造はリソースの属性情報に基づく分類であり、あるリソースが特定のカテゴリにカテゴリライズされている場合、そのカテゴリの制約となっている属性については意識的、もしくは無意識的に既知である、と考えるのが妥当である。

したがって、ユーザオントロジーからは、当該のユーザがカテゴリの制約となっている属性について知っているか判断できる一方で、制約となっていないような属性については知っているとも知らないとも判断することはできない。

以上1と2を考慮すると、書籍のタイトルのようにユーザが知識を持つにも関わらずオントロジー上に表れない属性が存在する一方で、価格のように知識として持ちにくい属性がカテゴリの制約として存在する場合があります。ユーザオントロジーをユーザの知識を示すプロフィールとして扱う場合、適切な情報が得られるように留意する必要がある。

3. 対象リソースとしての書籍の妥当性

A3フレームワークでは、ユーザオントロジーの構築、およびリソースの選択を行う際に、ユーザオントロジーの中のリソースの数に基く操作が行われる。

書籍は膨大な数の作品が出版されており、個人のユーザでもかなりの数の書籍を所有したり、読んだりするということが考えられる対象である。それにともないユーザの知識としてユーザオントロジーに加えられる数も大きくなるのが容易に考えられる。音楽CDなども同様のリソースである。

このようなリソースを対象とした場合、(1)式の n_p や(2)式の n_{p_k} が大きくなってしまい、式の有効性が失われる可能性がある。

他の問題としてリソース選択におけるコンテキストの影響が考えられる。

4. コンテキストの影響

リソースの選択は(2)式に基づいて候補となるリソースに重み付けを行い、その値の大きなものをユーザが関心を持つリソースとして提示する、という手法を採っている。(2)式はユーザがこれまで関心を持ってきたリソースの持つ属性に対して、より多くのリソースが持つ属性に大きな重みを与えるものである。

しかし、実際にはユーザのいるコンテキストによって重視する属性は変化する。例えば軽さといった実用性を重視している場合、対象となるリソースの形状などは比較的軽視されるような場合がある。この場合、当該ユーザのユーザオントロジーの構造に関わらず、重量といった属性を重視した重み付けを行う必要が生じる。

4.3.3 A3 フレームワークにおけるユーザオントロジーの操作

A3 フレームワークは、適応のために必要な機能をウェブシステムに提供する汎用的なフレームワークである。すなわち、どのようなウェブシステムが構築されるかを予め知ることはできず、どのようなリソースについて扱う必要があるか分からない。しかし、全てのリソースについてどのような属性が存在するかを知ることではできない。また、どの属性をユーザオントロジー構築の際にカテゴリの制約として良いのか、カテゴリの制約とはならないがユーザが知識を持つと考えられるものはどれか、ということを手前に定義することも不可能である。

この問題に対する対処としては大きく次の2つが考えられる。

1. ウェブシステムの作成者が、そのシステムで扱うリソースの属性について明示的に記述する。
2. ウェブ上の情報などを利用して推定する。

1は書籍のISBNのような定義的なものに対して有効である。しかし、ウェブシステムの作成者に負荷を強い点と、作成者の間で一貫性をどのようにして保つか、という点が問題となる。

2はウェブシステムの作成者の負荷を軽減するという点で有益である。しかし、4.3.2節の4で述べたように、コンテキストの影響などを考えると、任意の時点で動的に推定するといった条件も必要となってくる。

5 関連研究

ユーザモデリングに関する研究として[11, 15, 18]などがある。これらは一般的なモデリングに関する話題を扱ったものであったり、そこでセマンティックウェブの技術を利用する際の利点を述べたものである。A3フレームワークの特徴の一つである構築したプロフィール(ユーザオントロジー)を、ウェブシステムからどのように利用するか、あるいはその手段を提供する、といった観点から行われている研究ではない。

また、オントロジーのマージやマッピングなどの操作に関する研究として[12, 13, 20]などがある。これらの研究は、オントロジーの作成そのものにあり、どのようにしてマッピングやマージを行い、オントロジーをどのように共通の概念体系として扱うか、という観点からの研究である。これらはユーザのプロファイル

として、最初から一つのオントロジーを構築しようというA3のポリシーとは異なったものである。

セマンティックウェブ[2]は、ウェブ上の様々なコンテンツにメタデータを付与し、計算機に意味的な処理を行わせることを目的とした試みである。RDFやOWLは可読性と情報の共有を目的として提案されている、コンテンツやリソースの記述のための記述法である。しかし、書き方を提供するだけでは、その記法が守られるかどうかは保証できず、また、記述される内容の正確性や一貫性は保証されない。一方、A3フレームワークでは、UOMで一意にユーザオントロジーを構築することで、機構的にそれを保証している。

6 おわりに

近年の情報の増加やネットワークの発展を背景にして、ユーザ適応技術への要求が強くなっている。そのような要求に対して、我々はウェブシステムに適応に必要な機能を提供するためのフレームワークであるA3(Adaptation Anywhere & Anytime)フレームワークを提案してきた。このフレームワークでは、ユーザオントロジーと呼ばれるオントロジーを、ユーザプロフィールとして利用している。

