Tạp chí Tin học và Điều khiển học, T. 15, S. 3 (1999), 1–7

MỘT SỐ TÍNH CHẤT CỦA QUÁ TRÌNH THỪA KẾ KIỂU TRONG MÔ HÌNH CƠ SỞ DỮ LIỆU HƯỚNG ĐỐI TƯỢNG

ĐOÀN VĂN BAN

Abstract. Inheritance is introduced in object-oriented systems to enhance code reuse and create more compact and openable software [2], [3], [7]. Powerful object models adopt multiple inheritance, allowing a type (or class) to inherit from more than one supertype. Unfortunately, this powerful modeling mechanism can generate inheritance conflicts [6], which arise when the same property (or operation) is defined in more than one supertype, or a property (or operation) already present in one supertype, is locally redefined in subtype (overriding). In this article, we focus on the structural component of object-oriented database schema, addressing inheritance and conflict resolution for typed properties.

1. GIỚI THIỆU

Cơ chế thừa kế được sử dụng trong các hệ thống hướng đối tượng nhằm hỗ trợ để tạo ra những phần mềm cô đọng, có tính mở và sử dụng lại nhiều hơn [2], [3], [7]. Những mô hình đối tượng mạnh đều chấp nhận cơ chế thừa kế bội, cho phép một kiểu (hay lớp) thừa kế từ nhiều hơn một kiểu cơ sở. Nhưng chính từ cơ chế này lại có thể tạo ra những xung đột giữa các kiểu thừa kế [1]. Trong bài báo này chúng tôi tập trung nghiên cứu một số tính chất cơ bản của quá trình thừa kế kiểu thông qua mối quan hệ của chúng trong lược đồ cơ sở dữ liệu đối tượng. Dựa vào những tính chất đó chúng ta có thể đề xuất được những giải pháp giải quyết được vấn đề xung đột trong hệ thống kiểu của lược đồ.

2. MÔ HÌNH DỮ LIỆU CỦA CƠ SỞ DỮ LIỆU ĐỐI TƯỢNG

Trong một hệ cơ sở dữ liệu (CSDL), đối tượng là một thực thể (người, vật, sự kiện, hay một khái niệm, v.v...) được xác định duy nhất thông qua phần định danh của các đối tượng OID (Object IDentifier). OID được gán cho các đối tượng bởi hệ quản trị CSDL và sẽ không thay đổi trong suốt thời gian tồn tại của các đối tượng [5], [6].

Các đối tượng được mô tả bởi tập đặc trưng bao gồm cả dữ liệu, những thuộc tính mô tả các tính chất của các thực thể và cả các thao tác (hay còn gọi là hàm hoặc phương thức) mô tả hành vi ứng xử của các đối tượng và được tổ chức theo kỹ thuật bao gói (encapsulation) và che giấu thông tin (information hiding) [3].

Theo cách tiếp cận hướng đối tượng, hệ thống phần mềm được xem như là tập hợp các đối tượng tác động qua lại với nhau, trao đổi với nhau bằng các thông báo để thực hiện nhiệm vụ của hệ thống [2].

Những đối tượng có cùng tập các thuộc tính và hành vi giống nhau, nghĩa là có cùng không gian trạng thái và cùng cách ứng xử, sẽ thuộc trong cùng một kiểu (hay một lớp). Trong hệ thống, đối tượng là thể hiện của một kiểu nào đó.

