**TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN**

**KHOA KHOA HỌC MÁY TÍNH**

**\_\_\_ \*\*\* \_\_\_**



**MÔN HỌC  
Các hệ Cơ sở Tri thức**

**Báo cáo đồ án**

Đề tài: **Xây dựng hệ chuyên gia tư vấn mua máy tính**

**GVLT**: Nguyễn Thị Ngọc Diễm

**Thành viên**:

* **Nguyễn Trần Minh Tân** 13520747
* **Lê Thị Tuyết Mai** 13520489
* **Nguyễn Minh Phúc** 13520638

**Hồ Chí Minh, Ngày 20 Tháng 1 Năm 2016**

CHƯƠNG 1: TỔNG QUAN VỀ ĐỀ TÀI

1. ĐẶT VẤN ĐỀ

Trước hết, thực tế cho thấy một điều rằng thế giới ngày nay nói chung và lĩnh Khoa

học Máy tính nói riêng đang thay đổi một cách rất mạnh mẽ. Những vấn đề tưởng chừng như không thể giải quyết bằng một người, hai người, thì bây giờ chỉ cần một chiếc máy tính là có thể giải quyết một cách nhanh chóng. Không dừng lại ở đó, những lĩnh vực mà trước đây Khoa học Máy tính không thể chen chân vào thì nay cũng đã bắt đầu thấy được sự hiện diện theo một cách rất tích cực. Tại sao lại như thế, tất cả nằm ở con người. Nhân loại rất vất vả trong việc tìm ra cách để lưu trữ thông tin, nhưng khi đã lưu trữ được rồi, chúng ta lại có thêm tham vọng đó là giúp chiếc máy tính có thể dựa vào những nguồn thông tin lưu trữ, sự kiện có sẵn đó mà có thể có khả năng lập luận, suy luận để rút ra được những kết luận từ những nguồn dữ liệu như trên.

Trí tuệ nhân tạo (Artificial Intelligence) ra đời như là một điều tất yếu đi cùng với sự phát triển của Khoa học Máy tính, và ứng dụng của chúng thì được sử dụng rộng rãi. Hệ Chuyên Gia (Expert System) là một hệ thống dựa trên tri thức, là một chương trình máy tính chứa một số tri thức đặc thù của một hoặc nhiều chuyên gia là con người về một chủ

đề cụ thể, giúp giải quyết các vấn đề phức tạp trong cuộc sống. Tại đây nhóm quyết định vẫn đề tìm hiểu là tư vấn người dùng mua máy tính.

Trên thị trường hiện nay muốn tìm ra một sảng phẩm tốt, bền và có thường hiệu, hay những sản phẩm mới được ra có thành phần và chất lượng cao hơn những sản phẩm khác. Đối với hàng ngàn hàng triệu sinh viên có lẽ máy tính là cái mà mọi người ai cũng sở hữu cho mình, giá trị của chiếc máy tính rất lớn, có thể với nhiều sinh viên nó đơn giản không có giá trị lớn, nhưng đối với một lượng lớn sinh viên khác nó là tài sản quý báu nhất của sinh viên. Vì vậy để mua một chiếc máy tính ngon hay nói cách khác là ‘xịn’ có cấu hình ngon , có chức năng tốt đã những vấn đề sinh viên tìm hiểu. Sinh viên tìm hiểu ở đâu ? Sinh viên muốn mua máy tính có thể sẽ lên seach như sau : những chiếc máy tính mới nhất hiện nay ? Gía thành của máy tính có cao không ? Ram bao nhiêu thì dùng được ? Chơi game được ? Màn hình ? ….. Nhiều thông tin mà sinh viên cầN.

Vậy những thông tin được lấy đó họ có thể tổng hợp và tư vấn cho bạn 1 chiếc má tính phù hợp với yêu cầu của bạn không?

Vì vậy nhóm chúng em đề xuất xây dựng chương trình tư vấn mua máy tính .

1. MỤC TIÊU NGHIÊN CỨU

Dựa trên những phân tích ở trên, trước hết mục tiêu là muốn tạo ra sự tiện lợi trong

Trong việc cọn mua máy tính cho hợp ý mình vừa tiến. xây dựng một chương trình tìm kiếm tư vấn để khách hàng mua máy tính theo ý mình một cách chuyên nghiệp nhất có thể.

Tìm hiểu các phương pháp biểu diễn tri thức, các kỹ thuật suy diễn dùng trong hệ chuyên gia cho việc tư vấn mua máy tính một cách hiệu quả nhất . Cuối cùng là xây dựng được một hệ thống như mong đợi.

