

TỔNG QUAN VỀ SEMANTIC WEB

Lê Hoài Bắc – Nguyễn Thanh Nghị

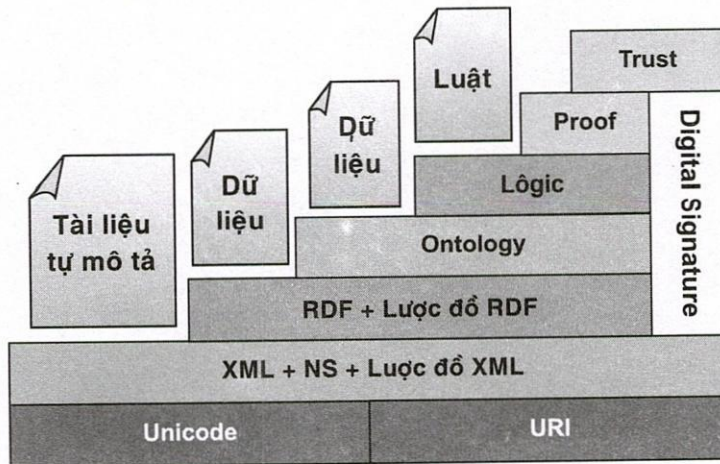
Khoa Công nghệ thông tin, Trường Đại học Khoa học Tự nhiên – ĐHQG-HCM

(Bài nhận ngày 30 tháng 7 năm 2003)

TÓM TẮT: Bài báo này trình bày về sự hình thành và phát triển của Semantic Web, các thành phần chính của Semantic Web gồm: RDF (Resource Description Framework), lược đồ RDF, OIL (Ontology Inference Layer), logic suy luận, chứng minh suy diễn của hệ thống thông qua việc liên kết các dữ kiện, và cách thể hiện sự tin cậy lẫn nhau giữa các hệ thống được minh họa ở tầng Trust. Phần cuối cùng trình bày một số ứng dụng trên môi trường Semantic Web.

1. Giới thiệu

Ban đầu Web được tạo ra để con người thao tác: đọc, hiểu, lấy thông tin,... và dù rằng máy tính có thể đọc được thông tin trên Web, nhưng nó không hiểu được các thông tin này. “Semantic Web là phần mở rộng của Web mà trong đó thông tin được xử lý một cách tự động bằng máy tính, làm tăng khả năng hợp tác làm việc giữa máy tính và con người” (Tim Berners Lee - cũng là người đề ra các chuẩn về WWW, URIs, HTTP, và HTML cho Web). Kiến trúc các tầng của Semantic Web được trình bày như hình 1. Trong đó, tầng Unicode, URI (Uniform Resource Identifier) và tầng XML + NS (Name Space) + Lược đồ XML là các tầng chuẩn của Web hiện nay, không phải mục tiêu trình bày trong bài báo này. Các tầng chính của Semantic Web được mô tả chi tiết trong phần 2.



Hình 1. Các tầng trên Semantic Web

Một hướng tiếp cận khác về Semantic Web của James Hendler là sử dụng ngôn ngữ đánh dấu cho các agent (DARPA Agent Markup Language - DAML) để chúng có thể hiểu và xử lý thông tin trên Web một cách tự động. Mặc dù DAML và Semantic Web được phát triển độc lập nhau nhưng mục tiêu là như nhau. Do đó, ta có thể nói DAML đang phát triển các công nghệ cho Semantic Web. Nhóm nghiên cứu DAML đang tìm hiểu rất kỹ lưỡng từ những khía cạnh khác nhau của RDF, lược đồ RDF, ontologies, suy luận logic... và các kết quả nghiên cứu của họ cũng đang được giám sát bởi W3C (W3C (World Wide Web Consortium) là một tổ chức chuyên chuẩn hóa các chuẩn giao tiếp trên internet).

2. Các tầng chính của Semantic Web

2.1 Tầng RDF + Luộc đồ RDF

Trong khi XML bị giới hạn trong việc mô tả tài liệu để máy có thể hiểu được thì RDF có thể giải quyết được giới hạn này. Giải pháp được đưa ra là dùng *siêu dữ liệu (metadata)* mô tả tài liệu trên Web để máy có thể hiểu được chúng. Siêu dữ liệu là một dạng dữ liệu dùng để mô tả dữ liệu khác. Hay nói cách khác siêu dữ liệu là những thông tin mô tả tài nguyên trên Web. Chẳng hạn như, sách là một loại tài nguyên trên Web (<http://www.amazon.com/CSharp.htm>), khi đó các thông tin mô tả cho tài nguyên này như: tên tác giả, tên tựa sách, ngày xuất bản, ... chính là siêu dữ liệu.

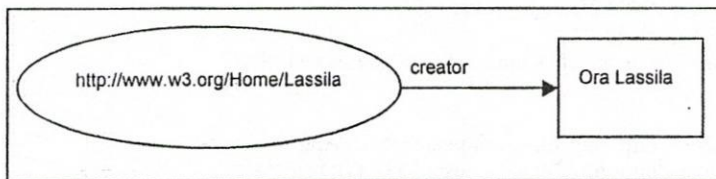
2.1.1 RDF (Resource Description Framework) [1, 2, 9, 10]

RDF là tầng nền tảng của *Semantic Web*, được đề xuất bởi tổ chức W3C. Nó cho phép trao đổi thông tin giữa các ứng dụng trên Web mà máy có thể hiểu được. Mô hình RDF và tập cú pháp cơ bản là hai thành phần chính của RDF để biểu diễn siêu dữ liệu.

