

GIAO THỨC CPM VÀ MỘT ĐỀ XUẤT ỨNG DỤNG TRONG VIỄN THÔNG

LÊ HỮU LẬP, NGUYỄN KHẮC LỊCH

1. MỞ ĐẦU

Giao thức nổi tiếng nhất 2PC do Gray đề xuất [1] và được đồng đảo sử dụng cho cơ sở dữ liệu phân tán (DDBMS) [2 - 4]. Nhưng giao thức này lại bộc lộ nhiều khó khăn và nhược điểm khi áp dụng đối với môi trường phân tán di động [5]. Trong đó đáng chú ý là việc hủy bỏ giao tác (hủy bỏ các ứng dụng) một khi người sử dụng ngắt kết nối thiết bị di động tạm thời để xử lý ngoại tuyến [6]. Để giải quyết vấn đề hủy bỏ giao tác này tác giả đề xuất một giao thức hợp thức cải tiến "The Commit Protocol for Mobile Distributed Transaction Management, CPM". Giao thức này hỗ trợ việc thực hiện ngoại tuyến, di chuyển, tính toán không kết nối và giảm được chi phí truyền thông vì bản chất của giao thức này không cần pha chuẩn bị như 2PC. Phần tiếp theo của bài báo tập trung vào việc giới thiệu đề xuất của tác giả về giao thức CPM và các thuật toán có liên quan; Tiếp đến, một ứng dụng tiêu biểu CPM trong Viễn thông được trình bày. Cuối cùng tác giả đưa ra những kết luận và hướng ứng dụng của giao thức này.

2. GIAO THỨC CPM VÀ CÁC THUẬT TOÁN TƯƠNG ỨNG

2.1. Giao thức CPM

Với CPM, các thành phần tương tác trong giai đoạn thực hiện và kết thúc gồm: - yêu cầu việc thực hiện của một giao tác (Application, APP); - các thành viên thực hiện các thao tác của giao tác (Participant, P); - đại diện cho các thành viên trong việc thực hiện, kết thúc và phục hồi của giao thức CPM (Participant Agent, PA); - bộ điều phối điều khiển sự hợp thức của giao tác (Coordinator, C). Ở đây C luôn nằm trên mạng cố định hoặc mạng vô tuyến chất lượng cao (vì lý do an toàn), còn các thành phần khác thì tùy ý (có thể trên mạng cố định hoặc di động). Đây là giao thức điều khiển việc kết thúc của giao tác nên trong phạm vi của giao thức CPM, chúng tôi chỉ mô tả các tiến trình hợp thức, còn các hoạt động khác như lập lịch trình của giao tác [7 - 9] hay phân bổ các giao tác con thực hiện tại các trạm được giả định là đã sẵn có.

Nội dung của giao thức CPM được mô tả như sau đây.

Gọi $T = \{T_1, T_2, \dots, T_n\}$ là tập các giao tác, $P = \{P^1, P^2, \dots, P^m\}$ là tập các thành viên sẽ thực hiện tập giao tác T , $S = \{S^1, S^2, \dots, S^m\}$ là tập các trạm tương ứng với các thành viên P .

Với giao tác T_i bất kỳ nào đó $T_i \in T$. Gọi $J = \{1, 2, \dots, a\} \subset \{1, 2, \dots, m\}$, tập chỉ số ứng với các thành viên cùng thực hiện giao tác T_i .

Bộ lập lịch trình sẽ chia giao tác T_i thành các giao tác nhánh được thực hiện tại các thành viên tương ứng: $T_i = \{T_{ij} / j \in J\}$, với j bất kỳ thì $T_{ij} = \{op\}$, $op \in \{Read, Write\}$.

Giả sử tại trạm S^k nào đó, $S^k \in S$ có ứng dụng APP^k yêu cầu hệ thống thực hiện giao tác T_i và tương ứng có P^k, PA^k .

