

特集 「Semantic Web とその周辺」

# DAML プログラムと Semantic Web

よりオントロジカルな世界へ

The DAML Program and the Semantic Web  
Toward a More Ontological World

益岡 竜介  
Ryusuke Masuoka

米国富士通研究所  
Fujitsu Laboratories of America, Inc.  
rmasuoka@fla.fujitsu.com, <http://www.flacp.fujitsulabs.com/~rmasuoka>

**Keywords:** semantic web, DAML, DAML + OIL, DAML-S, semantic integration.

## 1. はじめに

DAML (DARPA Agent Markup Language)<sup>\*1</sup> プログラム [DAML] は、オントロジー言語と対応したツールを開発し、Semantic Web の考えを実現していこうというものである。この DAML プログラムは米国国防省の研究開発機関である Defense Advanced Research Projects Agency (DARPA) [DARPA] のプログラムである。1999 年に小さなグループで検討を開始し、2000 年 8 月に正式に始まり、終了予定は 2003 年 9 月である。プログラムのサイズは年間約 2,000 万ドル、3 年間で約 6,000 万ドルである。参加グループは、15 ~ 20 グループ<sup>\*2</sup> あり、約 150 人の研究者が参加している。

DAML プログラムの当初のプロジェクトマネージャである James Hendler の強力なリーダーシップにより、現在までに Semantic Web の一つの基礎となるべきオントロジー言語 DAML + OIL [DAML + OIL] を定義し、それに対応した多くのツール [DAML Tools] やアプリケーション [DAML Applications] をつくり上げている。

最近 (2001 年秋) の DAML プログラムの変更としては、プロジェクトマネージャが James Hendler より Murray Burke になったこと、また DARPA の中での DAML プログラムの担当部署が ITO (Information Technology Office) [ITO] より、IXO (Information Exploitation Office) [IXO] になったことがある。後者の影響としては、想定アプリケーションの重点が調査活動 (Intelligence) から作戦活動 (Operation (兵站など)) に移されるなどが考えられる。

研究者間の交流は主に、活発なやり取りがあるメーリングリスト、約半年に一回行われる PI (Principal Investigator) Meeting [DAML Meetings] などを通じて行われる。もちろんそのほかにも、Semantic Web 関係の学会参加や個人的な交流などによっても行われている。

DAML プログラムは Tim Berners-Lee (TBL) が提案する Semantic Web [Port 02, TBL 99, TBL 01, W3C SW] の実現に向けての重要な要素である。TBL が引っ張る W3C で行われていく標準化活動が Semantic Web という車の車輪の一つであるとすると、DAML プログラムにおける、言語をつくり、実際に動くツールやアプリケーションをつくっていく活動は、もう一つの車輪に相当しよう。

本解説では、その DAML プログラムについての紹介を行い、さらに Semantic Web とともに DAML プログラムを、よりオントロジカルな方向に向かっている世界の中に位置づける。

## 2. DAML + OIL の概要

本章では、DAML プログラムの目玉というべき Semantic Web オントロジー言語である DAML + OIL [DAML + OIL] について、その概要を紹介する。DAML + OIL は W3C の Resource Description Framework (RDF) RDF Schema (RDFS) [RDF/RDFS] の基礎の上に実現された Semantic Web オントロジー言語である。当初、DAML プログラム独自の DAML-ONT [DAML-ONT] として 2000 年 10 月に公開されたが、主にヨーロッパで研究開発されていた Ontology Inference Layer (OIL) [OIL] と重なる部分が大きいため、それらを統合していこうという動きが出た。US/EU ad hoc Joint Committee on Agent Markup Languages [Joint Committee] がつくり、その成果が 2000 年 12 月に DAML + OIL という形で公開された。2002 年 5 月現在の DAML + OIL の最新版は 2001 年 3 月のものとなる。

\*1 DAML は daml (a は、black の a)、または dama (最初の a は、black の a、後の a は mustard の u) と発音されることが多い。

\*2 時によって前後する。2002 年 5 月現在、[DAML Researchers] によると 19 グループである。

現在この DAML+OIL は、DAML プログラムの最初のプログラムマネージャだった James Hendler も Co-chair をつとめる W3C の Web-Ontology (WebOnt) Working Group [WebOnto] に提出されている。この動き (DAML+OIL の W3C への提出) の理由の一つは、オープンに行っていることとはいえ、やはり DARPA は軍の研究組織であることにある。Semantic Web オントロジー言語は広く使われてこそ意味があるが、DARPA のものであると将来にわたって著作権などの問題が生じないとは言い切れない。企業などが安心して使えるように、一度 W3C へ提出することによりそういった問題を避ける目的がある。これを機会に DAML+OIL 言語自体は、WebOnto の Ontology Web Language (OWL) としてさらなる展開をしていくものと考えられる。より使いやすいように機能をレイヤに分け、必要に応じて機能を追加していくことになる。もちろん上記の DAML プロジェクトの Joint Committee でも、引き続きルールおよび問合せ言語などの検討を進めている。