Một trong những đặc tính quan trọng nhất của phương pháp hướng đối tượng là khả năng thừa kế và sử dụng lại. Một kiểu có thể thừa kế những đặc trưng của một hay nhiều kiểu cơ sở (lớp cha) và có thể được bổ sung một số đặc trưng mới theo yêu cầu. Nguyên nhân chính dẫn đến việc ứng dụng khái niệm thừa kế trong công nghệ phần mềm giống như luật thừa kế gia sản của xã hội loài người là tạo khả năng gộp, nhóm một số đặc trưng của hai hay nhiều kiểu đã được xây dựng tốt để tạo lập những kiểu cho những đối tượng mới đáp ứng được tính sử dụng lại trong quá trình phát triển phần mềm. Kiểu cơ sở để thừa kế thường được gọi là kiểu cha còn kiểu thừa kế thường

được gọi là kiểu dẫn xuất, hay là kiểu con. Quan hệ thừa kế sẽ tạo thành quan hệ thứ tự bộ phận, trong đó đối tượng của một kiểu sẽ là thể hiện của chính kiểu đó nhưng đồng thời có thể xem nó cũng là thể hiện của các kiểu cha, của những kiểu cơ sở trên nữa [1], [2], [7].

Sau đây, chúng ta nêu một số định nghĩa để hình thức hóa những khái niệm nêu trên và chủ yếu chỉ tập trung xét những cấu trúc dữ liệu trong các CSDL đối tượng.

Trước tiên chúng ta sử dụng một số ký hiệu như sau:

T - là tập hữu hạn các kiểu,

L - tập các kiểu nguyên thủy như những kiểu integer (số nguyên), real (số thực), char (ký tự), string (xâu ký tự), boolean (giá trị logic),

P - tập tất cả các thuộc tính của các đối tượng trong hệ thống,

O - tập tất cả các đối tượng và các đại lượng trong một CSDL.

Định nghĩa 1. *Quan hệ Sub thừa kế kiểu*

Cho trước các kiểu $\tau_i \in T$, $i = 1, ..., n$ và $\delta_l \in T \cup L$, $l = 1, 2, ..., k$. Kiểu τ thừa kế (trực tiếp) từ những kiểu τ_i và được bổ sung một số thuộc tính trong các kiểu δ_l sẽ được định nghĩa như sau:

$$\text{type} \quad \tau = \tau_1, \tau_2, \ldots, \tau_n \{p_1 : \delta_1; \ p_2 : \delta_2; \ldots ; \ p_k : \delta_k\}, \tag{1}$$

trong đó $p_l \in P$ với mọi $l = 1, 2, ..., k$. Chúng ta nói rằng kiểu τ thừa kế từ các kiểu τ_i, $i = 1, ..., n$ hay ngược lại τ_i là cha (trực tiếp) của τ và được ký hiệu τ Sub τ_i.

Trong hệ thống có những kiểu nguyên thủy hoặc được tạo ra bởi người sử dụng, không theo quan hệ thừa kế Sub, những kiểu đó chúng ta gọi là kiểu chuẩn. Kiểu chuẩn có thể là kiểu nguyên thủy hoặc kiểu được tạo lập như sau:

$$\text{type} \quad \tau = \{p_1 : \delta_1; \ p_2 : \delta_2; \ldots ; \ p_k : \delta_k\}, \tag{2}$$

trong đó $\delta_l \in R \cup L$, $p_l \in P$, $l = 1, 2, ..., k$.

Gọi N là tập tất cả các kiểu chuẩn. Hiển nhiên là $L \subseteq N$.

Chúng ta dễ dàng nhận thấy quan hệ Sub có tính phản xạ: τ Sub τ, với mọi $\tau \in T$. Từ đó chúng ta có thể xây dựng bao đóng bắc cầu của quan hệ cha/con (Sub) và ký hiệu là $\leq_1$, gọi là quan hệ thừa kế kiểu.