1. ĐỐI TƯỢNG NGHIÊN CỨU

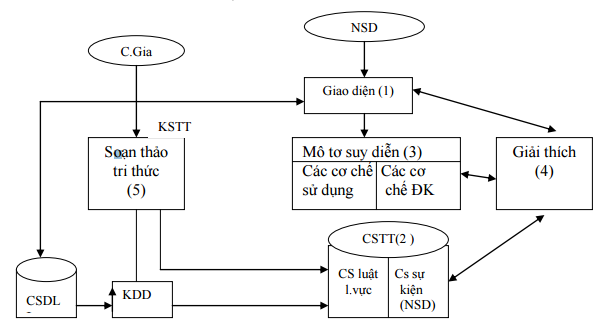
Đề tài sẽ xoay quanh các vấn đề về miền tri thức máy tính, tìm ra những chiếc máy tính mà tư vấn theo yêu cầu phù hợp nhất có thể và dùng các kỹ thuật trong việc tổ chức tri thức, lưu trữ tri thức và suy diễn.

1. GIỚI HẠN VÀ PHẠM VI NGHIÊN CỨU

Vì thời gian thực hiện đề tài có hạn nên nhóm chúng em xin phép chọn miền tri thức là laptop và PC.

Giới hạn nghiên cứu của bọn em ở mức tư vấn người dùng mua laptop cho phù hợp với yêu cầu của người dùng.

1. PHƯƠNG PHÁP NGHIÊN CỨU
   * Tìm hiểu tài liệu: nghiên cứu tài liệu, thông tin liên quan, tri thức từ chuyên gia, tìm nguồn tài liệu liên quan đến vấn đề đang thực thi.
   * Phân tích dữ liệu : Tri thức lấy về ở dạng thô, phân loại sơ bộ tri thức
   * Lưu trữ tri thức: Phân loại tri thức xong chúng ta thực hiện việc lưu trữ tri thức theo mô hình cơ sở tri thức mà chúng ta đã thiết kế
   * Suy luận tìm ra kết quả dựa vào input của người dùng Dựa vào tri thức thức được lưu trữ dùng chiến lược suy diễn để tìm ra kết quả.
2. NỘI DUNG THỰC HIỆN



CHƯƠNG 2: CƠ SỞ LÝ THUYẾT

1. HỆ CƠ SỞ TRI THỨC

2.1.1. Cơ sở tri thức.

Tri thức là những gì mà một người có thể biết và hiểu được. Tri thức có thể được phân loại thành tri thức có cấu trúc hoặc không có cấu trúc, tri thức rõ ràng hay là tri thức ngụ ý, không rõ ràng. Những gì mà chúng ta biết được thường là các tri thức rõ ràng. Tri thức không có cấu trúc mà vẫn hiểu được, nhưng không được phát biểu rõ ràng là các tri thức ngầm ý, tri thức không rõ. Khi tri thức được tổ chức để có thể chia sẻ thì lúc đó tri thức được gọi là tri thức có cấu trúc. Để có thể chuyển đổi từ tri thức không rõ ràng sang tri thức rõ ràng thì tri thức đó cần phải được cấu trúc hóa và định dạng lại.

Phân loại tri thức: như đã đề cập ở trên, tri thức có nhiều loại tùy thuộc vào tính chất cấu trúc và tính chất rõ ràng của tri thức. Tri thức có thể được phân loại thành các loại tri thức sau, đây là các loại tri thức thường gặp trong thực tế:

Tri thức có 2 dạng tồn tại chính là tri thức ẩn và tri thức hiện

* Tri thức hiện là những tri thức được giải thích và mã hóa dưới dạng văn bản, tài liệu, âm thanh, phim, ảnh,… thông qua ngôn ngữ có lời hoặc không lời, nguyên tắc hệ thống, chương trình máy tính, chuẩn mực hay các phương tiện khác. Đây là những tri thức đã được thể hiện ra ngoài và dễ dàng chuyển giao, thường được tiếp nhận qua hệ thống giáo dục và đào tạo chính quy.
* Tri thức ẩn là những tri thức thu được từ sự trải nghiệm thực tế, dạng tri thức này thường ẩn trong mỗi cá nhân và rất khó "mã hóa" và chuyển giao, thường bao gồm: niềm tin, giá trị, kinh nghiệm, bí quyết, kỹ năng... VD: Trong bóng đá, các cầu thủ chuyên nghiệp có khả năng cảm nhận bóng rất tốt. Đây là một dạng tri thức ẩn, nó nằm trong mỗi cầu thủ. Nó không thể "mã hóa" thành văn bản, không thể chuyển giao, mà người ta chỉ có thể có bằng cách tự mình luyện tập.

Tùy thuộc vào ứng dụng cụ thể mà tri thức được biểu diễn theo những phương thức khác nhau. Ứng với mỗi dạng biểu diễn đó có một cơ chế để xử lý tri thức đó.

Hệ cơ sở tri thức là một tập hợp các cơ sở lập luận, các luật, các qui trình, thủ tục

được tổ chức thành các lược đồ (giản đồ). Đó là tập hợp của tất cả các thông tin cũng như tất cả kiến thức về một lĩnh vực cụ thể nào đó.