DAML+OIL の Web オントロジー言語における位置づけは [Gómez-Pérez 02] に詳しい。DAML+OIL は Semantic Web オントロジー言語の中では表現力がかなり高い。Cyc [Cycorp] や Ontolingua [Ontolingua] のように、First Order Logic (FOL) などによる知識や公理を自由自在に書けるところまでとはいかないが、RDF/RDFS などにはない、オントロジーでよく使われる要素が多く追加されている。

### 3. よりオントロジカルな方向性

本章ではよりオントロジカルな方向性について議論し、その中に Semantic Web や DAML プログラムの活動を位置づける。

オントロジーにはいろいろ定義はあるが、よく引用されるのが Gruber の論文 [Ontolingua] からの以下である\*3。

“An ontology is a formal explicit specification of a shared conceptualization.”

一般に身体をもたないソフトウェアにとって、ある概念の理解を例えば特定の身体的な経験に基づかせることは難しい。そのようなソフトウェアは概念間の関係性の知識をもつことにより、情報の処理やメッセージのやり取りを可能にする。この概念間の関係性の知識をオントロジーと呼ぶ。

ここで概念間の関係性の知識を「より多く」あるいは「より豊かに」もつソフトウェアなどのシステムを「よりオントロジカルな」システムと定義する。

一つのソフトウェアのシステムがあれば、多少暗黙的かもしれないが、そこには必ず概念間の関係性の知識としてのオントロジーがある。それはシステムの作成者がそのシステムが対象とする問題を抽象化する行為と、その結果としてのオントロジーが不可分であるからである。

そしてその内在するオントロジーにより、機械やソフトウェアなどのシステムは情報の意味を理解する。

上記のように書くと機械が本当に意味を理解するのかといった哲学的議論になってしまいかねないが、ここでは「意味の理解」に操作的な定義を与える。ここで情報の「意味の理解」とは、その情報を「別の状況で使える」こととする。「意味をよりよく理解している」とは、情報を「より多くの異なる状況で使える」こととする。

システムが何もなく魔法のように意味の理解ができるわけではなく、何らかの clue や glue\*4 となるものが必要である。それがオントロジーであり、意味 (semantics) と呼ぶべきものでもある。よりオントロジカルなシステムはいろいろな情報の意味をよりよく理解することができる\*5。

そしてシステムが情報の意味をよりよく理解するならば、より少ない人間の関与で、情報を統合したり、情報を有効活用できる場合が増える。人間の関与が少なくて済めば、作業に対するコストを下げたり、(実際的に) 可能な作業の地平を広げることができる。

例えばネットワーク上の膨大な XML 文書のリソースを対象に探したり、操作したりすることを考える。それらの作業を行うシステムに XML 文書の (概念に対応する) タグの間の関係性の知識がより多くあれば、より多くの異なる状況でそれぞれの XML 文書にある情報を使うことができ、作業に関係する人間がその関与をより少なくすることができる。それにより作業に必要なコストを大きく下げたり、今までできなかったような検索を可能にできる。

コンピュータによるシステムができたときからそこにはオントロジーが存在し、世界はずっとよりオントロジカルな方向に向かってきた\*6。しかし Internet や Web の到来により、システムをよりオントロジカルにする必要性がさらに強くなった。よりオープンになり、より多種多様なシステムとやり取りするために、それぞれのシステムが情報の意味をよりよく理解する必要があるからである。

上記のようなオントロジカルな方向性の中で、Semantic Web は Web の枠組みの中で機械やシステムをよりオントロジカルにし、情報の意味をよりよく理解

\*4 clue は理解のためのヒントのようなもの、glue はいろいろな情報を結びつけるものといった意味で使っている。

\*5 もちろんもっているオントロジーをシステムがちゃんと使えることが前提である。

\*6 例えば、プログラミング言語やプロセス間通信の高度化がある。

\*3 オントロジーの定義については [FIPA 98 Part 12] の Annex A “Ontologies and Conceptualizations” も参考になる。

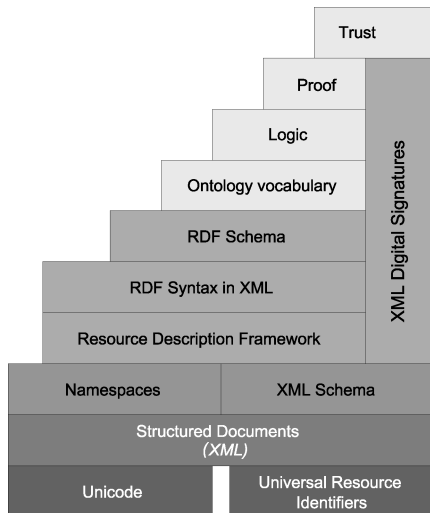


図1 Tim Berners-Lee の Semantic Web のビジョン。URI, Unicode, XML, Namespaces, XML Schema の上に構築される RDF, RDFS, Ontology vocabulary, Logic, Proof, Trust などの要素を含む。DAML のプロジェクトマネージャであったときの James Hendler との議論の中から生まれたそうである。別名 “the layer cake” とも呼ばれる

