

PHÁT TRIỂN KỸ THUẬT MAONT BẰNG JSP ĐẢM BẢO GẮN BÓ DỮ LIỆU TRONG CÁC HỆ THỐNG THÔNG TIN ĐÀO TẠO

THE DEVELOPMENT OF TECHNICAL MAONT BY JSP DATA COHERENCE IN EDUCATIONAL INFORMATION SYSTEMS

PHẠM ĐÌNH HÂN

Trường Cao đẳng Công nghệ, Đại học Đà Nẵng

LÊ VĂN SƠN

Trường Đại học Sư phạm, Đại học Đà Nẵng

TÓM TẮT

Bài báo nghiên cứu phát triển hệ thống thông tin đào tạo với mục đích đảm bảo gắn bó thông tin trong các cơ sở dữ liệu được truy cập từ xa với số lượng lớn qua mạng viễn thông.

Giải pháp được nghiên cứu phát triển kỹ thuật MAONT (Model Advanced of Open Nested Transaction) trên cơ sở lý thuyết danh sách di chuyển. Việc thực nghiệm được tiến hành trên mạng Intranet bằng JSP (Java Server Page) với nhiều Server theo hướng đa truy cập vào CSDL đào tạo dùng chung với các phép đọc, đọc và ghi đồng thời.

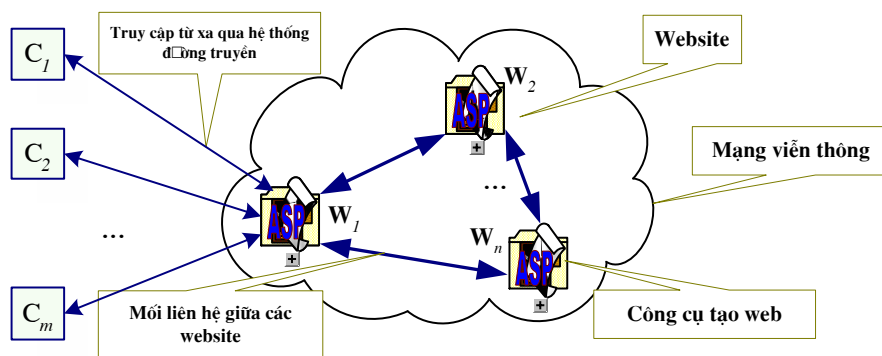
ABSTRACT

The research paper studies and develops the educational information systems on the basis of the technical solution that allows assuring data coherence in data base being accessed from a distance in large amount through the telecommunication network.

The study focuses on the development of MAONT (Model Advanced of Open Nested Transaction) on the basis of the theory of mobile list. Experimentation is carried out on the Intranet by JSP (Java Server Page) with several servers through the tendency of multi-access in training database commonly used with methods of “read, read and write” synchronically.

1. ĐẶT VẤN ĐỀ

Phần lớn các hệ thống thông tin đào tạo hiện nay đều được xây dựng trên cơ sở nhiều Server mắc nối với nhau thông qua đường truyền. Trên mỗi Server được cài đặt các chương trình quản lý và các cơ sở dữ liệu của hệ thống, có thể được mô tả như hình 1.



Hình 1. Mô hình tổng quát của hệ thống thông tin đào tạo

Hệ thống thông tin này cho phép truy cập đồng thời bởi nhiều người sử dụng (NSD), trong đó các thao tác cập nhật bao gồm cả việc đọc, đọc và ghi dữ liệu đan xen nhau và mang

tính ngẫu nhiên. Với cấu trúc như vậy, hệ thống thông tin đào tạo là hệ phức tạp và có những đặc điểm:

1. *Cập nhật thường xuyên* để làm tươi thông tin trong các CSDL đào tạo.
2. Cho phép *truy cập từ xa* các yêu cầu có liên quan đến công tác đào tạo.
3. Thực hiện *đăng ký từ xa* trong thi, kiểm tra và ghi danh.

Hệ thống với những đặc điểm trên đòi hỏi phải đảm bảo tính gắn bó dữ liệu bằng những giải pháp kỹ thuật đáng tin cậy.

Xuất phát từ tình hình thực tiễn nêu trên, kết hợp với thành quả nghiên cứu mô hình MAONT đã được trình bày trong tài liệu tại tạp chí Khoa học & Công nghệ các trường Đại học Kỹ thuật Việt Nam (Tr 27, số 46+47/2004), chúng tôi tiếp tục nghiên cứu và phát triển mô hình MAONT bằng JSP trong môi trường phân tán nhằm đảm bảo tính gắn bó dữ liệu cho các hệ thống thông tin đào tạo khi có các truy cập đồng thời của người sử dụng (NSD).