Quá trình thu thập tri thức được gọi là quá trình tiếp nhận tri thức và định dạng tri thức được thực hiện từ nhiều nguồn khác nhau, đặc biệt là từ các chuyên gianh trong các lĩnh vực cụ thể. Công việc này là một trong những bước quan trọng và thường mất rất nhiều thời gian cũng như công sức trong quá trình xây dựng một hệ cơ sở tri thức được sử dụng trong hệ chuyên gia. Trong quá trình phát triển một hệ chuyên gia, các kỹ sư tri thức, những nhà chuyên môn trong lĩnh vực trí tuệ nhân tạo, có nhiệm vụ thu thập kiến thức từ các chuyên gia thuộc lĩnh vực chuyên ngành, sau đó “sao chép” các tri thức đó vào cơ sở tri thức và diễn đạt các tri thức đó dưới dạng có thể dùng được trong hệ chuyên gia.

2.1.2. Hệ chuyên gia

Hệ chuyên gia, còn gọi làhệ thống dựa tri thức, là một[chương trình máy tính](https://vi.wikipedia.org/wiki/Ph%E1%BA%A7n_m%E1%BB%81m) chứa một số tri thức đặc thù của một hoặc nhiều chuyên gia con người về một chủ đề cụ thể nào đó. Các chương trình thuộc loại này đã được phát triển từ các[thập niên 1960](https://vi.wikipedia.org/wiki/Th%E1%BA%ADp_ni%C3%AAn_1960) và [1970](https://vi.wikipedia.org/wiki/Th%E1%BA%ADp_ni%C3%AAn_1970), và trở thành ứng dụng [thương mại](https://vi.wikipedia.org/wiki/Th%C6%B0%C6%A1ng_m%E1%BA%A1i) từ [thập niên 1980](https://vi.wikipedia.org/wiki/Th%E1%BA%ADp_ni%C3%AAn_1980). Dạng phổ biến nhất của hệ chuyên gia là một chương trình gồm một tập[**luật**](https://vi.wikipedia.org/w/index.php?title=Lu%E1%BA%ADt_(l%C3%B4gic)&action=edit&redlink=1)phân tích thông tin (thường được cung cấp bởi người sử dụng hệ thống) về một lớp vấn đề cụ thể, cũng như đưa ra các phân tích về các vấn đề đó, và tùy theo thiết kế chương trình mà đưa lời khuyên về trình tự các hành động cần thực hiện để giải quyết vấn đề. Đây là một hệ thống sử dụng các khả năng lập luận để đạt tới các kết luận.

Hệ chuyên gia sử dụng các tri thức của những chuyên gia để giải quyết các vấn đề( bài toán) khác nhau thuộc mọi lĩnh vực.

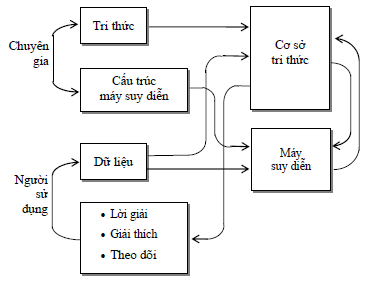
Tri thức ( knowledge ) trong hệ chuyên gia phản ánh sự tinh thông được thu thập từ các chuyên gia hay các nhà bác học, từ sách vở, tạp chí, từ internet với nguồn thông tin vô cùng lớn. Các thuật ngữ hệ chuyên gia, hệ thống dựa trên tri thức ( knowledge-

19

based system ) hay hệ chuyên gia dựa trên tri thức( knowledge-based expert system ) thường có cùng nghĩa.

Một hệ chuyên gia gồm nhiều thành phần nhưng chú trọng 2 phần chính đó là: Cơ sở tri thức(knowledge base) và động cơ suy diễn (inference engine). Cơ sở tri thức chứa các tri thức để từ đó, động cơ suy diễn tạo ra câu trả lời.

Người sử dụng ( user ) cung cấp sự kiện (facts) là những gì đã biết, đã có thật hay những thông tin có ích cho hệ chuyên gia và nhận được những câu trả lời là những lời khuyên hay những gợi ý đúng đắn.



Việc xây dựng một hệ chuyên gia được hiểu như là việc ứng dụng các kiến thức đã học , kiến thức khoa học vào việc vận hành và bảo trì tri thức, người làm được những việc như trên chính là kỹ sư tri thức (Knowledge Engineer). Người kỹ sư tri thức phải đảm bảo rằng máy tính luôn có đủ tri thức để giải quyết một vấn đề nào đó. Người kỹ sư tri thức phải chọn một hoặc nhiều hình thức miêu tả các tri thức được yêu cầu dưới dạng các mô hình kí tự trong bộ nhớ của máy tính và như vậy anh ta phải chọn một phương pháp biểu

20

diễn tri thức. Kỹ sư tri thức phải đảm bảo rằng máy tính có thể sử dụng tri thức một cách có hiệu quả bằng việc chọn ra một trong các phương pháp biểu diễn.