することを可能にすることを目指している。

TBL による図1 [TBL 01] は、Web の枠組みの中で、下の段の基礎の上に（概念を表す）タグ間のより豊かな関係性を記述する手段を積み上げていくことによって、よりオントロジカルな世界への向かう一つの道を示している。そして DAML は、“Ontology vocabulary” の層とそれより上の層を中心に、よりオントロジカルなシステムの実現を目指している。

図1の各層の概念的な説明は [浦本 01] に詳しいのでそれを参照された。ここでは次の4章で Ontology vocabulary の一つの実現である DAML + OIL も含めそれぞれの層で何を実現できるのかを具体的に説明していく。

#### 4. XML から DAML + OIL の先へ

本章では、XML, RDF, RDFS, DAML + OIL, DAML + OIL の先と順を追って、それぞれオントロジーのどのような要素を実現するのかを、またそれぞれ何を可能にするのかを具体的に簡単な例により、説明する。

##### 4.1 XML

HyperText Markup Language (HTML) では、タグは文書の要素あるいは文書の表示方法を表す特定のものしか使うことはできなかった。

eXtensible Markup Language (XML) [XML] の出現で自由にタグを使えるようになり、それらのタグにより、自己記述的な XML 文書を書くことが可能になった。XML が出た当初、XML は HTML を置き換えるのだと主張されていたときもあった。しかし明らかに、XML は人と機械（主にコンピュータ）の間で使われるよりも、

機械やソフトウェアたちが扱うデータ、あるいはそれらの間でのやり取りされる構造化された情報やメッセージのフォーマットとしての大きな可能性を見いだした\*7。S 式などほかにもいろいろあり得る文法レベルが、自己記述的で Web と整合性の高い XML に統一しつつあることは Semantic Web の基礎を固めるということであり、意味が大きい。

しかし XML の成功に伴い、個々のドメインごとに雨後の筈のごとく、??ML といったものが現れている。各分野で XML に基づいた ??ML が出てくること自体はその分野での情報の相互運用性が高まり、いいことではある。しかし、それだけでは TBL の夢見た、エージェントたちがネットワークからいろいろな情報を集め、統合し、ネットワーク上のリソースや人間とやり取りする Semantic Web の実現は不可能である。

なぜか？ XML を使ったとしても、情報の相互運用性を実現するためにはやり取りするものどうしが事前にやり取りされる XML 文書におけるタグやその意味、使われるモデルを取り決めておく必要がある。逆にいえば、これらの事前の取決めがなければ、機械にとって意味を理解することができない。

基本的にソフトウェアにとってどんなタグもそれだけでは意味をもたない。XML 文書で使われているタグは、“hasFather” だろうが、“K2C0007” であろうが、はたまた “Hello-World” だろうが、ソフトウェアにとっては関係ない。XML を使うと決めても、同じ情報を表す方法は、タグ名の選択、表現するためのモデルの使い方などで千差万別である。“William” の父が “Charles” である（ことを含む）情報の XML による記述の仕方には例えば以下のようなものが考えられる（もちろんこれらだけに限らない）。

```
<hasFather>
  <Person>William</Person>
  <Person>Charles</Person>
</hasFather>

<isFather>
  <Person>Charles</Person>
  <Person>William</Person>
</isFather>

<Person Father=Charles>
  William</Person>

<Family-Relationship>
  <Father>Charles</Father>
  <Mother>Diana</Mother>
  <Self>William</Self>
  <Brother>Henry</Brother>
</Family-Relationship>
```

\*7 特に、機械やソフトウェア間の構造化された情報のやり取りのための XML に基づく Simple Object Access Protocol (SOAP) [SOAP]、およびそのインタフェースを記述するための Web Service Description Language (WSDL) [WSDL] の可能性は大きい。

例えばネットワーク上のいろいろなリソースを使って “William” という人の祖先を探すことを考える\*8。たとえネットワーク上のいろいろなリソースがすべて XML をその情報の表現に使っていたとしても、上のように別々の表し方で表され、またヒントもなければ、人手を介さず機械だけでそれらの情報を統合して祖先の情報を探すことはできない。