Sau đây là các bước thực hiện của giao thức:

b1. APP^k gửi chuỗi các thao tác của T_i cần thực hiện đến PA^k .

b2. PA^k gửi T_i đến bộ lập lịch trình.

b3. PA^k lấy từ bộ lập lịch trình các giao tác nhánh và gửi đến các thành viên tương ứng $P^j, j \in J$, trước khi gửi đi PA^k ghi nhật kí từng thao tác của các giao tác nhánh thuộc T_i vào bộ nhớ (RAM).

b4. PA^k khởi động timeout và chờ các xác nhận từ các P^j .

b5. Các $P^j, j \in J$ sẽ thực hiện cục bộ từng T_{ij} tương ứng. Sau khi thực hiện xong từng thao tác sẽ gửi xác nhận của thao tác tương ứng về cho PA^k .

b6. Nếu hết thời gian và PA^k không nhận được toàn bộ xác nhận ACK^j của T_i từ các thành viên thì sẽ gửi lệnh huỷ bỏ giao tác đến C và C quảng bá lệnh huỷ bỏ đến các thành viên.

b7. Ngược lại nếu PA^k nhận được toàn bộ xác nhận của T_i từ các thành viên thì sẽ gửi lệnh hợp thức của giao tác đó đến C. Đến thời điểm này thì các tính chất ACI của giao tác đã được đảm bảo cục bộ bởi các P^j cho tất cả các giao tác nhánh của giao tác T_i . Tuy nhiên, các thành viên sẽ không thấy được sự kết thúc của giao tác nên chúng không đảm bảo tính bền vững D của giao tác. Tính chất D sẽ được đảm bảo bởi C bằng cách PA^k sẽ đẩy nhật kí của T_i đến C và C sẽ ghi ra bộ lưu trữ ổn định.

b8. Sau khi hoàn thành các bước trên thì các tính chất ACID của giao tác được bảo đảm cho tất cả các nhánh của giao tác T_i . Vì vậy, C quyết định hợp thức và ghi quyết định đó vào bộ lưu trữ ổn định, sau đó quảng bá quyết định hợp thức đến tất cả các thành viên $PA^j (j \in J)$ và đợi nhận các xác nhận, PA^j nhận được quyết định hợp thức sẽ đưa ra thêm nhiệm vụ ghi lại và thực hiện lệnh hợp thức đó cho P^j . Sau khi thực hiện xong P^j sẽ chuyển xác nhận đến PA^j và PA^j chuyển tiếp đến C. Khi C nhận được toàn bộ xác nhận từ các P^j thì sẽ hoàn tất giao tác và thông báo ok đến PA^k , sau đó PA^k sẽ gửi ok đến APP^k .

2.2. Các thuật toán trong CPM

Các khái niệm mô tả trong thuật toán này đã được trình bày trong phần mô tả CPM

Thuật toán tại APP

Begin

1. APP^k : Gửi chuỗi các thao tác của giao tác T_i đến PA^k
2. APP^k đợi đến khi nhận được lệnh ok hoặc not ok

End

Thuật toán tại PA

Begin

IF(PA^k nhận được giao tác T_i)

Then

Begin


```

        PAj kiểm tra log tại Pj
        IF (tồn tại bản ghi "commit")
        Then
            PAj gửi xác nhận về cho C
        Else
            PAj yêu cầu Pj thực hiện lại giao tác nhánh
            Ty nhờ nhật ký của C
        End
    End
Else
    PAj đưa lệnh abort cho Pj
End
End
End
*****
Thuật toán tại P (thành viên)
Begin
    Pi, j ∈ J thực hiện các thao tác và gửi xác nhận về cho PAk
    Pj đợi nhận lệnh commit từ PAj tương ứng
    IF (Pj nhận được lệnh ghi bản ghi "commit" và lệnh commit) Then
        Begin
            Pj tiến hành hợp thức giao tác nhánh Ty
            Pj ghi bản ghi "commit" vào nhật ký cục bộ LocalLog
        End
    Else
        Pj hủy bỏ giao tác nhánh Ty
    End
End
*****
Thuật toán C (Bộ điều phối)
Begin
    IF (C nhận được quyết định abort từ đại diện thành viên) Then
        Begin
            C quảng bá lệnh abort đến các Pj, ∀ j ∈ J
        End
    Else
        IF (C nhận được nhật ký Log và lệnh commit từ đại diện thành viên)
        Then
            Begin
                C ghi nhật ký vào bộ nhớ ổn định (stable storage)
                C quảng bá lệnh commit đến các Pj, ∀ j ∈ J
                C đợi để nhận xác nhận từ các đại diện thành viên
                IF (C nhận được hết các xác nhận của các giao tác nhánh Ty)
                Then