**Các đặc tính hệ chuyên gia**

1. **Tách biệt giữa tri thực và suy diễn**

* **Cơ sở tri thức và duy diễn được tách rời. Phân tách cơ sở tri thức và mô tơ suy diễn có giá trị trong hệ chuyên gia, đảm bảo tính độc lập trong việc mã hóa tri thức và xử lý tri thức đó.**
* **Phân tách tri thức ra khỏi động cơ suy diễn để tạo điều kiện biểu diễn tri thực một cách tự nhiên hơn.**
* **Cơ sở tri thức được tách biệt khỏi cấu trúc điều khiển cấp thấp của chương trình, những người phát triển hệ chuyên gia có thể tập trung một cách trức tiếp vào việc nắm bắt và tổ chức giải quyết vấn đề hơn là việc thiwcj hiện các chi tiết trong việc cài đặt máy tính.**
* **Sự tách biệt cho phép thay đổi một phần cơ sở tri thực mà không ảnh hưởng lớn đến các phần khác của chương trình.**
* **Sự tách biệt cho phép thay đổi một phần cơ sở tri thức mà không ảnh hưởng lớn đến các phần khác của chương trình**
* **Sự tách biệt này cho phép một phần mềm điều khiển và giao tiếp, có thể sử dụng nhiều hệ thống khác nhau/**

**Ví dụ : Xây dựng một Shell. Các nhà phát triển xây dựng một chương trình cốt lõi của hệ thống. Sau đó tạo ra cơ sở tri thức để giải quyết bài toán.**

* **Đây là một đặc tính nổi bật trong hệ chuyên gia khác so với phần mềm thông thường.**

1. **Tri thức chuyên gia**

* **Đặc điểm nổi bật hệ chuyên gia là khả năng thu thập tri thức của chuyên gia. Tri thức bao gồm tri thức về lĩnh vực và tri thức kĩ năng giải quyết vấn đề. Các tri thức hu được từ chuyên gia không nhất thiết phải là các ý tưởng sang chói hay độc đáo mà đặc biệt và sâu về lĩnh vực cụ thể.**

1. **Lập luận may rủi**

* **Các chuyên gia thường dùng kinh nghiệm để giả đúng và hiệu quả baiftoans đang xét. Qua kinh nghiệm đã dùng, họ hiểu vấn đề một cách thực tế và lưu trữ dưới dạng may rủi. chiến lược may rủi được dùng trong hệ chuyên gia để giúp người ta đi nhanh đến giải pháp. Trong hệ chuyên gia chiến lực lập luận may rủi không giống như các thủ tục chính xác của chương trình bình thường.**

1. **Lập luận không chính xác**

**Hệ chuyên gia được coi là thành công trong ứng dụng cần đến lập luận không chính xác. Những loại ứng dụng này được đặc trưng bằng thông tin không chắc chắn, nhọc nhằng. Trong thực tế thường xảy ra chẳng hạn như bác sĩ khám bện cho bệnh nhân vào cấp cứu trong hoàn cảnh không có nhiều thông tin về bệnh nhân.**

**Các phương pháp biểu diễn tri thức**

1. Thể hiện tri thức bằng cặp ba đối tượng – thuộc tính giá trị
2. Thể hiện các sự kiện không chắn chắn

Trong hệ chuyên gia các sự kiện thường không khẳng định chính xác đúng hay sai và độ chắc chắn tuyệt đối. Đối với các sự kiện chắc chắn người tà không hoàn toàn biết rõ. Không có gì đảm bảo một sự kiện đúng. Vì vậy người ta dùng khái niệm “ mức độ tin cây” vào sự kiện hay viết tắt là cf ( certainty Factor )

Các sự kiện mờ

Trong thực tế cho thấy rằng thể hiện các vấn đề của thế giới thực đôi khi cần dùng đến thuật ngữ nhập nhằng. Chẳng hạn với câu ông ấy “ cao “ là không rõ rang. Không rõ rang vì không xác định được “ cao “ có ý nghĩa gì. Các thuật ngữ nhập nhằng được thể hiện trong tập mờ. Khi đó ta có tập mờ thể hiện chiều caoo của con người.

1. Thể hiện tri thức nhờ các luật

Các sự kiện được cung cấp có ý nghĩa rất lớn đối với hoạt dộng của hệ chuyên gia. Các sự kiện này cho phép hệ thống

## 2.3 Ontology

### 2.3.1 Khái niệm Ontology

Khoa học máy tính: Ontology là một mô hình dữ liệu biểu diễn một lĩnh vực và được sử dụng để suy luận về các đối tượng trong lĩnh vực đó và mối quan hệ giữa chúng.