#### 4.2 RDF

RDF/RDFS [RDF/RDFS] は TBL の Semantic Web に向けての次の一歩である。その中の RDF (Resource Description Framework) はメタデータの記述のための枠組みを与えるものである。Semantic Web の文脈では知識の記述の基礎を与えるものととらえられる。

RDF ではリソース、プロパティ、値の組による言明 (RDF Statement) を可能にする。例えば以下のようにして、「William の父が Charles である」ということを記述することができる\*9。

```
<rdf:Description about="William">
  <hasFather resource="Charles">
</rdf:Description>
```

ここで “William” がリソース、“hasFather” がプロパティ、“Charles” が値となる (図 2)。

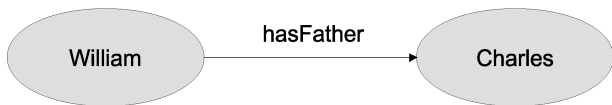


図 2 RDF の言明の図。RDF の言明はリソース、プロパティ、値の組で表される。ここでは “William” がリソース、“hasFather” がプロパティ、“Charles” が値となる。この言明は「William の父が Charles である」ことを表している

XML との違いは、RDF では単純ではあるが知識を表現するためのモデル (リソース、プロパティ、値の組) を固定したことである。XML の場合のように同じことでも表現が千差万別になることなく、どのように表現されるかについての枠 (Framework) を与えた。このことにより RDF の上に構成される多くのオントロジーの要素、すなわちタグ間の関係性に関する知識の表現の基礎を与えた。

しかし、まだ個々のタグの意味に関しては全く恣意的

である。したがって以下のような記述があったとしても、“William” の直系の祖先 (“hasAncestor”) としてソフトウェアが見つけれられるのは “Edward” しかない。

```
<rdf:Description about="William">
  <hasFather resource="Charles"/>
  <hasBrother resource="Henry"/>
  <hasAncestor resource="Edward"/>
</rdf:Description>
```

“William” の父 (“hasFather”) である “Charles” が直系の祖先であることは人間にとっては明白であるが、ソフトウェアにとっては付加的な情報を与えてやらない限り、理解できない。したがって、RDF だけでは、“William” の直系の祖先として、“hasFather” との正確なマッチの結果である “Edward” しか見つけることができないのである。

#### 4.3 RDFS

RDFS (RDF Schema) によって、より一般的な意味での本格的なオントロジーの記述が可能になる。枠組みとしては、オントロジーを記述した RDFS ファイルがあり、そのオントロジーに基づいて RDF による知識の記述がされているという形になる。

RDFS によって、概念 (あるいはタグ) 間の最も基本的な関係であるクラスやインスタンス、部分クラス、プロパティの定義域 (domain)、値域 (range)、タイプなどが表現できるようになった。

例えば以下の RDFS による記述は、クラス “Person” や “Female” がクラス “Animal” の部分クラス、プロパティ “hasMother” は “Animal” が domain で “Female” が range であることを表している。

```
<rdfs:Class ID="Animal">
</rdfs:Class>

<rdfs:Class ID="Person">
  <rdfs:subClassOf resource="#Animal"/>
</rdfs:Class>

<rdfs:Class ID="Female">
  <rdfs:subClassOf resource="#Animal"/>
</rdfs:Class>

<rdf:Description ID="hasMother">
  <rdfs:type resource="rdf:Property"/>
  <rdfs:domain resource="#Animal"/>
  <rdfs:range resource="#Female"/>
</rdf:Description>
```

タグに関するこういったメタな知識が得られれば、これらのタグによって記述されている知識を、より多くの別のところで使えるようになる。

以下のような三人の人に関する記述がネットワーク上にそれぞれ別々に分散してあることを考えよう (これらの情報は図 3 としてまとめた)。

\*8 この例を以下の説明に使っていく。これは本質を説明するための思考実験、簡単な例であり、かなり単純化している。実際にこれを実現するためには、さらにいろいろなことを考慮し、実現していく必要がある。

\*9 ここおよび以下の例では簡単のため、関係する部分だけ取り出している。特に XML の名前空間 (Namespaces) [XML Namespaces] に関して省略している。名前空間は Semantic Web のスケーラビリティを実現する重要な要素であり、図 1 の一つの要素でもあるが、本解説ではこれ以上詳しく説明しない。

```
<Person ID="William">
  <hasFather resource="Charles"/>
  <hasBrother resource="Henry"/>
  <hasAncestor resource="Edward"/>
</Person>
```

```
<Person ID="Elizabeth">
  <hasChild resource="Charles"/>
  <hasSister resource="Margaret"/>
</Person>
```

```
<Person ID="Margaret">
  <hasFather resource="George"/>
  <hasSister resource="Elizabeth"/>
</Person>
```

さらに RDFS による以下の知識 (オントロジー) があつたとする .

```
<rdf:Description ID="hasAncestor">
  <rdfs:type resource="rdf:Property"/>
</rdf:Description>

<rdf:Description ID="hasParent">
  <rdfs:type resource="rdf:Property"/>
  <rdfs:subPropertyOf
    rdf:resource="#hasAncestor"/>
</rdf:Description>

<rdf:Description ID="hasFather">
  <rdfs:type resource="rdf:Property"/>
  <rdfs:subPropertyOf
    rdf:resource="#hasParent"/>
  <rdfs:domain resource="#Animal"/>
  <rdfs:range resource="#Male"/>
</rdf:Description>
```

これらの情報から, William の直系の祖先 (リソース “William” に対してプロパティ “hasAncestor” の値になり得る人) を探すことを考えよう .