Định nghĩa 2. *Quan hệ thừa kế kiểu $\leq_1$ được định nghĩa như sau:*

(i) $\tau \leq_1 \tau$ với mọi $\tau \in T$,

(ii) $\tau_1, \tau_2, \tau_3 \in T$, τ_1 Sub τ_2 và $\tau_2 \leq_1 \tau_3$ thì $\tau_1 \leq_1 \tau_3$, (3)

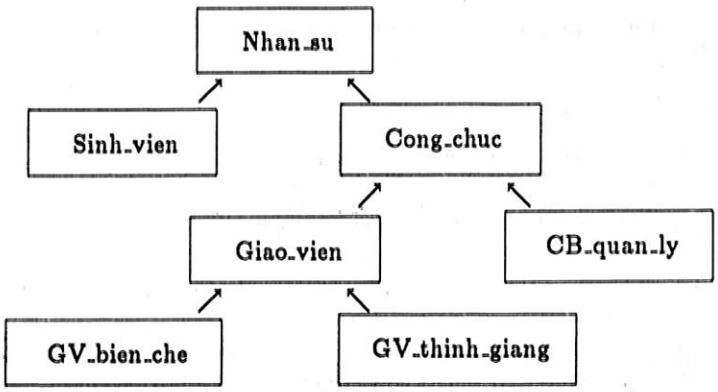
(iii) $\tau_1 \leq_1 \tau_2$ thì $\{a_i : \tau_1\} \leq_1 \{a_i : \tau_2\}$.

Khi đó $(T, \leq_1)$ sẽ tạo thành cấu trúc phân cấp theo quan hệ thừa kế. Dựa trên cấu trúc phân cấp này chúng ta có thể xác định được tập các kiểu cơ sở (cha, ông, bà, v.v...) của một kiểu dẫn xuất (thừa kế) thông qua hàm Sup: $T \to T$ được xác định như sau:

$$\text{Sup}(\tau) = \begin{cases} \bigcup_{i=1}^{n} \{\text{Sup}(\tau_i)\} & \text{nếu } \tau \text{ là kiểu được định nghĩa dạng (1)} \\ \emptyset & \text{ngược lại} \end{cases}$$

Như vậy Sup(τ) xác định tất cả các kiểu cơ sở của τ trong quan hệ thừa kế.

Ví dụ 1. Xét hệ thống các lớp (kiểu) của hệ thống quản lý nhân sự trong các trường đại học được mô tả như sau:



Cấu trúc phân cấp theo quan hệ thừa kế giữa các lớp

Các cạnh theo mũi tên $\rightarrow$ xác định theo quan hệ thừa kế trực tiếp (Sub) giữa các kiểu, những cạnh trên đường đi của cấu trúc phân cấp trên đều có quan hệ $\leq_1$ với nhau. Ngoài ra, chúng ta còn xác định được các kiểu cơ sở của tất cả các kiểu Sup(Nhan_su) = $\emptyset$, Sup(GV_bien_che) = {Nhan_su, Cong_chuc, Giao_vien}, v.v...

Trên cấu trúc phân cấp các kiểu chúng ta cũng có thể xác định được tập các thể hiện (đối tượng) của một kiểu nào đó dựa trên hai hàm sau:

o_type: $O \rightarrow (T \cup L)$ cho biết kiểu của một thực thể trong hệ thống,

Ext: $T \rightarrow O$ cho biết các thể hiện của một kiểu sao cho

$$\text{Ext}(\tau) = \{o \in O \mid \text{0\_type}(o) \leq_1 t\}. \tag{4}$$

Một số tính chất mô tả ngữ nghĩa tự nhiên của quan hệ $\leq_1$ được phát biểu như sau:

Bổ đề 1. $\forall \tau, \tau_1, \tau_2 \in T$

(i) $o\text{\_type}(o) = \tau \Rightarrow o \in \text{Ext}(\tau)$,

(ii) $o\text{\_type}(o) \leq_1 \tau \Leftrightarrow o \in \text{Ext}(\tau)$,

(iii) $\text{Ext}(\tau_1) \subseteq \text{Ext}(\tau_2) \Leftrightarrow \tau_1 \leq_1 \tau_2$.

Chứng minh.

(i) và (ii) suy trực tiếp từ (4).