Quan điểm Gruber (1993): “Ontology là một “đặc tả” “tường minh” của một sự khái niệm hóa” (An ontology is an explicit specification of a conceptualization).

**Kết luận**: Ontology là phương thức biểu diễn tri thức nhằm biểu diễn tất cả các khái niệm cùng với các mối quan hệ giữa chúng trong 1 lĩnh vực nào đó. Các khái niệm và quan hệ này hình thành nên bộ từ vựng dung chung cho những người có nhu cầu chia sẻ thông tin với nhau trên mạng.

### 2.3.2 Các thành phần của Ontology

Các thành phần phổ biến của ontology bao gồm:

* Các lớp (Classes) - Khái niệm: các khái niệm về các đối tượng hoặc các loại vật chất
* Các cá thể (Individuals hay Instance) - Thể hiện: các đối tượng (các đối tượng cơ bản)
* Các thuộc tính (Properties): các đặc tính, đặc trưng, các đặc điểm hoặc các tham số mà các đối tượng có thể có.
* Các mối quan hệ (Relations): các cách để các lớp, các thực thể có thể được liên kết tới các lớp (hoặc thực thể) khác.
* Các hàm (Functions): Là một loại thuộc tính hay quan hệ đặc biệt, trong đó phần tử thứ n là duy nhất đối với n-1 phần tử còn lại.
* Các giới hạn: các mô tả được phát biểu chính xác và hợp lệ.
* Các luật (Rules): được phát biểu dưới dạng if – then. Với mỗi một tập luật R sẽ có 1 tập đỉnh ứng với luật đó. Kí hiệu Nodes(R).
* Các tiền đề (Axioms): Biểu diễn các phát biểu luôn đúng mà không cần phải chứng minh hay giải thích.
* Các sự kiện (Facts): Các thay đổi của các thuộc tính hoặc các mối quan hệ

***Các thành phần chính***

#### Các lớp (Classes)

Là trung tâm của hầu hết các ontology.

Mô tả các khái niệm trong miền lĩnh vực.

Các lớp thường được tổ chức phân cấp và áp dụng kỹ thuật thừa kế.

Một lớp có thể có các lớp con biểu diễn khái niệm cụ thể hơn so với lớp cha.

#### Các cá thể (Individual)

Biểu diễn các phần tử riêng biệt của khái niệm.

Là các thành phần cơ bản, nền tảng của 1 Ontology.

Là sự thể hiện của một lớp. Mỗi thể hiện của lớp biểu diễn một sự việc cụ thể hóa của khái niệm đó.

Trong một lớp có thể có nhiều sự thể hiện khác nhau.

#### Các thuộc tính (Properties)

Thuộc tính là sự mô tả các đối tượng trong Ontology.

Bao gồm 2 thành phần: Tên và Giá trị.

#### Các mối quan hệ (Relation)

Quan hệ giữa các đối tượng trong ontology cho biết đối tượng này liên hệ với đối tượng khác như thế nào.

Các loại quan hệ:

* Quan hệ giữa các lớp.
* Quan hệ giữa các thực thể.
  + Quan hệ giữa một thực thể và một lớp
  + Quan hệ giữa các cá thể và lớp.

### 2.3.3 Phân loại Ontology

Phân loại dựa trên phạm vi của các đối tượng được mô tả:

* Knowledge Representation ontology (ontology biểu diễn tri thức).

+Dựa trên các cách biểu diễn tri thức truyền thống.

+Ví dụ: Frame-Ontology.

* General/Common ontology (ontology tổng quát).

+ Từ vựng liên quan đến mọi thứ, sự kiện, thời gian, không gian, ...

+ Ví dụ: ontology về bảng trao đổi giữa meter và inch.

* Metadata-ontology (cung cấp từ vựng).

+Định nghĩa các ontology.

+Ví dụ: Registry Ontology, dùng để quản lý các ontology khác.

* Domain ontology (ontology lĩnh vực).

+Từ vựng của các khái niệm chỉ trong một phạm vi vừa và nhỏ.

+Ví dụ: ontology về lý thuyết hoặc các nguyên lý cơ bản của một miền.

* Task ontology (ontology tác vụ).

+Hệ thống các từ vựng có thể cùng hoặc không cùng phạm vi ứng dụng cụ thể.

* Domain – task ontology (ontology lĩnh vực – tác vụ).
* Application ontology (ontology ứng dụng).

### 2.3.4 Vai trò và lợi ích sử dụng Ontology

* Chia sẻ kiến thức chung giữa con người hoặc những tác tử phần mềm với nhau
* Có thể tái sử dụng kiến thức về một lĩnh vực.
* Làm rõ ràng những giả định thuộc chuyên ngành.
* Có thể phân tích và suy luận kiến thức chuyên ngành.
  + Ví dụ: Con người có thể đọc được dữ liệu với cấu trúc của XML dễ dàng.
* Biểu diễn tri thức phân tán giữa một số trang web.
* Cung cấp các thông tin giàu ngữ nghĩa.