すると, “hasParent” が “hasAncestor” の, “hasFather” が “hasParent” の部分プロパティであることから “William” の父である “Charles” が, “William” の直系の祖先であることを推論できるようになる . また祖先探しとは直接は関係ないが, “hasFather” の値となっている “Charles” が “Male” であることを推論することも可能である .

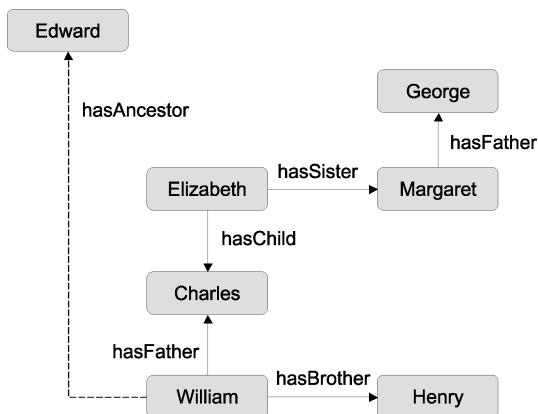


図3 例で使われている William の家系図

#### 4.4 DAML + OIL

DAML + OIL [DAML + OIL] は TBL による図 1 の “Ontology vocabulary” の層の実現を目指してつくられた .

XML, RDF, RDFS の基礎の上につくられた DAML + OIL は RDFS によるタグ間の関係性の記述に加えて, クラスやプロパティなどに関してさらに多様な関係性が表現できる . 例えばクラスについては, あるクラスが, いくつかのクラスの共通部分である (“intersectionOf”) こと, 互いに共通部分のないクラスのユニオンである (“disjointUnionOf”) ことなどを表現できる . またプロパティに関しては, 取り得る値の数 (“cardinality”), あるプロパティの逆 (“inverseOf”) であること, 推移的である (“Transitive-Property”) ことなどを表現できる .

これらによって RDFS と何が違ってくるのか? タグ間のより深い関係性を与えられることにより, より高い機能を効率的に提供できる可能性がある . 例えば, 前にあげた “William” の直系の祖先探しについてもう一度考えてみる . 今までの情報に加えて, DAML + OIL による以下のようなオントロジーが与えられているとする<sup>\*10</sup> .

```
<daml:Class rdf:about="#Animal">
  ...
  <rdfs:subClassOf>
    <daml:Restriction
      daml:cardinality="2">
        <daml:onProperty
          rdf:resource="#hasParent"/>
        </daml:Restriction>
    </rdfs:subClassOf>
  </daml:Class>

<daml:ObjectProperty
  rdf:ID="hasChild">
  <daml:inverseOf
    rdf:resource="#hasParent"/>
</daml:ObjectProperty>

<daml:TransitiveProperty
  rdf:ID="hasAncestor">
  <rdfs:label>hasAncestor</rdfs:label>
</daml:TransitiveProperty>
```

このオントロジーが与える clue や glue により, 直系の祖先を探す際により多くのヒントを使うことができる . 例えば以下のようなことが可能である .

- “hasChild” が “hasParent” の逆のプロパティ (“inverseOf”) であること, (RDFS の際に出してきた) “hasParent” が “hasAncestor” の部分プロパティ (“subPropertyOf”) であること, および “hasAncestor” が推移的 (“TransitiveProperty”) であることから, “Elizabeth” も “William” の直系の

\*10 DAML + OIL のオントロジー例 [DAML + OIL-Ex] を参考にした .