(iii) Nếu $\text{Ext}(\tau_1) \subseteq \text{Ext}(\tau_2)$, nghĩa là $\forall o \in \text{Ext}(\tau_1)$ thì $o \in \text{Ext}(\tau_2)$. Do vậy với những đối tượng o_1 mà $o\text{\_type}(o_1) = \tau_1$ thì hiển nhiên là $o_1 \in \text{Ext}(\tau_2)$ và suy ra $o\text{\_type}(o_1) = \tau_1 \leq_1 \tau_2$. Trường hợp ngược lại suy trực tiếp từ định nghĩa. □

Cấu trúc phân cấp các kiểu như trên đóng vai trò rất quan trọng trong quá trình thiết kế và xây dựng những hệ thống hướng đối tượng [3, 7] và trong việc xây dựng những cấu trúc dữ liệu cho các CSDL đối tượng [4, 8].

Định nghĩa 3. *Lược đồ CSDL đối tượng*

Lược đồ CSDL hướng đối tượng (gọi tắt là lược đồ đối tượng) là một bộ ba $\Sigma = \langle T, L, P \rangle$, trong đó T là tập các kiểu, L - tập các kiểu nguyên thủy và P - tập các thuộc tính của tất cả các đối tượng thỏa mãn hai tính chất sau:

(i) Tất cả những kiểu $\tau \in T$ phải được định nghĩa duy nhất, nghĩa là xuất hiện đúng một lần ở vế trái trong các định nghĩa kiểu của T.

(ii) Với mọi kiểu $\tau \in T$, $\tau \notin \text{Sup}(\tau)$, nghĩa là quan hệ thừa kế không tạo thành chu trình.

(iii) Không có các bộ lồng nhau trong định nghĩa kiểu, mọi kiểu thừa kế và khai báo thuộc tính phải thông qua tên gọi kiểu.

Định nghĩa 4. Lược đồ đối tượng được gọi là lược đồ đối tượng dạng chuẩn nếu tất cả các kiểu $\tau \in T$ đều ở dạng chuẩn ($\tau \in N$).

Lược đồ đối tượng dạng chuẩn thì $T = N$.

Ví dụ 2. Xét lược đồ gồm những thành phần như sau:

type Giao_vien = {Ho_ten: string; Dien_thoai: integer; Truong: string};
type Cong_chuc = {Ho_ten: string; Dien_thoai: string; Luong: real};
type Vien_chuc = Giao_vien, Cong_chuc {dia_chi: string}.

Trong lược đồ trên, kiểu Giao_vien, Cong_chuc là ở dạng chuẩn còn Vien_chuc được xây dựng dựa trên quan hệ thừa kế Sub và không ở dạng chuẩn.

Quá trình xác định các thành phần của những kiểu thừa kế từ các kiểu cha có thể dẫn tới xung đột hay không nhất quán về kiểu. Lược đồ mà có những kiểu dẫn tới xung đột chúng ta gọi là không nhất quán. Ví dụ kiểu Vien_chuc sẽ được xác định thông qua Giao_vien và Cong_chuc. Vấn đề sẽ nảy sinh khi kiểu dẫn xuất thừa kế cùng tên thuộc tính như Dien_thoai nhưng kiểu tương ứng ở các kiểu cha lại không tương thích, như Dien_thoai ở Giao_vien có kiểu integer còn ở Cong_chuc lại khai báo kiểu string. Tất nhiên kiểu integer và string là không tương thích cho nên lược đồ ở ví dụ 2 là không nhất quán. Giá trị của những thuộc tính có kiểu không nhất quán dẫn ra từ quan hệ thừa kế là không xác định và chúng ta ký hiệu là $\perp$. Khi đó Vien_chuc có thể viết lại như sau:

type Vien_chuc = {Ho_ten: string; Dien_thoai: $\perp$; Luong: real; Dia_chi: string}.