### 2.3.5 Ontology và các phương pháp biểu diễn

**Bước 1**: Xác định lĩnh vực và phạm vi của Ontology.

* Ontology miêu tả lĩnh vực nào?
* Ontology phục vụ cho mục đích gì?
* Cơ sở tri thức trong Ontology sẽ trả lời những câu hỏi gì?
* Ontology đã có đủ thông tin để trả lời các câu hỏi được quan tâm trên cơ sở tri thức hay không?

**Bước 2**: Xem xét việc sử dụng lại các ontology có sẵn.

Giảm thiểu công sức xây dựng 1 ontology.

Khi tương tác giữa các ứng dụng với nhau, các ứng dụng cần hiểu các lớp, thực thể, quan hệ, … của nhau để trao đổi thông tin.

**Bước 3**: Liệt kê các thuật ngữ quan trọng.

Liệt kê các thuật ngữ xuất hiện trong miền quan tâm.

Các thuật ngữ là danh từ sẽ trở thành các lớp, danh từ sẽ trở thành thuộc tính, động từ là quan hệ giữa các lớp.

**Bước 4**: Xác định các lớp và phân cấp của các lớp

Định nghĩa các lớp từ một số thuật ngữ đã liệt kê ở bước 3

Xây dựng cấu trúc phân lớp, phân cấp theo quan hệ lớp cha-lớp con.

**Bước 5**: Xác định các thuộc tính.

Định nghĩa thuộc tính của lớp bên trong. Có 2 loại:

* Thuộc tính đơn: các giá trị đơn: chuỗi, số, …
* Thuộc tính phức: có thể tham khảo đến một đối tượng khác.

**Bước 6**: Xác định ràng buộc của các thuộc tính.

Ràng buộc quan trọng nhất đối với một thuộc tính:

* Lượng số: quy địnhsố giá trị mà 1 thuộc tính có thể nhận
* Kiểu: các kiểu mà thuộc tính có thể nhận là chuỗi, số, liệt kê, kiểu thực thế,

**Bước 7**: Tạo các thể hiện / thực thể.

Tạo thực thể cho mỗi lớp và gán giá trị cho các thuộc tính.

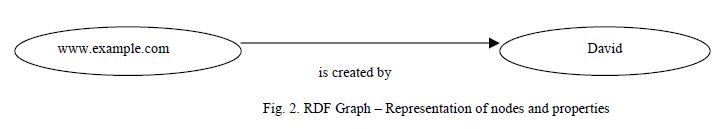
### 2.3.6 Các ngôn ngữ đặc tả Ontology

#### a. RDF

RDF (Resource Description Framework) làm cho thông tin có ngữ nghĩa như là những thông tin máy tính có thể truy cập.Nó là một mô hình chuẩn cho trao đổi dữ liệu trên web. RDF có nhiều tính năng để tạo điều kiện sát nhập dữ liệu ngay cả khi các lược đồ cơ bản khác nhau, và đặc biệt là hỗ trợ sự phát triển của lược đồ theo thời gian.

RDF mở rộng cấu trúc liên kết của trang web sử dụng URI để đặt tên cho mối quan hệ giữa các sự vật cũng như hai đầu của đường link (điều này thường được gọi là "triple"). Sử dụng mô hình đơn giản này, nó cho phép dữ liệu có cấu trúc và bán cấu trúc được pha trộn, tiếp xúc, và chia sẻ trên các ứng dụng khác nhau.

Cấu trúc liên kết này tạo thành một định hướng, gán nhãn đồ thị, khi đó các cạnh đại diện cho các liên kết có tên giữa hai nguồn tài nguyên (đại diện bởi các nút đồ thị). Biểu đồ này xem là mô hình đơn giản nhất có thể cho RDF ​​[3] và thường được sử dụng trong giải thích dễ hiểu trực quan.



*Hình 5: Ví dụ về một khái niệm biểu diễn bằng đồ thị RDF*

**Các thành phần node và các thành phần thuộc tính** (*Node elements and property elements)*

Một đồ thị RDF được đưa ra trong hình, nơi mà các nút được biểu diễn như là hình bầu dục biểu thị các lớp và cạnh đại diện cho các thuộc tính.Lớp và các thuộc tính có thể là một số tài nguyên khác.