Mô hình RDF cơ bản gồm ba đối tượng: *tài nguyên, tính chất, và phát biểu*. Tài nguyên là tất cả những gì được mô tả bằng biểu thức RDF. Nó có thể là một trang Web, cũng có thể là một phần của trang Web, hoặc là một tập các trang Web, *Tính chất* là quan hệ dùng để mô tả đặc tính của tài nguyên. *Phát biểu* gồm: chủ ngữ, vị ngữ, bổ ngữ. Phần chủ ngữ mô tả các tài nguyên trong phát biểu, thông thường là các địa chỉ URI. Vị ngữ là phần xác định tính chất của tài nguyên. Bổ ngữ là giá trị cụ thể nào đó hoặc có thể là một tài nguyên khác. Chẳng hạn như, phát biểu sau:

“Ora Lassila is the creator of <http://www.w3.org/Home/Lassila>”

sẽ được mô hình hóa thành mô hình RDF như hình 2. Trong đó phần chủ ngữ là tài nguyên “<http://www.w3.org/Home/Lassila>”, vị ngữ là “creator”, và bổ ngữ là “Ora Lassila”.



Hình 2. Mô hình RDF minh họa phát biểu trên.

Tập cú pháp cơ bản của RDF:

- [1] RDF ::= ['<rdf:RDF>'] description* ['</rdf:RDF>']
- [2] description ::= '<rdf:Description' idAboutAttr? '>' propertyElt* '</rdf:Description>'
- [3] idAboutAttr ::= idAttr | aboutAttr
- [4] aboutAttr ::= 'about=' URI-reference ''
- [5] idAttr ::= 'ID=' IDsymbol ''
- [6] propertyElt ::= '<' propName '>' value '</' propName '>' | '<' propName resourceAttr '>'
- [7] propName ::= QName
- [8] value ::= description | string
- [9] resourceAttr ::= 'resource=' tham chiếu URI ''
- [10] QName ::= [NSprefix '.'] name
- [11] URI-reference ::= string, interpreted per [URI]
- [12] IDsymbol ::= (bất kỳ ID nào hợp lệ nào của XML)
- [13] name ::= (bất kỳ tên hợp lệ nào của XML)
- [14] NSprefix ::= (bất kỳ tiếp đầu ngữ namespace hợp lệ nào)
- [15] string ::= (bất kỳ chuỗi nào)

Khi đó, phát biểu:

“Ora Lassila is the creator of <http://www.w3.org/Home/Lassila>”

sẽ được viết theo cú pháp RDF như sau:

```
<rdf:RDF
xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"

```

```

xmlns:s="http://description.vn/schema/"
<rdf:Description about="http://www.w3.org/Home/Lassila">
  <s:Creator>Ora Lassila</s:Creator>
</rdf:Description>
</rdf:RDF>

```

Các khái niệm nâng cao trong RDF là mô hình *container* và tập cú pháp nâng cao mô tả chúng. Do chúng ta thường tham chiếu đến một tập các tài nguyên. Chẳng hạn như, khi ta nói rằng công trình này được thực hiện bởi ông Nguyễn Văn A, Trần Ngọc B, ...; hoặc việc liệt kê danh sách các học viên của một khóa học đều là các thao tác lên một tập tài nguyên. RDF *container* được dùng để lưu danh sách các tài nguyên hoặc các kiểu giá trị cụ thể (chuỗi các ký tự, kí số, ...). Mô hình *container* có ba loại đối tượng sau: *bag*, *sequence*, *alternative*. *Bag* là danh sách không có thứ tự của các tài nguyên. Nó được dùng để khai báo thuộc tính có nhiều giá trị và không quan tâm đến thứ tự của những giá trị này. *Sequence* là danh sách có thứ tự của các tài nguyên và thứ tự những giá trị của các tài nguyên này cần được quan tâm đến. *Alternative* là một danh sách các tài nguyên hoặc các giá trị, được dùng để biểu diễn các giá trị lựa chọn của một thuộc tính.

Để biểu diễn một tập hợp các tài nguyên, RDF dùng một tài nguyên mới để xác định một tập tài nguyên cần được biểu diễn. Tài nguyên mới này phải được khai báo như là một thể hiện của một trong ba loại đối tượng *container* được đề cập trên và dùng tính chất *rdf:type* để khai báo loại đối tượng *container*. Mỗi quan hệ thành viên giữa *container* và các tài nguyên của tập hợp được xác định bằng những tên đơn giản như : “_1”, “_2”, “_3” Chẳng hạn như, xét phát biểu sau:

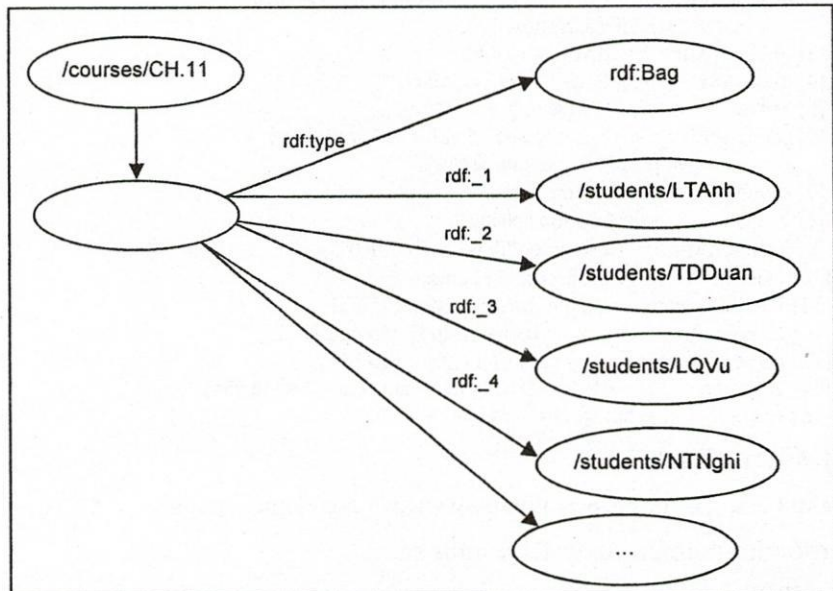
“The students in course CH.11 are LTAAnh, TDDuan, LQVu, NTNghi, ...”

có mô hình RDF *container* như hình 3 và được biểu diễn bằng cú pháp RDF *container* như sau:

```

<rdf:RDF
  xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
  xmlns:s="http://description.org/schema/"
  <rdf:Description about="http://www.is-edu.hcmuns.edu.vn/courses/CH.11">
    <s:students>
      <rdf:Bag>
        <rdf:li resource="http://mycollege.edu/students/LTAAnh"/>
        <rdf:li resource="http://mycollege.edu/students/TDDuan"/>
        <rdf:li resource="http://mycollege.edu/students/LQVu"/>
        <rdf:li resource="http://mycollege.edu/students/NTNghi"/>
        ...
      </rdf:Bag>
    </s:students>
  </rdf:Description>
</rdf:RDF>

```



Hình 3. Mô hình RDF *container* minh họa phát biểu trên.

2.1.2 Lược đồ RDF [2, 9, 10]

RDF là nền tảng cho việc xử lý siêu dữ liệu. RDF cho phép định nghĩa mô hình RDF mô tả các quan hệ giữa các tài nguyên thông qua các tính chất và giá trị của các tính chất. Tuy nhiên, RDF không cung cấp cơ chế để khai báo các tài nguyên, tính chất, và các quan hệ giữa chúng. Lược đồ RDF đảm nhiệm vai trò này, nó cho phép định nghĩa các tài nguyên, thuộc tính cũng như các quan hệ qua lại giữa chúng để thể hiện ngữ nghĩa.

Tương tự như lược đồ XML, nhưng lược đồ RDF cung cấp tập từ khóa cho phép người dùng định nghĩa bộ từ khóa cụ thể cho dữ liệu RDF, và cho phép thể hiện ngữ nghĩa của chúng thông qua việc định nghĩa các quan hệ giữa nó với các đối tượng liên quan đã được định nghĩa trước. Tập từ khóa của lược đồ RDF được chia thành các nhóm cơ bản sau: nhóm từ khóa liên quan đến các lớp, nhóm liên quan đến các thuộc tính, và các nhóm từ khóa khác.

Nhóm từ khóa liên quan các đến lớp (Core Classes): `rdfs:Resource`, `rdfs:Class`, `rd:Property`, Tất cả những gì được mô tả bằng biểu thức RDF được xem như một thể hiện của lớp `rdfs:Resource`. Lớp `rd:Property` đại diện cho tất cả các tính chất dùng để mô tả các đặc tính của lớp `rdfs:Resource`. Lớp `rdfs:Class` được dùng để định nghĩa các khái niệm trong lược đồ RDF. Chẳng hạn như, khái niệm “Nghệ sĩ” sẽ được định nghĩa như sau:

```
<rdfs:Class rdf:ID="Nghệ_sĩ">
  <rdfs:comment>Lớp nghệ sĩ</rdfs:comment>
</rdfs:Class>
```

Nhóm từ khóa liên quan đến các tính chất (Core Properties): `rd:type`, `rdfs:subClassOf`, `rdfs:subProperty`, Từ khóa `rd:type` được dùng để khai báo các thể hiện của một lớp. `rdfs:subClassOf` và `rdfs:subProperty` được dùng để định nghĩa một lớp hoặc tính chất là lớp con hoặc tính chất con của một lớp khác. Chẳng hạn như việc khai báo lớp “Nhạc sĩ” và lớp “Họa sĩ” là lớp con của lớp “Nghệ sĩ” sẽ được viết như sau:

```
<rdfs:Class rdf:ID="Nhạc_sĩ">
  <rdfs:subClassOf rdf:resource="#Nghệ_sĩ"/>
</rdfs:Class>

<rdfs:Class rdf:ID="Họa_sĩ">
  <rdfs:subClassOf rdf:resource="#Nghệ_sĩ"/>
</rdfs:Class>
```