Lược đồ trên sẽ trở thành nhất quán nếu chúng ta khai báo thuộc tính Dien_thoai của Giao_vien và Cong_chuc cùng một kiểu, ví dụ interger. Từ đó chúng ta dễ nhận thấy là những lược đồ nhất quán có thể chuyển đổi thành lược đồ dạng chuẩn nếu mọi kiểu $\tau \in T$ không ở dạng chuẩn đều có thể chuyển về dạng chuẩn sau một hữu hạn lần áp dụng quan hệ thừa kế để xác định tường minh các thành phần của τ.

Quá trình xác định tường minh các thành phần của các kiểu trong quan hệ thừa kế dẫn tới việc xác định kiểu con chung lớn nhất của các kiểu cha chúng. Một số thuộc tính của kiểu này có thể qui chiếu tới một số kiểu khác và quá trình dẫn xuất để tìm kiểu con chung lớn nhất có thể qui chiếu qua lại mà không kết thúc được.

Ví dụ 3. Xét lược đồ sau:

type Nguoi_lon = {Ho_ten: string; Tuoi: interger; Ban_huu: Cong_nhan};
type Cong_nhan = {Ho_ten: string; Co_quan: string; Ban_huu: Nguoi_lon};
type Can_bo = {Ho_ten: string; Luong: interger; Ban_huu: Can_bo};
type Nhan_vien = Cong_nhan, Can_bo {Dia_chi: string}.

Việc xác định tường minh các thành phần của Nhan_vien phải dựa vào các thành phần của Cong_nhan và Can_bo. Ở Cong_nhan thuộc tính Ban_huu qui chiếu tới Nguoi_lon, tiếp theo ở Nguoi_lon thuộc tính Ban_huu lại qui chiếu về Cong_nhan. Quá trình trên có thể lặp lại nhiều lần và rất khó khẳng định tính nhất quán của lược đồ đối tượng.

Vấn đề đệ qui trong định nghĩa kiểu đã được nghiên cứu trong [1], các tác giả đã xét một số tính chất quan trọng của quá trình thừa kế có đệ qui và điều kiện để kết thúc thừa kế.

Để xây dựng được những hệ thống CSDL đối tượng tin cậy, nhất quán, chúng ta phải nghiên cứu để đề xuất phương pháp kiểm tra tính đúng đắn, nhất quán dữ liệu lược đồ tương ứng. Trước hết chúng phải xác định được mối quan hệ giữa các lớp đối tượng trong hệ thống. Phần tiếp theo chúng ta chủ yếu tập trung xét các tính chất của quá trình thừa kế kiểu.

3. QUAN HỆ THỨ TỰ KIỂU CON

Chúng ta xây dựng quan hệ kiểu con $\leq$ giữa các kiểu có quan hệ thừa kế để nghiên cứu các tính chất của tất cả các thành phần đối với mọi kiểu trong hệ thống.

Định nghĩa 5. Cho trước lược đồ $\Sigma = \langle T, L, P\rangle$, trên T xác định một quan hệ $\leq$ như sau:

(i) $\forall \tau \in T \cup L,\ \tau \leq \tau$,

(ii) Nếu $\tau_1 = \{a_i : \eta_i;\ c_k : \gamma_k\}$ và $\tau_2 = \{a_i : \mu_i\}$ thì $\tau_1 \leq \tau_2 \Leftrightarrow \eta_i \leq \mu_i$ (5)

Nếu $\tau_1 \leq \tau_2$ thì chúng ta nói rằng τ_1 là kiểu con của τ_2.

Ví dụ 4.

type Nhan_su = {Ho_ten: string; Dien_thoai: integer},

type Giao_vien = {Ho_ten: string; Dien_thoai: integer; Truong: string}.

Hiển nhiên là Giao_vien $\leq$ Nhan_su.

Để chứng minh các tính chất của quan hệ kiểu con $\leq$ chúng ta có thể dựa vào quan hệ thứ tự suy diễn $\leq^{(n)}$, trong đó n là số nguyên dương.