ユーザオントロジーは、ユーザがウェブから得たリソースを分類した知識体系として定義されており、ユーザとA3フレームワーク上に構築されたウェブシステムとのインタラクションに基づいて動的に構築される。ユーザオントロジーの構築はリソースの属性情報に基づいて行われるが、あるリソースについてユーザが全ての属性情報を知っているということは考えにくいなど、どの属性を分類に用いて良いか、といった点が問題となる。本研究では、実際にユーザオントロジーを構築し、リソースと属性の関係、コンテキストの影響などについて考察を行った。

今後は考察の結果に基づき、動的に利用する属性を決定する手法の開発や、リソース選択におけるユーザオントロジーの利用についての改良などを行っていく必要がある。

参考文献

- [1] Beckett, D., RDF/SML Syntax Specification, <http://www.w3c.org/TR/rdf-syntax-grammar/>
- [2] Berners-Lee, T., Hendler, J., Lassile, O., The Semantic Web, Scientific American(2001)

- [3] Brickley,D.,Guha,R.V., RDF Vocabulay Description Language 1.0:RDF Schema, <http://www.w3.org/TR/rdf-schema/>
- [4] Chafee,J.,Gauch,S., Personal Ontologies for Web Navigation, In Proc. of CIKM'00(2000)
- [5] Chan,H.,Finin,T., An ontology for context-aware pervasive computing environments, Special Issue on Ontologies for Distributed Systems, Knowledge Engineering Review (2003)
- [6] Chen,C.,Chen,M.,Sun,Y.,PVA:A Self-Adaptive Personal View Agent, Journal of Intelligent Information System(2002)
- [7] Chen,L.,Sycara,K., A personal agent for browsing and searching, In Proc. 2nd Intl. Conf. Autonomous Agents(1998)
- [8] Clark,J., XSL Transformations(XSLT) Version1.0, <http://www.w3.org/TR/xslt>, 1999
- [9] Coutaz,J., Rey,G., Foundations for a theory of Contextors. Proc. of Computer-Aided Design of User Interfaces III, J. Vanderdonckt, C. Kolski Eds., Kluwer Academic Publ., pp. 13-32, 2002
- [10] Dean,M., Connolly,D., Harmelen,F., Hendler,J., Horrocks,I., McGuinness,D., Schneider,P., Stein,L., OWL Web Ontology Language 1.0 Reference, <http://w3.org/TR/owl-ref/>
- [11] Dolog,P.,Nejdl,W., Challenges and Benefits of the Semantic Web for User Modeling, Adaptive Hypermedia and Adaptive Web-Based System AH2003(2003)
- [12] Fridman,N.,Muse,M.A., An algorithm for merging and aligning ontologies:automation and tool support, In Proc. of the Workshop on Ontology Management at the Sixteenth National Conference on Artificial Intelligence(AAAI-99), AAAI Press(1999)
- [13] Fridman,N.,Muse,M.A., SMART:Automated Support for Ontology Merging and Alignment, In 12th Workshop on Knowledge Acquisition, Modeling and Management(1999)
- [14] Grimnes,G.AA., Chalmers,S., Edwards,P., Preece,A., 'Granitenights - a multi-agent visit scheduler utilizing semantic web technology' Seventh International Workshop on Cooperative Information Agents(2003)
- [15] Henze,N.,Nejdl,W., Knowledge Modeling for Open Adaptive Hypermedia, In Proc. of Adaptive Hypermedia and Adaptive Web-Based System AH2002(2002)
- [16] Kanjo,D.,Kawai,Y.,Tanaka,K., Ontology based Framework for Adaptive Web System, Workshop I:Enterprise Modeling and Ontology: Ingredients for Interoperability, in Intl. Conf. of Practical Aspect of Knowledge Management(2004)
- [17] Kay,J.,Kummerfeld,B.,Lauder,P., Personis: A Server for User Models, In Proc. of Adaptive Hypermedia and Adaptive Web-Based System AH2002,pp203-212(2002)
- [18] Kobsa.A., Generic User Modeling Systems, User Modeling and User-Adapted Interaction 11,pp49-63(2001)
- [19] Kurki,T.,Jokela,S.,Sulonen,R.,Turpeinen,M., Agents in delivering personalized content based on semantic metadata, In Proc. 1999 AAAI Spring Symposium Workshop on Intelligent Agents in Cyberspace,pp84-93(1999)
- [20] Noy,N.F.,Musen,M.A., The PROMPT suite: interactive tools for ontology merging and mapping. International Journal of Human-Computer Studies, 59(6),2003
- [21] Pretschner,A., Gauch,S., Ontology Based Personalized Search, In Proc. 11th Intl. Conf. on Tools with Artificial Intelligence (1999)
- [22] Extensible Markup Language (XML), <http://www.w3c.org/XML/>
- [23] Tanudjaja F., Mui L., Persona: A contextualized and personalized Web search, In Proceedings of the 35 Annual Hawaii International Conference on System Sciences(2002)