Việc khai báo tính chất “Tạo ra” chỉ được áp dụng trên lớp “Nghệ sĩ” và thuộc về lớp “Tác phẩm nghệ thuật” được viết như sau:

```
<rd:Property rdf:ID="Tạo_ra">
  <rdfs:domain>Nghệ_sĩ</rdfs:domain>
  <rdfs:range>Tác_phẩm_nghệ_thuật</rdfs:range>
</rd:Property>
```

từ khóa `rd:domain` được dùng để chỉ ra lớp ở vị trí chủ ngữ mà tính chất đó được áp dụng, và `rd:range` được dùng khai báo lớp ở vị trí bổ ngữ mà tính chất thuộc về. Khi đó việc khai báo tính chất “Sáng tác” và “Vẽ” là tính chất con của tính chất “Tạo ra” được viết như sau:

```
<rdfs:Property rdf:ID="Sáng_tác">
  <rdfs:subPropertyOf rdf:resource="#Tạo_ra"/>
  <rdfs:domain>Nhạc_sĩ</rdfs:domain>
  <rdfs:range>Bài_hát</rdfs:range>
</rdfs:Property>

<rdfs:Property rdf:ID="Vẽ">
  <rdfs:subPropertyOf rdf:resource="#Tạo_ra"/>
  <rdfs:domain>Họa_sĩ</rdfs:domain>
  <rdfs:range>Bức_tranh</rdfs:range>
</rdfs:Property>
```

Ví dụ minh họa kết hợp ngữ nghĩa trên lược đồ RDF và các thể hiện của nó trong một ứng dụng cụ thể được minh họa qua hình 4.

Lược đồ ở hình 4 được thể hiện thông qua cú pháp của RDF như sau:

```
<rdf:RDF xml:lang="en"
  xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
  xmlns:rdfs="http://www.w3.org/TR/1999/PR-rdf-schema-19990303#">
```

```

<! Phần định nghĩa lớp !>

<rdfs:Class rdf:ID="Nghệ_sĩ">
  <rdfs:comment>Lớp nghệ sĩ</rdfs:comment>
</rdfs:Class>

<rdfs:Class rdf:ID="Nhạc_sĩ">
  <rdfs:subClassOf rdf:resource="#Nghệ_sĩ"/>
</rdfs:Class>

<rdfs:Class rdf:ID="Họa_sĩ">
  <rdfs:subClassOf rdf:resource="#Nghệ_sĩ"/>
</rdfs:Class>
...

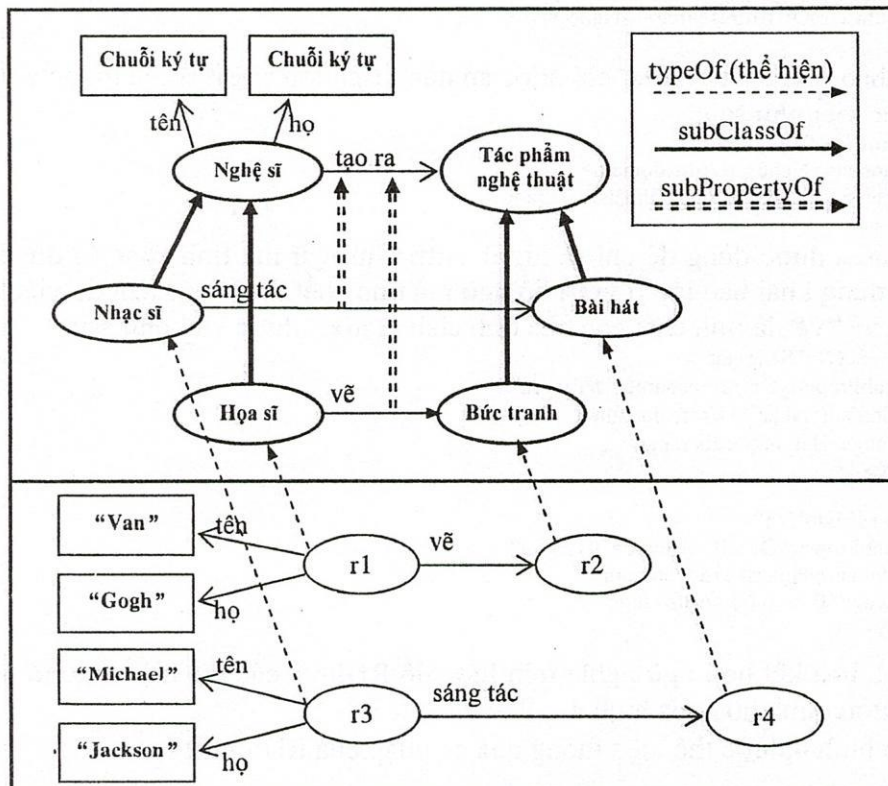
<! Phần định nghĩa các tính chất !>

<rdf:Property rdf:ID="Tạo_ra">
  <rdfs:domain>Nghệ_sĩ</rdfs:domain>
  <rdfs:range>Tác_phẩm_nghệ_thuật</rdfs:range>
</rdf:Property>

<rdfs:Property rdf:ID="Sáng_tác">
  <rdfs:subPropertyOf rdf:resource="#Tạo_ra"/>
  <rdfs:domain>Nhạc_sĩ</rdfs:domain>
  <rdfs:range>Bài_hát</rdfs:range>
</rdfs:Property>

<rdfs:Property rdf:ID="Vẽ">
  <rdfs:subPropertyOf rdf:resource="#Tạo_ra"/>
  <rdfs:domain>Họa_sĩ</rdfs:domain>
  <rdfs:range>Bức_tranh</rdfs:range>
</rdfs:Property>
...
</rdf:RDF>