Định nghĩa 6. Quan hệ suy diễn $\leq^{(n)}$ được định nghĩa như sau:

(i) $\tau \leq^{(n)} \tau$, với mọi $\tau \in T \cup L$ và $n \geq 0$,

(ii) Nếu $\tau_1 = \{a_i : \eta_i; c_k : \gamma_k\}$ và $\tau_2 = \{a_i : \mu_i\}$ thì với mọi $n \geq 0$,
$\tau_1 \leq^{(n)} \tau_2 \Leftrightarrow \forall i\ \exists n_i,\ 0 \leq n_i < n$ sao cho $\eta_i \leq^{(n_i)} \mu_i$. (6)

Định lý 1. *Hai quan hệ $\leq$ và $\leq^{(n)}$ là tương đương*

$$\forall \tau_1, \tau_2 \in T,\ \tau_1 \leq \tau_2 \Leftrightarrow \exists n \geq 0,\ \tau_1 \leq^{(n)} \tau_2. \tag{7}$$

Chứng minh.

Điều kiện cần: qui nạp theo số lần m áp dụng điều kiện (5).

Khi $m = 0$, nghĩa là $\tau_1 = \tau_2$. Vậy từ điều kiện (i) của các định nghĩa 5, 6 chúng ta có ngay $\tau_1 \leq \tau_1 \Rightarrow \tau_1 \leq^{(n)} \tau_1$. Giả thiết điều kiện cần đúng với $0 \leq k < m$, chúng ta cần chứng minh nó đúng với m.

Thật vậy, nếu $\tau_1 = \{a_i : \eta_i; c_k : \gamma_k\}$, $\tau_2 = \{a_i : \mu_i\}$ và $\tau_1 \leq \tau_2$ thì theo (5) suy ra $\eta_i \leq \mu_i$. Tất nhiên là $\eta_i \leq \mu_i$ sẽ có k_i lần áp dụng điều kiện (5) và $k_i < m$ (vì $k_i \leq m - 1$). Theo giả thiết qui nạp sẽ tồn tại $\eta_i \geq 0$ sao cho $\eta_i \leq^{(n_i)} \mu_i$. Đặt $n = \max\{n_i\} + 1$ ta sẽ có ngay $\tau_1 \leq^{(n)} \tau_2$.

Điều kiện đủ: qui nạp theo n.

Khi $n = 0$, $\tau_1 = \tau_2$ thì là hiển nhiên. Giả thiết điều kiện đủ đúng với $0 \leq k < n$, chúng ta chứng minh nó đúng với n.

Thật vậy, $\tau_1 \leq^{(n)} \tau_2$ và $\tau_1 = \{a_i : \eta_i; c_k : \gamma_k\}$, $\tau_2 = \{a_i : \mu_i\} \Rightarrow \forall i\ \exists n_i,\ 0 \leq n_i < n$ sao cho $\eta_i \leq^{(n_i)} \mu_i$. Theo giả thiết qui nạp ra $\eta_i \leq \mu_i$ và theo (5) chúng ta có ngay $\tau_1 \leq \tau_2$.

Bổ đề 2.

$$\forall \tau_1, \tau_2 \in T,\ \tau_1 \leq \tau_2 \Rightarrow \tau_1 \leq_1 \tau_2 \tag{8}$$

Chứng minh. $\forall \tau_1, \tau_2 \in T$, $\tau_1 \leq \tau_2$, vậy theo (7) ta có $\tau_1 \leq^{(n)} \tau_2$. Tương tự như trên chúng ta có thể chứng minh $\tau_1 \leq^{(n)} \tau_2 \Rightarrow \tau_1 \leq_1 \tau_2$ qui nạp theo n. □

Từ các bổ đề 1 và 2 chúng ta có khẳng định sau

Định lý 2.

$$\forall \tau_1, \tau_2 \in T,\ \tau_1 \leq \tau_2 \Leftrightarrow \mathrm{Ext}(\tau_1) \subseteq \mathrm{Ext}(\tau_2). \tag{9}$$

Như vậy kiểu con có phạm vi xác định hẹp hơn phạm vi của những kiểu ở mức cao hơn. Điều này phù hợp với ngữ nghĩa của quan hệ thứ tự tự nhiên của kiểu dữ liệu trong hệ CSDL.