**Hạn chế của sức mạnh ý nghĩa của RDF Schema**

RDF và RDFS cho phép các đại diện của một số kiến ​​thức ontology. Nguyên thủy mô hình chính của RDF / RDFS liên quan đến việc tổ chức từ vựng trong hệ thống phân cấp kiểu: phân lớp và thay thế các mối quan hệ thuộc tính, miền và giới hạn phạm vi, và các trường hợp của các lớp. Tuy nhiên một số tính năng khác bị mất. Đó là:

* Phạm vi cục bộ của thuộc tính: RDF: phạm vi biểu thị phạm vi của một thuộc tính, xem xét câu "Animal eat plants". Quy tắc này được áp dụng cho tất cả các loại động vật. Vì vậy, trong RDF Schema nó không thể áp dụng chỉ một giới hạn phạm vi đến các lớp nhất định. Ví dụ theo quy tắc trước nói rằng con bò chỉ ăn thực vật, trong khi các động vật khác có thể ăn thịt là một câu không hợp lệ.
* Tính tách biệt của các lớp: Đôi khi cần phải nói rằng các lớp thường tách rời. Ví dụ, nam và nữ thường tách rời. Nhưng trong RDF Schema được sử dụng để đại diện cho mối quan hệ phân lớp ví dụ nữ là lớp con của người.
* Kết hợp logic của các lớp: Đôi khi có một nhu cầu để xây dựng lớp mới bằng cách kết hợp các lớp khác sử dụng phép hợp, phép giao và phần bù. Ví dụ, biểu thị người lớp để được hợp của các lớp tách rời là nam và nữ. RDF Schema không cho phép định nghĩa như vậy.
* Hạn chế cardinality: Hạn chế về bao nhiêu giá trị khác nhau một thuộc tính có thể hoặc phải được biểu diễn như là tỷ lệ cardinality. Ví dụ, một người có đúng hai cha mẹ, và một khóa học được giảng dạy bởi ít nhất một giảng viên. Một lần nữa, những hạn chế đó là không thể thể hiện trong RDF Schema.
* Đặc điểm của thuộc tính: Đôi khi nó rất hữu ích để cho rằng một thuộc tính là bắc cầu (như "greater than"), độc đáo (như "is mother of"), hoặc nghịch đảo của thuộc tính khác (như "eats" và "is eaten by"). RDF không cho phép xác định các loại tính năng.

#### b. OWL

OWL (The Web Ontology Language) là một ngôn ngữ gần như XML dùng để mô tả các hệ cơ sở tri thức.OWL là một ngôn ngữ đánh dấu dùng để xuất bản và chia sẻ dữ liệu trên Internet thông qua những mô hình dữ liệu gọi là “ontology”.Ontology mô tả một lĩnh vực (domain) và diễn tả những đối tượng trong lĩnh vực đó cùng những mối quan hệ giữa các đối tượng này. OWL là phần mở rộng về từ vựng của RDF và được kế thừa từ ngôn ngữ DAML+OIL Web ontology – một dự án được hỗ trợ bởi W3C. OWL biểu diễn ý nghĩa của các thuật ngữ trong các từ vựng và mối liên hệ giữa các thuật ngữ này để đảm bảo phù hợp với quá trình xử lý bởi các phần mềm.

OWL được xem như là một kỹ thuật trọng yếu để cài đặt cho Semantic Web trong tương lai. OWL được thiết kế đặc biệt để cung cấp một cách thức thông dụng trong việc xử lý nội dung thông tin của Web. Ngôn ngữ này được kỳ vọng rằng sẽ cho phép các hệ thống máy tính có thể đọc được thay thế cho con người. Vì OWL được viết bởi XML, các thông tin OWL có thể dễ dàng trao đổi giữa các kiểu hệ thống máy tính khác nhau, sử dụng các hệ điều hành và các ngôn ngữ ứng dụng khác nhau.Mục đích chính của OWL là sẽ cung cấp các chuẩn để tạo ra một nền tảng để quản lý tài sản, tích hợp mức doanh nghiệp và để chia sẻ cũng như tái sử dụng dữ liệu trên Web. OWL được phát triển bởi nó có nhiều tiện lợi để biểu diễn ý nghĩa và ngữ nghĩa hơn so với XML, RDF và RDFS, và vì OWL ra đời sau các ngôn ngữ này, nó có khả năng biểu diễn các nội dung mà máy có thể biểu diễn được trên Web.

Các phiên bản của OWL: Hiện nay có ba loại OWL : OWL Lite, OWL DL (description logic), và OWL Full.