```

Begin

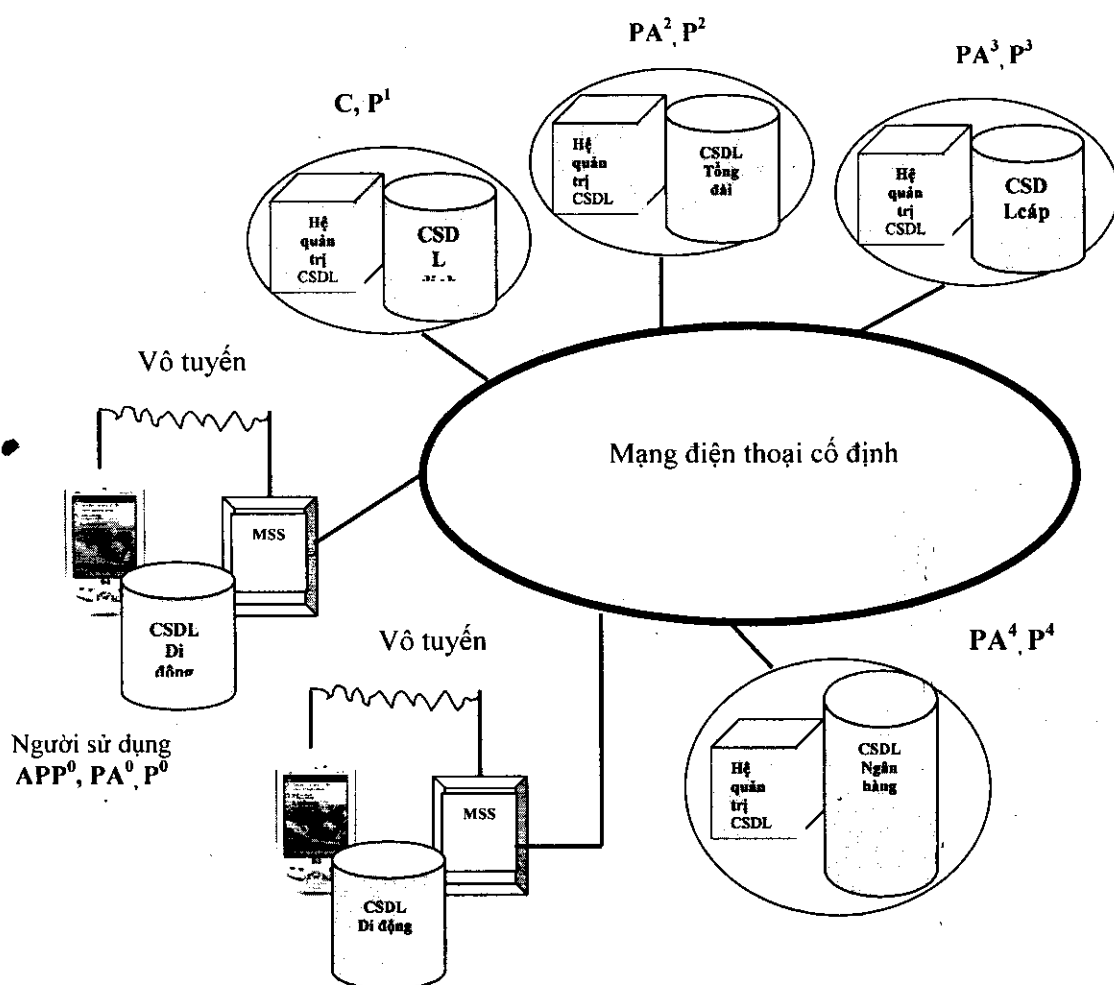
*C hoàn tất giao tác và gửi thông báo ok về
cho ứng dụng tương ứng*