Như trên chúng ta đã xét, quan hệ kiểu con $\leq$ tạo ra cấu trúc phân cấp các kiểu trong hệ thống. Vấn đề đặt ra đó có thể tạo thành thứ tự bộ phận hay không? Để trả lời câu hỏi đó chúng ta xét tính chất sau.

Bổ đề 3. *Với mọi số nguyên* n, m *và* $\forall \tau_1, \tau_2, \tau_3 \in T$.

(i) $\tau_1 \leq^{(n)} \tau_2$ *và* $\tau_2 \leq^{(m)} \tau_1 \Rightarrow \tau_1 = \tau_2$. (10)

(ii) $\tau_1 \leq^{(n)} \tau_2$ *và* $\tau_2 \leq^{(m)} \tau_3 \Rightarrow \tau_1 \leq^{(r)} \tau_3$, *trong đó* $r = \max\{m, n\}$. (11)

(iii) $\tau_1 \leq^{(n)} \tau_2 \Rightarrow \tau_1 \leq^{(m)} \tau_2$ *với mọi* $m > n$. (12)

Chứng minh. Tính chất (i) và (ii) có thể chứng minh qui nạp theo $r = \max\{m, n\}$.
Khi $r = 0$ thì cả hai tính chất trên đều đúng (suy trực tiếp từ định nghĩa). Giả thiết chúng đúng với mọi $k < r$, cần chứng minh cũng đúng với r.

(i) Khi $n = 0$ hoặc $m = 0$ thì có ngay $\tau_1 = \tau_2$. Vậy chúng ta chỉ xét trường hợp $n > 0$ mà $m > 0$, và $r = \max\{n, m\}$. Từ (6) thấy ngay rằng τ_1 và τ_2 chỉ có những thuộc tính chung, nghĩa là $\tau_1 = \{a_i : \eta_i\}$, $\tau_2 = \{a_i : \mu_i\}$. Bởi vì $\tau_1 \leq^{(n)} \tau_2$ nên $\forall i\ \exists n_i,\ 0 \leq n_i < n \leq r$ để $\eta_i \leq^{(n_i)} \mu_i$. Hơn nữa vì $\tau_2 \leq^{(m)} \tau_1$ nên $\forall i\ \exists n_i,\ 0 \leq m_i < m \leq r$ để $\mu_i \leq^{(m_i)} \eta_i$. Theo giả thiết qui nạp ta có $\eta_i = \mu_i$, tức là $\tau_1 = \tau_2$.

(ii) Tương tự như trên chúng ta chỉ xét trường hợp $n > 0$, $m > 0$ và $r = \max\{n, m\}$. Theo định nghĩa thì $\tau_1 = \{a_i : \eta_i;\ b_j : \delta_j;\ c_k : \gamma_k\}$, $\tau_2 = \{a_i : \eta_i';\ b_j : \delta_j'\}$ và $\tau_3 = \{a_i : \eta_i''\}$. Vì $\tau_1 \leq^{(n)} \tau_2 \Rightarrow \forall i, j\ \exists n_i, m_j : 0 \leq n_i < n \leq r,\ 0 \leq m_j < m \leq r$ sao cho $\eta_i \leq^{(n_i)} \eta_i'$, $\delta_j \leq^{(m_j)} \delta_j'$ và $\tau_2 \leq^{(m)} \tau_3 \Rightarrow \forall i\ \exists k_i : 0 \leq k_i < m \leq r$ để cho $\eta_i' \leq^{(k_i)} \eta_i''$.
Như vậy, với mọi i chúng ta có thể chọn $r_i = \max\{n_i, k_i\} < r$ thì có ngay $\eta_i \leq^{(r_i)} \eta_i''$ và từ đó suy ra $\tau_1 \leq^{(r)} \tau_3$.