2. CƠ SỞ KỸ THUẬT CỦA MAONT

MAONT được cải tiến cho phép các giao dịch tiến hành đồng thời do một giao dịch có thể chứa một giao dịch con khác theo phương pháp tiếp cận khoá chốt, nhằm vào việc cập nhật, đăng ký từ xa trong hệ thống thông tin đào tạo.

Với cách tiếp cận đó thì việc quản lý các khoá và việc thực thi các giao dịch được thực hiện theo qui tắc:

1. Mỗi giao dịch con thực thi như một giao dịch và khi hoàn tất sẽ chuyển khoá của nó cho giao dịch cha.
2. Một giao dịch cha kế thừa các khoá và các cập nhật của những giao dịch con đã uỷ thác của nó.
3. Trạng thái kế thừa chỉ xảy ra khi các giao dịch cha kế thừa là thấy được giao dịch con đã uỷ thác. Tuy nhiên muốn truy xuất trạng thái này, một giao dịch cha kế thừa phải nhận được các khoá thích hợp. Tương tranh khoá được xác định giống như trong giao dịch phẳng, ngoại trừ sẽ bỏ qua các khoá kế thừa còn được giữ lại bởi một tổ tiên của một giao dịch con đang yêu cầu.
4. Nếu một giao dịch con bị hủy bỏ thì tất cả các khoá và cập nhật mà giao dịch con đó và các giao dịch cha kế thừa của nó có đều bỏ đi. Cha của giao dịch con bị hủy bỏ không nhất thiết phải hủy bỏ.

Với các giao dịch lồng mở thì các khoá được một giao dịch con giữ sẽ được giải phóng ngay khi nó uỷ thác hoặc hủy bỏ và điều này được phản ánh ngay trong các bảng khoá.

Theo định nghĩa [1,3] thì *Giao dịch là tập hợp của hàng loạt các sự kiện mà tất cả đều phải được hoàn tất đến cùng hoặc không có bất cứ sự kiện nào được thực hiện*. Và thông thường, khi triển khai thiết kế trên các công cụ tạo web động, người ta có thể phân ra 2 loại giao dịch:

- *Giao dịch CSDL* (database transactions) được thiết kế ngay trong giai đoạn xây dựng hệ quản trị các CSDL và thủ tục này có chức năng điều khiển giao dịch.
- *Giao dịch cơ sở kết nối* (connection-based transactions) cho phép xác định từ bên ngoài CSDL, thường là chức năng của một đối tượng nào đó, ví dụ như đối tượng *connection*.

Trong phạm vi bài viết này, giao dịch cơ sở kết nối được nghiên cứu với các đối tượng chủ yếu là Connection, SqlConnection và OdbcConnection để tiến hành cập nhật thông tin.

Bảng 1 phản ánh các phương thức quan trọng trong quá trình thiết kế các hệ thống cập nhật CSDL bằng giao dịch với các thao tác tương ứng là *bắt đầu, kết thúc, uỷ thác* và *khôi phục*.

Bảng 1. Các phương thức chuẩn trên giao dịch

TT	Tên gọi	Thuyết minh
1	BeginTransaction <name>	Bắt đầu một giao dịch có tên là <i>name</i> trên một liên kết
2	End	Ranh giới kết thúc giao dịch
3	Commit	Khẳng định tất cả các thay đổi kể từ thời điểm bắt đầu giao dịch
4	Rollback	Khôi phục lại toàn bộ các thay đổi của giao dịch kể từ khi nó bắt đầu thực hiện

Những khía cạnh liên quan đến công cụ, DBMS, ODBC, soạn thảo, xử lý và thực thi các website không được trình bày trong phạm vi báo cáo này. Ta có thể tìm thấy những vấn đề này trong các tài liệu, đặc biệt là trong [1,3,4,5,7].

3. GIẢI PHÁP PHÁT TRIỂN BẰNG JSP

Một trong những công nghệ chính được sử dụng trong đề tài nghiên cứu là Java Servlet. Một Servlet là một chương trình phía Server phục vụ các yêu cầu HTTP và trả về kết quả là một hồi đáp HTTP. Khi người dùng gửi một yêu cầu HTTP cho một Servlet, Server chỉ đơn giản sử dụng một tiến trình để xử lý yêu cầu, do đó khi có nhiều yêu cầu sẽ không tạo ra nhiều tiến trình như các chương trình CGI (Common Gateway Interface) thường làm.