```



Hình 4. Ví dụ minh họa kết hợp ngữ nghĩa

Các thể hiện của lược đồ RDF ở phần dưới trong hình 4 được biểu diễn bằng cú pháp RDF như sau:

```
<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
  xmlns:myschema="http://myexample/1.0/">
  <rdf:Description rdf:ID="r1">
    <myschema:tên>Van</myschema:tên>
    <myschema:họ>Gogh</myschema:họ>
    <rdf:type rdf:resource="http://myexample.vn/Nghệ_sĩ"/>
    <myschema:vẽ>
      <rdf:Description rdf:ID="r2">
        <rdf:type rdf:resource="http://myexample.vn/1.0/Bức_tranh"/>
      </rdf:Description>
    </myschema:vẽ>
  </rdf:Description>

  <rdf:Description rdf:ID="r3">
    <myschema:tên>Michael</myschema:tên>
    <myschema:họ>Jackson</myschema:họ>
    <rdf:type rdf:resource="http://myexample.vn/Nghệ_sĩ"/>
    <myschema:sángtác>
      <rdf:Description rdf:ID="r4">
        <rdf:type rdf:resource="http://myexample.vn/Bài_hát"/>
      </rdf:Description>
    </myschema:sángtác>
  </rdf:Description>
</rdf:RDF>
```

2.2 Tầng Ontology [3]

Lược đồ RDF cho phép định nghĩa những từ vựng, cấu trúc và các ràng buộc trong việc mô tả siêu dữ liệu. Nhưng lược đồ RDF chưa thật sự đầy đủ cho việc mô hình hóa và hỗ trợ suy luận trên *Semantic Web*. Ngôn ngữ OIL (*Ontology Inference Layer - OIL*) được đề ra là một dạng mở rộng của lược đồ RDF. Nó cho phép thể hiện ngữ nghĩa hình thức, giúp hỗ trợ suy diễn tự động.

Tập từ khóa của ngôn ngữ ontology OIL được phân làm hai nhóm chính: nhóm từ khóa *ontology-container*: bao gồm các từ khóa thông dụng như: title, creator, subject, description, ..., và nhóm từ khóa để định nghĩa ontology (*ontology-definitions*) cho phép định nghĩa các lớp, tính chất, và các ràng buộc, ... chẳng hạn như: class-def, slot-def, slot-constraint, has-value, ...

Ví dụ sau minh họa một ontology mô tả đời sống thiên nhiên theo tập từ khóa của OIL:

```
ontology-container
  title "African Animals"
  creator "Ian Horrocks"
  subject "animal, food, vegetarians"
  description "A didactic example ontology describing African animals"
  ...
ontology-definitions
  slot-def eats
    inverse is-eaten-by
  slot-def has-part
    inverse is-part-of
    properties transitive
  class-def animal
  class-def plant
    subclass-of NOT animal
  class-def tree
    subclass-of plant
  class-def plant
    slot-constraint is-part-of
      has-value tree
  class-def defined carnivore
    subclass-of animal
    slot-constraint eats
      value-type animal
  class-def defined herbivore
    subclass-of animal, NOT carnivore
```

```

slot-constraint eats
  value-type
    plant OR
    slot-constraint is-part-of
    has-value plant
    
```

Việc mô tả theo ngôn ngữ OIL giàu ngữ nghĩa hơn RDF. Chẳng hạn như đoạn chương trình trên:

```

class-def defined herbivore
  subclass-of animal, NOT carnivore
  slot-constraint eats
    value-type
      plant OR
      slot-constraint is-part-of
      has-value plant
    
```

mô tả động vật ăn cỏ (herbivore) là động vật (animal) nhưng không phải là động vật ăn thịt (carnivore), có tính chất là ăn (eat), và chỉ ăn thực vật (eat plant) hoặc những bộ phận của thực vật (is-part-of plant).

Phần đầu của ví dụ trên mô tả:

```

slot-def has-part
  inverse is-part-of
  properties transitive
    
```

cho biết *is-part-of* có nghĩa ngược lại với *has-part* được thể hiện qua từ khóa *inverse*. Ngoài ra, ta cũng dễ dàng nhận thấy những từ khóa: NOT, OR, has-value, ... không xuất hiện trong lược đồ RDF nhưng chúng có trong OIL, đó là lý do khiến cho OIL giàu ngữ nghĩa trong mô tả hơn.