祖先であることがわかる。

- 祖先を探す途中で誰かの親を探す必要が生じた場合に、親は二人しかいないのだから (“hasParent” の “cardinality” が 2)、もし異なる二人を親として見つけたら、その人の親はその二人に限られるはずである。二人見つけた段階でその人の親の検索を終了することができる<sup>\*11</sup>。

#### 4.5 DAML + OIL の先

DAML + OIL の次に現在 Joint Committee で検討されている DAML-L (DAML-Logic の意味) は TBL による図 1 の “Logic” の層を実現することを目指している。これにより First Order Logic (FOL) などによる知識や公理を記述できるようになる。

DAML-L が実現されるとどのようなことが可能になるのか？ 今までの “William” の直系の祖先を探す例を再び用いる。例えば DAML-L を使えばさらに「ある人の兄弟姉妹の親は、その人自身の親である」といった知識を記述することができるようになる。その知識があれば、すでに直系の祖先であることがわかっている “Elizabeth” と “Margaret” の姉妹関係、そしてその “Margaret” と “George” の親子関係から、“William” の直系の祖先として “George” を見つけることができる。このことは DAML + OIL で記述できる範囲のオントロジでは見つけることができなかった。

TBL による図 1 の “Proof” と “Trust” の層はまだ手探りの段階であるが、どのようなことを目指しているのかを簡単に以下に述べる ([Swartz 01] 参照)。

図 1 の “Proof” の層では、なぜそういう結論に至ったのかの理由付けの記述手段を提供することを旨とする。理由付けを記述し、共有できるようにすることにより、結論を検証したり、理由付けをほかのシステムで再利用したりすることが可能になる。

XML Digital Signature などの電子署名の技術により、ある文書を特定の人を書いたことは確認できるようになる。しかし著者の特定ができて、その著者を知らなければ、それだけではその文書の信頼性をはかることはできない。図 1 の “Trust” の層では信用に関する情報の記述手段を提供することを目指す。Web によって、マスメディアを通さず、自分が全く知らない、ほかの通常の個人が書いた文書に直接アクセスすることが日常茶飯事になったが、“Trust” の層により信用に関する情報の共有が可能になり、直接知らない人を書いた文書でもその信頼性をはかる（少なくとも見当をつける）ことが可能になる。実生活でも行われている同様のことを “Web of Trust” ([TBL 99] Chap. 11 “Privacy” 参照)

として Web に導入し、(Semantic) Web の土台をより強固なものにする。

## 5. DAML ツール・アプリ・オントロジ

言語をいくらつくっても、それだけでは社会は変わらない。DAML プログラムの特徴は、DAML + OIL をもとに実際に使ってもらうことを目指してツール [DAML Tools]、アプリケーション [DAML Applications] をつくることが強力に推し進めている点である。また DAML Ontology Library [DAML Ontology Library] には DAML + OIL に基づいて記述された 170 以上のオントロジがある。これらのオントロジはそれぞれ実際のアプリケーションなどに使うことを考えてつくられており、重要な資産である。

DAML のツール・アプリケーション・オントロジについては、上にあげた参考 Web ページにそれぞれ詳しいリストがある。ぜひそれらのページからリンクをたどり、いろいろ探索してみることを薦める。

## 6. DAML の今後

2002 年 2 月の DAML PI Meeting の DAML の今後を検討するセッションでは、以下のような個別セッションが設定され、それぞれのセッションに分かれ、議論が行われた (詳細な内容は、[DAML 2002/02 PI Meeting] の Thursday, 14 February 2002 の Breakout Session および BOF の Outbrief を参照)。

DAML Experiment: DARPA や米国の軍関係者に成果を示すデモについて検討した。

DAML Tools: 現在つくられているツール、また求められているツールのリストをまとめた。

DAML-Services: DAML Services について、ほかのアクティビティとの関係や今後どのように広めていかなどについて検討した。このグループの成果である DAML-S [DAML-S] のバージョン 1.0 を 6 か月以内に出すことを決議した。

Security: DAML の Security として求められていることを検討し、必要となるオントロジ (Credentials, Security Technique, Policies, Deontic Concepts, Agents, Actions), シナリオ, ツールなどをあげた。

Time: 時間に関連したオントロジについて検討した。

ESG: 米国海軍の Expeditionary Sensor Grid (ESG) [ESG] プロジェクトに関係した DAML の活動についてまとめ、検討した。

Query: DAML Query Language (DQL) に求められることや最初の案を検討した。

Rules: DAML におけるルールの記述について検討した。

これらのうち Query や Rules で検討された内容が 4.5 節で述べた DAML-L になっていくものと考えられる。

\*11 もちろん見つけた結果の確認のために、さらに探し続けることはあり得る。

これらの中で特に注目すべきは DAML-Services およびそこで検討されている DAML-S である。DAML + OIL に基づく Web Service オントロジーである DAML-S は、プロパティや能力を記述するタグによる Web Service の記述によって、自動的なサービス発見、実行、統合、実行監視などを可能にする。

現在広がりつつある Web Services の今後の向かう先としては、Semantic Integration と呼ばれる、Web Services の動的な統合であろう。特に多種多様な情報を扱う個人環境においてその意味は大きい。DAML-S などによって、これが実現されていけば、TBL の考えるエージェントたちがネットワーク上の Web Service を含むリソースを動的にいろいろ組み合わせ、ユーザの要求を実現する Semantic Web の実現に向けて大きく前進する ([Preece 02] の Guest Editors 'Introduction 参照)。