(iii) được suy ra trực tiếp từ Định nghĩa 5.

Định lý 3. *Quan hệ kiểu con* $\leq$ *là thứ tự bộ phận.*

Chứng minh. Tính phản xạ hiển nhiên vì $\tau \leq \tau$ với mọi $\tau \in T$.

Tính bắc cầu: Nếu $\tau_1 \leq \tau_2$, $\tau_2 \leq \tau_3$ thì theo (7) ta có $\tau_1 \leq^{(n)} \tau_2$ và $\tau_2 \leq^{(m)} \tau_3$, với n, m nguyên dương nào đó. Từ (11) ta có $\tau_1 \leq^{(r)} \tau_3$, với $r = \max\{n, m\}$, nghĩa là $\tau_1 \leq \tau_3$ (Định lý 1).

Tính phản xứng: Nếu $\tau_1 \leq \tau_2$, $\tau_2 \leq \tau_1$ thì theo (7) ta có $\tau_1 \leq^{(n)} \tau_2$ và $\tau_2 \leq^{(m)} \tau_1$, với n, m nguyên dương nào đó và từ (10) suy ra $\tau_1 = \tau_2$. □

4. KẾT LUẬN

Từ cấu trúc phân cấp dựa trên quan hệ thừa kế kiểu $\leq_1$ chúng ta xây dựng được quan hệ thứ tự bộ phận $\leq$ để nghiên cứu những tính chất của hệ thống kiểu trong lược đồ đối tượng của hệ thống hướng đối tượng. Sử dụng những quan hệ nêu trên, chúng ta có thể tiếp tục nghiên cứu để biến đổi tất cả những kiểu không dạng chuẩn về dạng chuẩn đối với những lược đồ phi mâu thuẫn dựa trên việc xây dựng quá trình xác định kiểu con chung nhất của các kiểu cơ sở ở từng mức và nghiên cứu được tính đúng đắn, nhất quán dữ liệu của hệ thống dựa trên đồ thị quan hệ thừa kế hoặc cấu trúc dàn của hệ thống kiểu. Những vấn đề nêu trên chúng tôi sẽ tiếp tục nghiên cứu và dự định sẽ giới thiệu ở bài báo tiếp theo.

TÀI LIỆU THAM KHẢO

[1] Amadio R.M and Cardelli R., Subtyping Recursive Types, *ACM Trans. Program. Lang. Systems* **15** (4) (1993) 575-631.

[2] Donald K., *Practical Application of Object-Oriented Techniques to Relational Databases*, John Wiley and Sons, 1994.

[3] Đoàn Văn Ban, *Phân tích, thiết kế và lập trình hướng đối tượng*, NXB Thống kê, Hà Nội, 1997.

[4] Đoàn Văn Ban, Cấu trúc dữ liệu trong các hệ cơ sở dữ liệu đối tượng, báo cáo tại *Xemina "Một số vấn đề chọn lọc trong công nghệ thông tin"*, ĐH Bách Khoa Hà Nội, 1999.

[5] Formica A. et al., Object-Oriented Database Schema Analysis and Inheritance Processing: A Graph-theoretic Approach, *Data Knowledge Eng. Journal* **24** (2) (1997) 157-181.

[6] Kim W., Object - oriented databases: Definition and research directions, *IEEE Trans. Knowledge Eng.* **2** (3) (1990).

[7] Missikoff M. and Tolati M., MOSAIKO - A system for conceptual modeling and rapid prototyping of object - oriented database applications, *Proceeding of the ACM SIGMOD Confer. Minneapolits, MN*, May 24-27, 1994, 508-519.

[8] Mueck A. T. and Polaschek M. L., *Index Data Structures on Object - Oriented Databases*, Kluwer Academic Publishers, 1997.

Nhận bài ngày 20 - 4 - 1999

Viện Công nghệ thông tin