Để tận dụng tính thông dụng trong việc mô tả tài nguyên và trao đổi định dạng giữa các ứng dụng của RDF, một số từ khóa nâng cao của OIL: OR, NOT, has-value, inverse, ... đã được bổ sung vào lược đồ RDF bằng cách thêm không gian tên "oil:" và phía trước các từ khóa của OIL như bảng sau:

Từ khóa bổ sung cho lược đồ RDF	Thành phần cơ sở của OIL	Kiểu
oil:AND	AND	Lớp
oil:OR	OR	Lớp
oil:NOT	NOT	Lớp
oil:HasValue	has-value	Thuộc tính
oil:ClassExpression	class-expression	Lớp
...

2.3 Tầng Logic [3]

Tầng logic được xem như là một cơ sở luật trên *Semantic Web*. Bản chất của cơ sở luật này có dạng như một hệ chuyên gia. Nó được xây dựng dựa trên tập từ khóa của *ontology*, và được định nghĩa như sau:

```

<rdfs:Class rdf:ID="RuleBase">
  <rdfs:subClassOf rdf:resource="http://www.w3.org/TR/1999/PR-rdf-
    schema-19990303#Resource"/>
</rdfs:Class>
    
```

khi đó ta có thể mô tả cho suy diễn *tam đoạn luận (syllogism)* như sau:

```

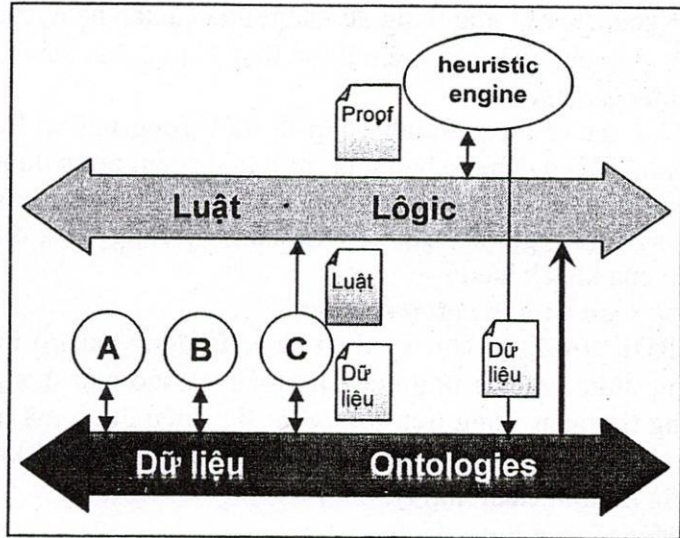
<rdf:Description xmlns:syllogism="http://old.greece/syllogism/">
  <rdf:type rdf:resource="http://www.ontoknowledge.org/oil/rdfs/schema#RuleBase"/>
  <syllogism:premise>
    If it rains, you get wet
  </syllogism:premise>
  <syllogism:fact>
    it rains
  </syllogism:fact>
  <syllogism:conclusion>
    you get wet
  </syllogism:conclusion>
</rdf:Description>
    
```

2.4 Tầng Proof [8]

Như chúng ta đã biết, tầng logic giúp hỗ trợ suy luận dựa vào cơ sở luật. Chẳng hạn như, ta có luật: “Nếu A bán nhiều hơn 100 sản phẩm, thì A thuộc nhóm bán hàng giỏi”. Khi đó, nếu có sự kiện “Jones bán được 101 sản phẩm”, thì hệ thống sẽ suy ra “Jones thuộc nhóm bán hàng giỏi”.

Tầng Proof chứng minh các suy diễn của hệ thống bằng cách liên kết các dữ kiện. Chẳng hạn như, từ dữ kiện “Jones bán được 55 quần” và “Jones bán được 46 áo” hệ thống sẽ liên kết chúng lại thành dữ kiện “Jones bán được 101 sản phẩm”, từ đó suy ra “Jones thuộc nhóm bán hàng giỏi”. Như vậy, nhiệm vụ của tầng Proof phải phát hiện ra các liên kết của các dữ kiện đầu vào cho các suy

diễn, nhưng việc phát hiện các liên kết trên môi trường Semantic Web rất khó khăn, có thể lên tới hàng nghìn, thậm chí hàng triệu liên kết. Quá trình xử lý phức tạp của chúng có thể khái quát như hình 5. Một số thành phần như A, B chỉ đơn thuần là cung cấp dữ liệu cho các thành phần khác. Các thành phần như C thông minh hơn, có thể dùng dữ liệu để xây dựng luật. Thành phần thông minh nhất đó là “*heuristic engine*”, nó dựa vào cơ sở luật và các phát biểu để rút ra kết luận, sau đó trả các kết quả về dưới dạng các chứng cứ làm đầu vào cho các suy diễn khác.