## 7. ま と め

2000 年の秋に University of Maryland, Baltimore County (UMBC) の Tim Finin 教授の研究室に滞在する機会を得た。そのときちょうど研究室で DAML のプロジェクトを始める段階であり、Semantic Web や DAML の動向や活動を知ることになった。滞在中にはそのとき DAML プロジェクトマネージャであった James Hendler にも会うことができた。DAML プログラムは始まったばかりにもかかわらず、その時点まですでに Semantic Web オントロジー言語である DAML-ONT の第一版を決定しており、James Hendler は期限を決めてその DAML-ONT に基づくオントロジーやアプリケーションを各グループに求めるなど、強力かつ情熱的に DAML プログラムを推し進めており、DAML プログラムに大きな可能性を感じた。

今までいくつかのオントロジーの標準化のプロセスを見ることがあったが、多くは期待にはずれるものであった。オントロジー言語を決めるものとは少し異なり、標準の上位オントロジーを決めようという活動ではあるが、ANSI Ad Hoc Group on Ontology Standards [ANSI Ontology] あるいは IEEE Standard Upper Ontology (SUO) Working Group [IEEE SUO] などでは議論が発散してしまっていた。社会へのインパクトを与えるためには相互運用性が必要であり、ほかを捨てていずれか一つを選ぶ必要があるのだが、なかなか一つに決めることができない。もちろん一つ一つの議論は根本的な点を論じてはいるが、議論全体に実用性という基準が感じられなかった。

FIPA [FIPA] の "Ontology Service Specification" [FIPA 98 Part 12] は標準としてまとめたが、それが基礎を置くエージェントのインフラストラクチャとしての社会の受容がまだであるため、残念ながら大きな影響

を与えるに至っていない。

Internet という場において、Hypertext での研究は Web という形で、Information Retrieval における研究は Google [Google] などの Search Engines という形で、すでに社会のあり方を変えつつある。次は AI やオントロジーではないだろうか。浦本氏が語るように [Uramoto 01], "Semantic Web は、AI 研究者にとって自分たちの考えを実際に使ってもらうための大きな土俵になるだろう"。

Internet や Web の以前からも、TBL による図 1 のように、徐々に下の段の基礎の上に次の基礎を積み上げながら、よりオントロジカルな世界へと向かってきた。Internet や Web の登場によりオントロジカルな方向への圧力が高まり、Semantic Web が現時点での道として提示された。それらの層は何度か積み直すことがあるかもしれないが、やがて徐々により高く積み上がっていくであろう。

社会に影響を与えながら、それらの層をうまく積み上げていくためには、オントロジーの要素をバランスを見計らいながら、順番に導入していくことが重要である。また技術のスケラビリティも社会に受け入れられる重要な要素である。本解説中では述べなかったが、スケラビリティを実現していくためには [Stephens 01] のようなオントロジー整列の技術や、自然言語と Semantic Web を結びつける技術が重要になると考えている。前者のオントロジー整列は時間的、空間的に変化し続けるオントロジーたちとの関係性を与え、後者は自然言語と Semantic Web の間の関係性を与えるという意味で、これらも大きな意味でのオントロジーの一部として捉えられる。

また 6 章の最後に述べたように、Semantic Web や DAML が大きな影響を与え始めるのは、個人環境における Semantic Integration の実現からになると考えている。

最後になったが、DAML プログラムのような、社会に大きな影響を与え得る Semantic Web の実現に向けた研究がより多く現れるきっかけに本解説がなれたなら、幸いである。

## 参 考 文 献

- [ANSI Ontology] ANSI Ad Hoc Group on Ontology Standards, <http://www-ksl.stanford.edu/onto-std/>
- [TBL 99] Berners-Lee, Tim with Mark Fischetti: Weaving the Web, HarperBusiness (1999). See also, <http://www.w3.org/People/Berners-Lee/Weaving/Overview.html>
- [TBL 01] Berners-Lee, Tim, Hendler, James and Lassila, Ora: The Semantic Web, Scientific American, Vol. 284, No. 5, pp. 34-43 (May, 2001)
- [Cycorp] Cycorp, Inc., <http://www.cyc.com/>
- [DAML] DARPA Agent Markup Language (DAML) <http://www.daml.org/>
- [DAML Applications] DAML Applications, <http://www.daml.org/applications/>

