

## VỀ MỘT PHƯƠNG PHÁP KẾT HỢP NGÔN NGỮ DỮ LIỆU VỚI NGÔN NGỮ LẬP TRÌNH

NGUYỄN VĂN TÂNG  
Trường Đại học Bách khoa Hà Nội

### 1. ĐẶT VẤN ĐỀ

Ngôn ngữ dữ liệu (NNDL) là công cụ giao tiếp giữa người sử dụng kho dữ liệu thông qua hệ quản trị dữ liệu với những chức năng chính: tạo lập, kiểm sửa và tìm kiếm. Ở chế độ chương trình, mỗi yêu cầu của người sử dụng được thể hiện bằng một chương trình viết trên NNDL. Nếu ta kết hợp (nhúng) NNDL vào một ngôn ngữ lập trình (NNLT) cấp cao nào đó thì ngôn ngữ nhúng kết hợp được khả năng của cả hai ngôn ngữ ban đầu. Khi cần xử lý một cách phức tạp trên những thông tin rút ra từ một cơ sở dữ liệu (như giải các bài toán tối ưu), người sử dụng có thể viết đoạn chương trình bằng NNLT thể hiện tính toán đó xen vào chương trình NNDL. Ngược lại, một chương trình viết trên NNLT đến đoạn cần thao tác các phép toán trên tệp thì đoạn đó được diễn đạt qua NNDL. Tóm lại, với một hệ nhúng yêu cầu của người sử dụng là một chương trình có thể viết lẫn cả hai ngôn ngữ gọi là chương trình hỗn hợp (CTHH). CTHH thông thường được xử lý hoặc bằng phương pháp tiền dịch hoặc bằng phương pháp gọi thủ tục. Bài này sẽ giới thiệu một phương pháp - tạm gọi là phương pháp tách - dựa trên cơ sở của phương pháp gọi thủ tục nhưng phân phân tích nhận biết của tất cả câu lệnh viết bằng NNDL được tiến hành trước trong giai đoạn dịch.

### 2. PHƯƠNG PHÁP TÁCH

Trong phương pháp gọi thủ tục, mỗi câu lệnh NNDL có thể biểu diễn qua một lệnh gọi thủ tục (lồng nhau) dạng loạt các thao tác cơ sở (TTCS). Mỗi TTCS bao gồm: phần PHÂN TÍCH với chức năng nhận biết, tạo các bảng giá trị tham biến, chỉ dẫn đề sẵn sàng cung cấp cho phần sau và phần thực hiện thao tác (phần HÀNH ĐỘNG). Trong phương pháp tách, phần PHÂN TÍCH của tất cả các TTCS đều nằm trong NNDL-Processor, còn phần HÀNH ĐỘNG (là các đơn thể đích) hợp thành thư viện NNDL-LIB. Chức năng của NNDL-Processor bao gồm:

- Đọc vào từng câu lệnh NNDL.
- Phân tích toàn bộ phần cú pháp và một phần ngữ nghĩa với các thông báo kèm theo.
- Điền vào các bảng giá trị tham biến, chỉ dẫn cho những đơn thể HÀNH ĐỘNG liên quan.
- Tạo lập đơn thể khung tương ứng bằng cách: liên kết các câu lệnh gọi các đơn thể HÀNH ĐỘNG liên quan, tạo qui trình đề sẵn sàng cung cấp tham biến, chỉ dẫn cho các đơn thể này từ các bảng, các mục lục lập sẵn.

Với phương pháp tách, CTHH được viết tách thành hai chương trình riêng biệt: chương trình thứ nhất (CT1) viết trên NNLT và chương trình thứ hai (CT2) trên NNDL. Ta có thể nhận được CT1 từ CTHH bằng cách chỉ giữ lại các câu lệnh viết trên NNLT còn các câu lệnh viết trên NNDL được thay thế bằng các câu lệnh liên hệ với CT2 qua kiểu gọi thủ tục ngoài. Bản thân các câu lệnh viết trên NNDL được xếp riêng thành CT2.

CT1 và CT2 được dịch độc lập (có thể tiến hành song song) trên 2 bộ dịch riêng biệt: CT1 qua bộ dịch ngôn ngữ chủ (NNLT-Compiler) còn CT2 qua NNDL-Processor. Cả 2 bộ dịch đều cho ra sản phẩm là các đơn thể đích. Qua bộ liên kết của hệ điều hành (LKED) ta thu được một đơn thể nạp với chức năng đúng như nội dung của CTHH nguồn ban đầu.

Người sử dụng có thể diễn đạt yêu cầu hay bài toán của mình chỉ bằng CT1 hay CT2. Trong các trường hợp riêng đó, nếu chỉ có CT1 thì hệ nhúng trở về làm việc thông thường như lập trình trên NNLT tương ứng. Trường hợp chỉ có chương trình CT2 thì hệ làm việc như một hệ thao tác dữ liệu không nhúng.