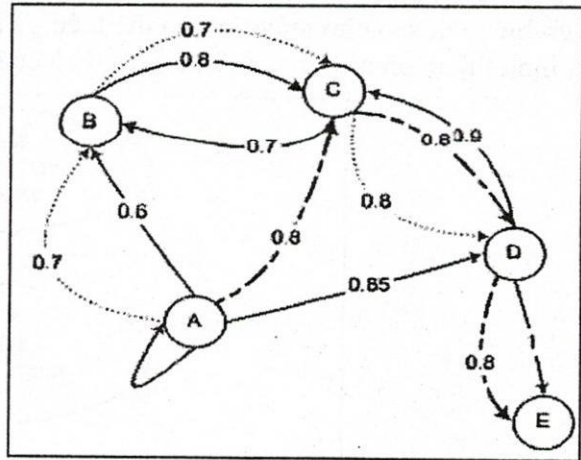


Hình 5. Quá trình xử lý thông tin ở tầng Proof

2.5 Tầng Trust [7, 8]

Trên môi trường *Semantic Web* việc mô tả thông tin không bị giới hạn, vì vậy cần phải có hệ thống để chứng thực ai là chủ nhân của các phát biểu. Đó là nguyên nhân của sự ra đời chữ ký điện tử, nó dựa trên các công trình toán học và cách mã hóa. Chữ ký điện tử giúp máy tính có thể đánh giá được mức độ tin cậy của các thông tin trên Web.

Mức độ cao hơn của chữ ký điện tử đó là sự tin cậy (*Trust*). Nó được phân thành các loại sau: sự tin cậy giữa các cá nhân (*Interpersonal Trust*), sự tin cậy của một hệ thống (*System Trust, Impersonal Trust*), sự tin cậy theo khuynh hướng (*Dispositional Trust*). **Sự tin cậy giữa các cá nhân** là một dạng tin cậy trực tiếp của một cá thể này đối với một cá thể khác trong một ngữ cảnh cụ thể nào đó. Chẳng hạn như, cô An tin tưởng anh Nam trong vai trò của một thợ sửa xe máy, nhưng lại không tin tưởng trong vai trò người giữ trẻ. **Sự tin cậy của một hệ thống**, nó không dựa trên bất kỳ đặc điểm, trạng thái của cá thể được tin cậy mà dựa trên độ tin cậy của hệ thống mà các cá thể đang tồn tại.



Hình 6. Đồ thị lan truyền độ tin cậy

Ví dụ như, hệ thống tiền tệ. **Sự tin cậy theo khuynh hướng** là sự tin cậy dựa trên quan điểm tin cậy chung của cá thể, hoàn toàn độc lập với môi trường.

Trên *Semantic Web* dùng cơ chế lan truyền để xác định độ tin cậy. Nó được minh họa như hình 6, là một đồ thị có hướng, nhân giá trị độ tin cậy được gán trên cạnh, mỗi cạnh (a,b) sẽ biểu diễn độ tin cậy của a đối với b được ký hiệu là $T(a,b)$. Khi đó, $T(a,b)$ sẽ khác với $T(b,a)$, và nếu ngữ cảnh tương tác giữa a và b khác nhau sẽ cho những giá trị độ tin cậy khác nhau.

3. Một số ứng dụng trên môi trường Semantic Web

3.1 Tìm kiếm ngữ nghĩa

Trong khi các cơ chế tìm kiếm trên Web hiện nay thực hiện tìm kiếm thiếu chính xác thì các cơ chế tìm kiếm trên *Semantic Web* cho ra kết quả tìm kiếm chính xác hơn. Vì cơ chế tìm kiếm của các *search engine* hiện nay dựa vào từ khóa, nó không thể hiện được ngữ nghĩa trong từng ngữ cảnh khác nhau. Do đó, không thể cho ra kết quả tìm kiếm chính xác được, các trang có cùng ý nghĩa với việc tìm kiếm không được tìm thấy. Trên *Semantic Web*, ngữ nghĩa của các từ trong từng ngữ cảnh sẽ được mô tả bằng RDF và được tích hợp vào các *search engine* để tạo ra khả năng tìm kiếm hiệu quả và chính xác hơn.

Ngoài ra, các ứng dụng về khám phá và xếp hạng các kết hợp ngữ nghĩa trên *Semantic Web* cũng giúp cho việc tìm kiếm được thực hiện chính xác với ngữ nghĩa mong muốn.

3.2 Internet Agent

Các *agent* thông minh cũng là một trong những hướng ứng dụng chính trên môi trường *Semantic Web*. Chẳng hạn như, một loại *agent* được dùng để thay thế cho khách hàng, chúng sẽ được dùng để thực hiện hầu hết các công việc của một khách hàng như: tham gia đấu thầu, thương lượng giá cả, sắp xếp và đăng ký công việc, đưa ra lời khuyên... và quản lý mọi hoạt động của khách hàng.