* *OWL Lite*: hỗ trợ cho những người dùng chủ yếu cần sự phân lớp theo thứ bậc và các ràng buộc đơn giản. Ví dụ: Trong khi nó hỗ trợ các ràng buộc về tập hợp, nó chỉ cho phép tập hợp giá trị của 0 hay 1. Điều này cho phép cung cấp các công cụ hỗ trợ OWL Lite dễ dàng hơn so với các bản khác.
* *OWL DL (OWL Description Logic*): hỗ trợ cho những người dùng cần sự diễn cảm tối đa trong khi cần duy trì tính tính toán toàn vẹn (tất cả các kết luận phải được đảm bảo để tính toán) và tính quyết định (tất cả các tính toán sẽ kết thúc trong khoảng thời gian hạn chế). OWL DL bao gồm tất cả các cấu trúc của ngôn ngữ OWL, nhưng chúng chỉ có thể được sử dụng với những hạn chế nào đó (Ví dụ: Trong khi một lớp có thể là một lớp con của rất nhiều lớp, một lớp không thể là một thể hiện của một lớp khác). OWL DL cũng được chỉ định theo sự tương ứng với logic mô tả, một lĩnh vực nghiên cứu trong logic đã tạo nên sự thiết lập chính thức của OWL.
* *OWL Full* muốn đề cập tới những người dùng cần sự diễn cảm tối đa và sự tự do của RDF mà không cần đảm bảo sự tính toán của các biểu thức. Ví dụ, trong OWL Full, một lớp có thể được xem xét đồng thời như là một tập của các cá thể và như là một cá thể trong chính bản thân nó. OWL Full cho phép một ontology gia cố thêm ý nghiã của các từ vựng được định nghĩa trước (RDF hoặc OWL).
* Các phiên bản này tách biệt về các tiện ích khác nhau, OWL Lite là phiên bản dễ hiểu nhất và phức tạp nhất là OWL Full.

**Mối liên hệ giữa các ngôn ngữ con của OWL:**

* Mọi ontology hợp lệ dựa trên OWL Lite đều là ontology hợp lệ trên OWL DL.
* Mọi ontology hợp lệ dựa trên OWL DL đều là ontology hợp lệ trên OWL Full.
* Mọi kết luận hợp lệ dựa trên OWL Lite đều là kết luận hợp lệ trên OWL DL.
* Mọi kết luận hợp lệ dựa trên OWL DL đều là kết luận hợp lệ trên OWL Full.

[CHƯƠNG 3: THIẾT KẾ CƠ SỞ TRI THỨC](#page28) [27](#page28)

3.1. [THU THẬP VÀ PHÂN LOẠI TRI THỨC](#page28) [27](#page28)

3.1.1. [Miền tri thức thu thập](#page28) [27](#page28)

3.1.2. [Nguồn thu thập](#page28) [27](#page28)

3.1.3. [Cách thức thu thập](#page29) [28](#page29)

3.1.4. [Kết quả thu thập](#page30) [29](#page30)

3.2. [MÔ HÌNH TRI THỨC CỦA ỨNG DỤNG](#page52) [51](#page52)

3.2.1. [Tập khái niệm C](#page53) [52](#page53)

3.2.2. [Tập Rela bao gồm các quan hệ giữa các tập khái niệm C](#page54) [53](#page54)

1. [Tập Rule bao gồm các luật trên các sự kiện liên quan đến các khái niệm C:](#page58) [57](#page58)

|  |  |  |
| --- | --- | --- |
| [3.3.](#page58) | [TỔ CHỨC LƯU TRỮ TRI THỨC TRÊN MÁY TÍNH ...........................](#page58) | [57](#page58) |
| [CHƯƠNG 4: THIẾT KẾ BỘ SUY DIỄN .........................................................................](#page61) | | [60](#page61) |
| [4.1.](#page61) | [MÔ HÌNH HÓA BÀI TOÁN ....................................................................](#page61) | [60](#page61) |
| [4.2.](#page62) | [PHÂN LOẠI SỰ KIỆN VÀ SO KHỚP SỰ KIỆN ...................................](#page62) | [61](#page62) |
| [4.3.](#page62) | [CHIẾN LƯỢC SUY DIỄN ........................................................................](#page62) | [61](#page62) |
| [Chương 5: ỨNG DỤNG – HỆ THỐNG CHẨN ĐOÁN BỆNH TAI – MŨI – HỌNG ...](#page64) | | [63](#page64) |
|  | 2 |  |

5.1. [THIẾT KẾ HỆ THỐNG](#page64) [63](#page64)

5.1.1. [Mục tiêu ứng dụng](#page64) [63](#page64)

5.1.2. [Yêu cầu và chức năng của hệ thống](#page64) [63](#page64)

5.1.3. [Kiến trúc của hệ thống](#page65) [64](#page65)

5.2.1. [Nền tảng công nghệ và công cụ](#page69) [68](#page69)

5.2.2. [Tổ chức các giao diện](#page70) [69](#page70)

5.2.3. [Các hàm xử lý chính](#page78) [77](#page78)

5.3. [THỬ NGHIỆM](#page85) [84](#page85)

5.3.1. [Cách thức thử nghiệm](#page85) [84](#page85)

5.3.2. [Kết quả thử nghiệm](#page86) [85](#page86)

5.3.3. [Nhận xét và đánh giá](#page90) [89](#page90)

[Chương 6: KẾT LUẬN](#page91) [90](#page91)

6.1. [Kết quả của đề tài](#page91) [90](#page91)

6.2. [Hạn chế của đề tài](#page91) [90](#page91)

6.3. [Hướng phát triển của đề tài](#page91) [90](#page91)

[Tài liệu tham khảo](#page92) [91](#page92)