Java Server Pages (JSP), tương tự như Active Server Pages (ASP) chứa HTML, mã Java, các thành phần Java Bean. JSP cung cấp cách nhúng các thành phần vào một File để phát sinh nội dung web. Khi người dùng yêu cầu một trang JSP, đầu tiên, Web Server phát sinh Servlet tương ứng, nếu servlet chưa tồn tại. Tiếp theo, server thực thi servlet và trả về nội dung kết quả cho trình duyệt web.

Để phát triển các servlet, chúng ta cần phải có Java Servlet Development Kit (JSDK), công cụ này chứa các giao diện và các lớp cần thiết để phát triển và chạy servlet. Các lớp và các giao diện này được chứa trong hai gói javax.servlet và javax.servlet.http. Gói đầu tiên cung cấp các giao diện cơ bản, gói thứ hai cung cấp các lớp được dẫn xuất từ *giao diện servlet* tổng quát và cung cấp các công cụ đặc biệt để phục vụ các yêu cầu HTTP.

Bảng 2. Các phương thức chính của giao diện Servlet

Ký hiệu	Init()	Service()	Destroy()	Get ServletConfig()	Get ServletInfo()
Thuyết minh	Tạo thể hiện của Servlet	Hồi đáp các yêu cầu của HTTP	Hủy thể hiện của Servlet	Truy cập tham số khởi tạo	Trả về một đối tượng String...

Mạng là linh hồn của Java. Tất cả các lớp cơ bản cho việc giao tiếp và làm việc với mạng đều nằm trong gói Java.net. Các lớp trong gói Java.net bao gồm hai loại: một là socket API (Application Program Interface), một là làm việc với các URL (Unified Resource Locator).

Socket API của Java cung cấp các công cụ để truy cập đến các giao thức mạng chuẩn được sử dụng để giao tiếp giữa các Host trên mạng Internet. Socket API là công cụ mức thấp nhất, chúng ta có thể sử dụng socket cho bất kỳ kiểu giao tiếp Client / Server nào và các ứng dụng ngang hàng trên mạng nhưng chúng ta phải cài đặt các giao thức ứng dụng mức thấp để điều khiển và thông dịch dữ liệu.

Socket là giao diện lập trình mức thấp cho các giao tiếp hướng mạng. Chúng gửi các dòng dữ liệu giữa các ứng dụng có thể hoặc không nằm trên cùng một host. Gói java.net hỗ trợ một giao diện hướng đối tượng, đơn giản hóa socket nên việc giao tiếp mạng trở nên dễ dàng hơn. Java cung cấp các socket để hỗ trợ cho 3 lớp phân biệt của các giao thức bên dưới: Socket, DatagramSocket, MulticastSocket. Trong phạm vi bài báo này chúng ta chỉ nghiên cứu về lớp Socket.

Lớp Socket sử dụng giao thức hướng kết nối TCP/IP, đảm bảo dữ liệu không bị mất trong quá trình truyền.

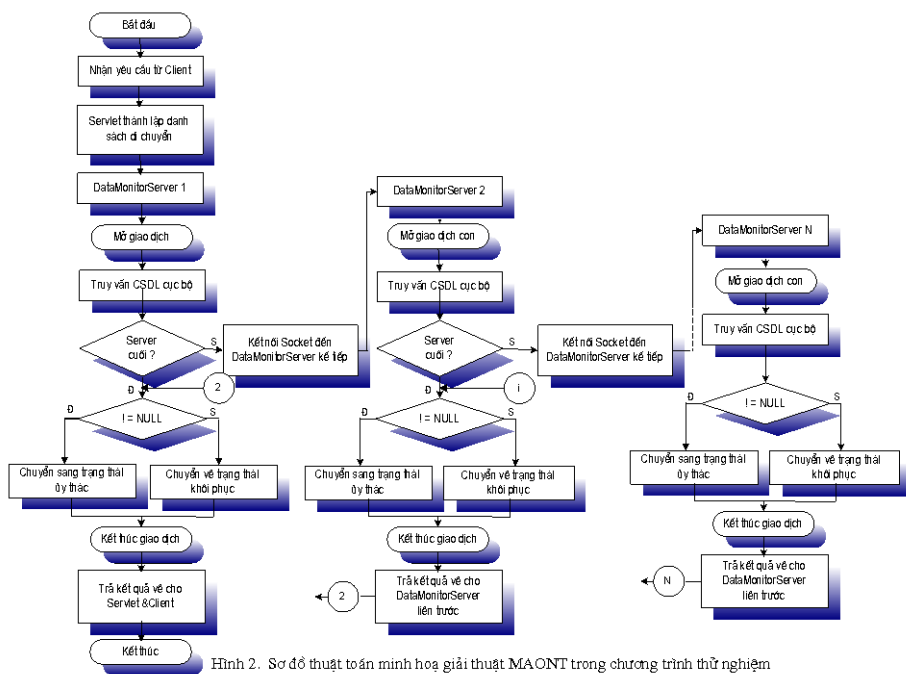
Các phương thức của Socket được sử dụng trong bài báo thể hiện ở bảng 3.