3.3 An toàn và bảo mật thông tin

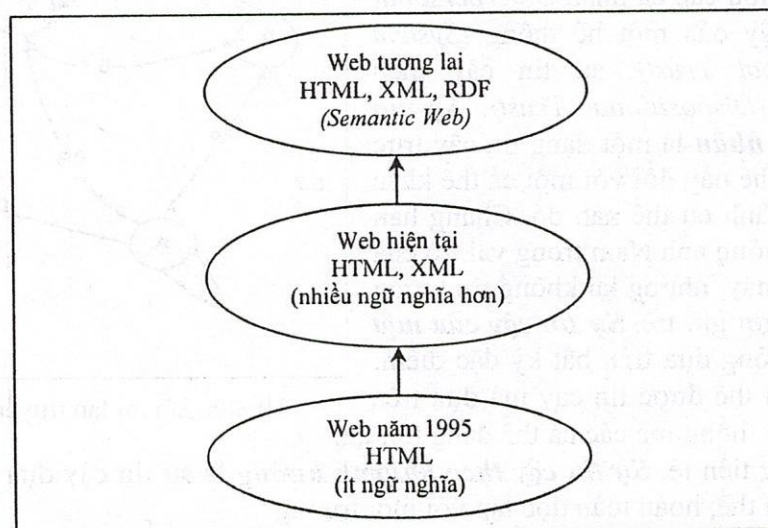
RDF cùng với chữ ký điện tử sẽ là đóng vai trò quan trọng cho việc xây dựng hệ thống chứng thực, và các ứng dụng an toàn và bảo mật thông tin trong thương mại điện tử. Những thông tin quan trọng trên *Semantic Web* đều được mã hóa và quản lý một cách kỹ lưỡng. Hầu hết các ứng dụng trên môi trường *Semantic Web* có khả năng quản lý, kiểm soát và phân bổ thông tin một cách hợp lý.

3.4 Một số ứng dụng khác

Các công nghệ của *Semantic Web* đang cố gắng tích hợp vào các máy palmtop, laptop, desktop, server,... để chúng có thể nói chuyện với nhau, hoặc ra quyết định một cách tự động.

4. Kết luận

Bài báo này trình bày các thành phần chính của *Semantic Web*, và một số hướng ứng dụng chính của *Semantic Web*. Với những nội dung đã trình bày, chúng ta thấy được tương lai gần của thế hệ *Semantic Web*. Tư tưởng chính của *Semantic Web* là đưa phần ngữ nghĩa vào Web hiện tại, sao cho máy tính có thể hiểu và xử lý được thông tin trên Web một cách tự động. Sự hình thành phát triển của *Semantic Web* có thể được khái quát như hình 7.



Hình 7. Sự hình thành và phát triển của Semantic Web

Hiện nay *Semantic Web* là một lĩnh vực còn khá mới mẻ và chưa có một hệ *Semantic Web* thực sự nào trên thế giới. Tất cả chỉ đang mới bắt đầu, và đang trong giai đoạn thử nghiệm.

Chúng ta đang mong đợi thế hệ Web này, cùng với các ứng dụng đa dạng và hữu ích của nó. Có thể nói *Semantic Web* sẽ tác động rất nhiều đến cuộc sống của tất cả chúng ta.

AN OVERVIEW OF SEMANTIC WEB

Le Hoai Bac, Nguyen Thanh Nghi

ABSTRACT: *This paper will present the development of Semantic Web, describe some main layers of Semantic Web such as RDFs (Resource Description Frameworks), RDFs Schemas, OILs (Ontology Inference Layers), reasoning logics, deductive proofs through putting all links into a proof, and how to fan out trust relationships from a system to others on the Trust layer. Finally, we will give some applications on Semantic Web environment.*

TÀI LIỆU THAM KHẢO

- [1]. Bhavani Thuraisingham, *XML Databases and the Semantic Web*, CRC Press, March 2002.
- [2]. Ian Horrocks, Stefan Decker, Serger Melnik, *The Semantic Web: The Roles of XML and RDF*, IEEE Press, October 2000.
- [3]. Ian Horrocks, Jeen Broekstra, Michel Klein, Stefan Decker, Dieter Fensel, *Adding formal semantics to the Web*, In Proceeding of 12th International Conference on Knowledge Engineering and Knowledge Management, September 2000.
- [4]. Jérôme Euzenat, *Eight Questions about Semantic Web Annotations*, IEEE Press, MARCH/APRIL 2002.
- [5]. Stefan Decker, Sergey Melnik, Frank Van Harmelen, Dieter Fensel, and Michel Klein, *The Semantic Web: the roles of XML and RDF*, IEEE Press, MAY/JUNE 2000.
- [6]. Jeff Heflin, James Hendler, *A portrait of the Semantic Web in action*, IEEE Press, MARCH/APRIL 2001.
- [7]. Mui, L., Mohtashemi, M., and Halberstadt, *A computational model of trust and reputation*, In Proc. of the 35th Hawaii Int'l Conf. on System Sciences, March 2002.
- [8]. Aaron Swartz, *The Semantic Web In Breadth*, LogicError Recommendation, May 2002; Available online at: <http://logicerror.com/semanticWeb-long>.
- [9]. O. Lassila and Ralph Swick, *Resource Description Framework (RDF) Model and Syntax Specification*, W3C Recommendation, Feb. 1999; available online at: <http://www.w3.org/TR/REC-rdf-syntax/>
- [10]. D. Brickley and R. Guha, *Resource Description Framework (RDF) Schema Specification*, W3C Candidate Recommendation, January 2003; available online at: <http://www.w3.org/TR/2003/WD-rdf-schema-20030123/>
- [11]. Klyne G., Carroll J. (Editors), *Resource Description Framework (RDF): Concepts and Abstract Syntax*, World Wide Web Consortium Working Draft, January 2003; available online at: <http://www.w3.org/TR/2003/WD-rdf-concepts-20030123/>
- [12]. A. Sheth and K. Anyanwu, *The ρ Operator: Discovering and Ranking Associations on the Semantic Web*, *SIGMOD Record*, Vol. 31, No. 4, Dec 2002.