- [DAML + OIL] DAML + OIL, <http://www.daml.org/2001/03/daml+oil-index.html>
- [DAML + OIL-Ex] DAML Example Ontology (daml + oil-ex), <http://www.daml.org/2001/03/daml+oil-ex>
- [DAML-ONT] DAML-ONT, <http://www.daml.org/2000/10/daml-ont.html>
- [DAML Ontology Library] DAML Ontology Library, <http://www.daml.org/ontologies/>
- [DAML Researchers] DAML Researchers, <http://www.daml.org/researchers.html>
- [DAML-S] DAML Services, <http://www.daml.org/services/>
- [DAML Tools] DAML Tools, <http://www.daml.org/tools/>
- [DAML 2002/02 PI Meeting] DAML February 2002 PI Meeting Agenda, <http://www.daml.org/meetings/2002/02/pi/agenda.html>
- [DARPA] Defense Advanced Research Projects Agency (DARPA), <http://www.darpa.mil/>
- [ITO] DARPA Information Technology Office (ITO), <http://www.darpa.mil/ito/>
- [IXO] DARPA Information Exploitation Office (IXO), <http://www.darpa.mil/ixo/>
- [ESG] Expeditionary Sensor Grid (ESG), <http://www.nwdc.navy.mil/Concepts/ESG.asp>
- [XML] eXtensible Markup Language (XML), <http://www.w3.org/XML/>
- [FIPA] Foundation for Intelligent Physical Agents (FIPA), <http://www.fipa.org/>
- [FIPA 98 Part 12] FIPA 98 Part 12 Version 1.0: Ontology Service Specification, <http://www.fipa.org/specs/fipa00006/> (次に日本語訳がある, <http://fipa.comtec.co.jp/fipatrans/>)
- [Gómez-Pérez 02] Gómez-Pérez, Asunción and Corcho, Oscar: Ontology Language for the Semantic Web, IEEE Intelligent Systems, pp. 54-60 (Jan/Feb, 2002)
- [Google] Google, <http://www.google.com/>
- [Ontolingua] Gruber, T. R: A Translation Approach to Portable Ontology Specifications, Knowledge Acquisition, Vol. 5, No. 2, pp. 199-220 (1993)
- [IEEE SUO] IEEE P1600.1 Standard Upper Ontology (SUO) Working Group, <http://suo.ieee.org/>
- [Joint Committee] Joint US/EU ad hoc Agent Markup Language Committee, <http://www.daml.org/committee/>
- [KSL] Knowledge Systems Laboratory (KSL), Stanford University, <http://www-ksl.stanford.edu/>
- [XML Namespaces] Namespaces in XML, <http://www.w3.org/TR/xml-names11/>
- [OIL] Ontology Inference Layer (OIL), <http://www.ontology.knowledge.org/oil>
- [DAML Meetings] Past DAML Meetings, <http://www.daml.org/meetings/>
- [Port 02] Port, Otis: The Next Web, Special Report, BusinessWeek, pp. 96-102 (March 4, 2002)
- [Preece 02] Preece, Alun and Decker, Stefan, eds.: Intelligent Web Services, IEEE Intelligent Systems, pp. 15-53 (Jan./Feb. 2002)
- [RDF/RDFS] Resource Description Language (RDF) RDFS Schema, <http://www.w3c.org/RDF/>
- [SOAP] Simple Object Access Protocol (SOAP), <http://www.w3.org/2002/ws/>
- [Stephens 01] Stephens, Larry M. and Huhns, Michael N.: Consensus Ontologies: Reconciling Web Pages and Agents, IEEE Internet Computing, Vol. 5, No. 5, pp. 90-95 (Sep./Oct. 2001)
- [Swartz 01] Swartz, Aaron and Hendler, James: The Semantic Web: A Network of Content for the Digital City, Proc. 2nd Annual Digital Cities Workshop, Kyoto, Japan (Oct. 2001)
- [浦本 01] 浦本直彦: Semantic Web 機械のための Web , 人工知能学会誌, Vol. 16, No. 3, pp. 412-419 (2001)
- [WebOnt] Web-Ontology (WebOnt) Working Group, <http://www.w3c.org/2001/sw/WebOnt/>
- [WSDL] Web Services Description Language (WSDL), <http://www.w3.org/TR/wsdl>
- [W3C SW] W3C Semantic Web, <http://www.w3c.org/2001/sw/>

2002年5月22日 受理

---

## 著者紹介

---

### 益岡 竜介



1962年生まれ。1985年東京大学理学部数学科卒業。1987年同大学院修士課程卒業。2000年同大学より数理科学博士号取得。博士(数理科学)。1988年(株)富士通研究所入社。1991年カーネギーメロン大学客員研究員。1993年富士通研究所に戻り、2001年より米国富士通研究所に移り、現在に至る。現在米国富士通研究所カレッジパーク研究所主任研究員およびメリーランド大学 Adjunct Professor。ニューラルネットワーク、エージェントなどの研究を経て、現在セマンティックウェブ、パーベシブコンピューティング、バイオインフォマティクス、インターネットセキュリティなどに興味をもって研究を進めている。