Bảng 3. Các phương thức chính của Socket

Ký hiệu	Constructor	WriteObject	ReadObject	Close
Thuyết minh	Tạo Socket kết nối từ xa	Ghi vào Socket một obj cài đặt giao diện Serializable	Đọc một đối tượng từ Socket	Đóng Socket

Lớp ServerSocket được sử dụng để thu nhận các kết nối từ client của server. ServerSocket chỉ được tạo một lần và nhận các kết nối đến từ các client. Phương thức accept của ServerSocket sẽ đợi cho đến khi có kết nối đến và trả về một đối tượng Socket.

Để triển khai các giải thuật của mô hình giao dịch lỏng mở phát triển MAONT trên công cụ tạo web động JSP vào hệ thống thông tin đào tạo nhằm duy trì đặc tính gắn bó dữ liệu, ta thực hiện hàng loạt các thao tác. Các thao tác này có thể được mô tả bằng thuật toán minh họa trong hình 2.

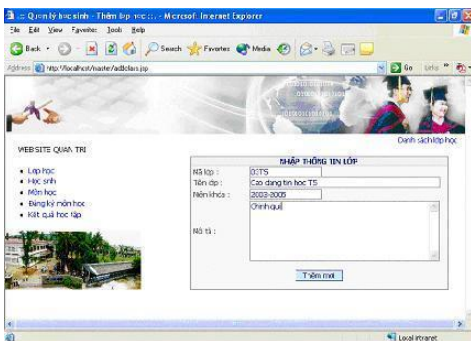


Hình 2. Sơ đồ thuật toán minh họa giải thuật MAONT trong chương trình thử nghiệm

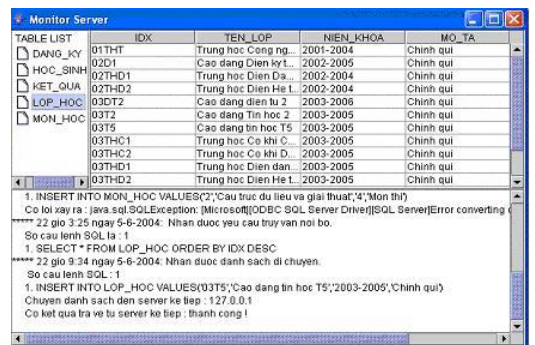
Các bước thực hiện thuật toán gồm:

1. Khởi động các DatamonitorServer trong hệ thống, cho phép NSD truy cập Web trên WebServer nằm tại Server1 để tiến hành cập nhật thông tin.
2. Servlet xử lý yêu cầu của NSD, tùy theo yêu cầu mà Servlet thành lập danh sách di chuyển.
3. Chuyển danh sách di chuyển đến DatamonitorServer đầu tiên trong danh sách di chuyển.
4. DatamonitorServer nhận được danh sách di chuyển, mở ra một giao dịch (transaction) thực thi các câu lệnh SQL trên CSDL cục bộ của mình. Sau khi kết thúc truy vấn, DatamonitorServer gửi danh sách di chuyển đến DatamonitorServer kế tiếp trong danh sách di chuyển.
5. Nếu DatamonitorServer là Server cuối cùng trong danh sách di chuyển - nếu việc thực thi các câu lệnh truy vấn trên CSDL cục bộ thành công thì chuyển sang trạng thái uỷ thác (CommitTransaction) và trả danh sách kết quả khác null về cho DatamonitorServer liền trước nó. Ngược lại trả về null và chuyển sang trạng thái khôi phục (RollbackTransaction).
6. DatamonitorServer nhận được danh sách kết quả từ Server sau nó trong danh sách di chuyển - nếu kết quả khác null thì chuyển sang trạng thái uỷ thác (CommitTransaction) và gán kết quả truy vấn cục bộ vào danh sách kết quả và trả về Server liền trước nó. Ngược lại, chuyển sang trạng thái khôi phục (Rollback transaction) trả kết quả về null cho Server liền trước.
7. Khi DatamonitorServer đầu tiên nhận được danh sách kết quả - nếu danh sách kết quả khác null thì chuyển sang trạng thái uỷ thác (CommitTransaction) và trả danh sách kết quả cho Servlet. Ngược lại, chuyển sang trạng thái khôi phục (Rollback transaction) và trả kết quả về null cho Servlet.
8. Servlet nhận được danh sách kết quả và xử lý dựa trên danh sách kết quả.