#### *Một số nhận xét về phương pháp tách:*

Nguyên lý làm việc độc lập nhau của NNLT-Compiler và NNDL-Processor với các sản phẩm là đơn thể dạng đích cho ta một số ưu điểm của phương pháp này. Khi dịch chương trình, nhất là đối với các chương trình lớn mà chi phí thời gian cho giai đoạn dịch là đáng kể, nếu một trong hai chương trình CT1 và CT2 có sai sót gì thì chỉ cần dịch lại đơn vị chương trình có sai sót đó. Đặc biệt nếu phần nào đã có sẵn đơn thể dạng đích thì chỉ còn giải quyết phần chưa có. Ở giai đoạn dịch, NNDL-Processor kiểm tra được toàn bộ cú pháp và một phần ngữ nghĩa trên lược đồ cấu trúc với toàn bộ các câu lệnh của CT2. Qua lược đồ cấu trúc, bộ xử lý có thể tiến hành tối ưu quá trình thực hiện một câu lệnh NNDL ở dạng loạt TTCS lồng nhau. Cả khi yêu cầu của người sử dụng là CTHH thay chỉ là CT2, NNDL-Processor đều chuyển thẳng chương trình dạng nguồn sang các đơn thể dạng đích (dịch một lần).

Tuy nhiên với phương pháp tách, do các đơn thể chương trình ngoài rời nhau dẫn đến hạn chế một phần sự giao lưu giữa NNLT và NNDL. Chẳng hạn giữa CT1 và CT2 không sử dụng được loại biến toàn cục.

#### *Mối giao lưu giữa NNLT và NNDL trong phương pháp tách:*

Về chương trình, mối giao lưu trong phương pháp này thể hiện qua lời gọi thủ tục lẫn nhau. Đối với CT1, bản thân NNLT cấp cao chọn đề nhúng đã có sẵn các câu lệnh gọi thủ tục ngoài. Khi xây dựng NNDL cho CT2 phải cài đặt khả năng hướng tới các thủ tục riêng của người sử dụng viết bằng NNLT.

Về dữ liệu, các thủ tục này có thể trao đổi với nhau qua tham biến, qua khả năng cùng truy nhập đến các tập dữ liệu (các kho dữ liệu ở bộ nhớ thứ cấp), qua việc cùng sử dụng loại biến ngoài hoặc cùng sử dụng vùng dữ liệu chung trong bộ nhớ trong (như COMMON trong FORTRAN) nếu NNLT được chọn đề nhúng cho phép. Để cùng truy nhập đến những tập dữ liệu chung, NNDL phải có khả năng thao tác các phép toán tệp trên cả những tệp có sẵn do CT1 cung cấp hoặc các tệp từ ngoài đưa vào. Các tệp này thường khá đa dạng về cấu trúc các bản ghi, như thuộc tính có thể loại đơn hay loại trường, giá trị của chúng có thể là xâu (kí tự hay bit), có thể là số với nhiều khuôn dạng thường thấy. Việc « trang bị » cho NNDL khả năng tạo lập và nhận biết cấu trúc các tệp đa dạng như trên trong một hệ nhúng là cần thiết.

### 3. MỘT ỨNG DỤNG PHƯƠNG PHÁP TÁCH

Tư tưởng của phương pháp tách đã được vận dụng trong quá trình thiết kế và cài đặt hệ nhúng BK trên máy tính EC-1022. Trong hệ này, NNLT được chọn đề nhúng là PL/1 và NNDL được xây dựng dựa trên mô hình quan hệ có mở rộng. BK xử lí trực tiếp các tệp có sẵn với cấu trúc bản ghi tương đối đa dạng như:

a) Thuộc tính có thể là loại đơn hay là trường.

b) Kiểu của chúng có thể là một trong 9 kiểu: xâu kí tự, xâu bit, số nhị phân tính nửa từ, số nhị phân tính cả từ, số thập phân tính, số nhị phân động một từ, số thập phân động một từ, số nhị phân động kép (hai từ) và số thập phân động kép.

Để áp dụng được các phép toán quan hệ cho những quan hệ không chuẩn, trong các TTCS có thêm phép tính chu trình (dạng do-step-until) có thể kèm phép kết thúc chu trình (end do). Trong các phép toán, ở những vị trí xét đến giá trị của một thuộc tính được mở rộng ra xét hàm một thuộc tính. Các hàm này là hàm ngoài của người sử dụng đưa vào. Thuộc tính có thể kèm theo biểu thức chỉ số.

Hệ BK được áp dụng trong việc quản lý học tập và giảng dạy trong ngành đại học. chi tiết về hệ này sẽ được giới thiệu trong một bài báo khác.

Nhận ngày 5-3-1985

#### TÀI LIỆU THAM KHẢO

1. Eswaran, Blasgen, Storage and access methode in relational databases IBM syst. J.16, 4 (1977).
2. Ullman J. Principles of Data Base Systems Prentice Hall, 1980.
3. Nguyễn Xuân Huy, Về việc thiết kế hệ quản lý cơ sở dữ liệu. Một số khía cạnh lập trình ứng dụng, Hà Nội 1984.
4. Van Gils T. The philosophy of an Extensible Language, Int. Computing Symposium 1972.

#### ABSTRACT

**On a method combining the data languager and the Proegamming language.**

A programming language and the data base language will be combined one to another in such a way, that an application source program, which a mixture of statements on host and data base languages instead, can be now separabed into two source modules, the one on the host language and the other on the data base. By the host standard compiler and data base processor they are independenly compiled into object modules. The LINKEP EDITOR of the operating system links those modules into a load module that corresponds to the original mixed progam.