Sau đây là một số hình ảnh của chương trình thử nghiệm trong hệ thống tin đào tạo:



Hình 3. Cửa sổ cập nhật thông tin lớp học



Hình 4. Cửa sổ MonitorServer

4. KẾT LUẬN

Dựa vào cơ sở lý thuyết của hệ tin học phân tán và nguyên lý các hệ cơ sở dữ liệu phân tán, việc xây dựng các hệ thống thông tin đào tạo thể hiện bằng các Website trên nền tảng gắn bó dữ liệu đã đạt được những kết quả về lý thuyết và thực nghiệm.

Kết quả lý thuyết thể hiện trong việc hình thành phương pháp cập nhật dữ liệu trong hệ thống thông tin dùng chung qua mạng viễn thông. *Các giải pháp kỹ thuật* giải quyết vấn đề gắn bó trên cơ sở nghiên cứu các đặc tính thay đổi dữ liệu trong quá trình cập nhật trong điều kiện có nhiều truy cập đồng thời trên mạng Intranet, từ đó ta có thể phát triển và áp dụng cho mạng Internet.

Kết quả cụ thể của bài báo:

1. Giải quyết được các yêu cầu đảm bảo tính gắn bó và tính không trùng trong quá trình cập nhật và đăng ký thông tin từ xa theo kiểu nhiều người sử dụng (NSD).
2. Sử dụng mô hình giao dịch lỏng mở phát triển MAONT để xây dựng giải pháp kỹ thuật đảm bảo gắn bó dữ liệu cho các hệ thống thông tin đào tạo.

Hướng phát triển của bài báo:

1. Phát triển các khả năng cung cấp các dịch vụ từ môi trường trung gian, đảm bảo tính toàn vẹn dữ liệu.
2. Hoàn thiện các chức năng kiểm soát lỗi trên hệ thống.
3. Phát triển các ứng dụng phân tán dựa trên mô hình MAONT của hệ thống đã xây dựng.

TÀI LIỆU THAM KHẢO

- [1] Lê Văn Sơn, *Hệ tin học phân tán*, NXB Đại học Quốc gia TP Hồ Chí Minh, 2002.
- [2] Lê Văn Sơn, “Nghiên cứu phát triển giải thuật quản lý nhiều bản sao cho các ứng dụng phức tạp”, *Tạp chí Khoa học và Công nghệ*, Đại học Đà Nẵng, số 1.2003, tr. 50-55, 2003.
- [3] M. Tamer Ozsu, Patrick Valduriez, Trần Đức Quang biên dịch, *Nguyên lý các hệ cơ sở dữ liệu phân tán*, Tập I và II, NXB Thống kê, Hà Nội, 1999.
- [4] Phạm Hữu Khang, *Lập trình ứng dụng chuyên nghiệp SQL Server 2000*, Tập I và II, NXB Lao động - Xã hội, Hà Nội, 2002.
- [5] Nguyễn Phương Lan, Hoàng Đức Hải, *Java lập trình mạng*, NXB Giáo dục.
- [6] Website về gắn bó / tra cứu trên Internet, 2001.
- [7] Phạm Đình Hân, Lê Văn Sơn, “Phát triển giải pháp kỹ thuật đảm bảo gắn bó dữ liệu cho các hệ thống thông tin đào tạo trên mạng Internet/Intranet”, *Tạp chí Khoa học và Công nghệ* các Trường Đại học Kỹ thuật, số 46+47/2004, Hà Nội, 2004.