End

End

End

3. ỨNG DỤNG CPM TRONG VIỄN THÔNG



Hình 1. Kiến trúc cơ sở dữ liệu di động phục vụ cho yêu cầu lắp đặt điện thoại

Phân tích việc áp dụng CPM vào bài toán yêu cầu lắp đặt điện thoại [5]. Giả thiết tất cả mọi khách hàng sử dụng tiền trong tài khoản cho việc thanh toán để minh họa đầy đủ cho giao thức CPM. Quá trình đặt dịch vụ được mô tả như sau:

Khách hàng sử dụng loại điện thoại di động thông minh (vừa có chức năng điện thoại di động vừa làm máy vi tính dạng Pocket PC) để kết nối vô tuyến đến máy chủ của nhà cung cấp dịch vụ Viễn thông và tải xuống các mẫu khai, bản đồ nơi khách hàng định đăng ký dịch vụ và các thông tin về dịch vụ. Sau đó khách hàng sẽ tạm ngắt ra khỏi mạng để tiến hành xử lý ngoại tuyến (xác định vị trí trên bản đồ, điền vào mẫu khai, lựa chọn các dịch vụ kèm theo như báo thức, bắt giữ cuộc gọi đến, gọi hội nghị, ...). Khi xử lý xong khách hàng sẽ kết nối lại vào mạng, máy chủ cung cấp dịch vụ sẽ căn cứ vào các thông tin của khách hàng để kiểm tra các điều kiện cho lắp đặt dịch vụ. Để kiểm tra các điều kiện máy chủ đặt dịch vụ sẽ kiểm tra bên máy chủ quản lý mạng cấp xem tại khu vực đó còn cáp không, tiếp tục máy chủ sẽ kiểm tra bên máy chủ quản lý tổng đài xem có còn số điện thoại không,..., sau đó kiểm tra bên máy chủ ngân hàng xem khách hàng có đủ tiền không. Nếu tất cả các điều kiện thỏa mãn máy chủ đặt dịch vụ sẽ thông báo cho khách hàng để khẳng định và hoàn tất việc đặt dịch vụ, ngược lại máy chủ sẽ thông báo các điều kiện còn thiếu và đẩy thông tin khách hàng vào danh sách chờ (waiting list).

Hình 1 mô tả kiến trúc tiêu biểu của việc đặt dịch vụ gồm: - Cơ sở dữ liệu (CSDL) đặt dịch vụ, - CSDL quản lý tổng đài, - CSDL quản lý cáp điện thoại, - CSDL Ngân hàng về tài khoản người sử dụng, - Người sử dụng (gồm những người sử dụng thiết bị di động như máy tính xách tay, PDA, Mobile phone, ... chứa các CSDL di động), - Mạng cố định và mạng vô tuyến.

Trong kiến trúc trên, các thành viên của giao thức CPM được bố trí tại CSDL tương ứng của: APP⁰, PA⁰ của di động NSD (thành viên P⁰), C của dịch vụ (thành viên P¹), PA², P² của tổng đài, PA³, P³ của quản lý cáp, PA⁴, P⁴ của ngân hàng. Tại CSDL dịch vụ có các quan hệ dịch_vu(ma_dv, ten_dv, mo_ta), khách_hang_yeu_cau(ma_kh,ten_kh,dia_chi_kh, tai_khoan, ma_kc), ban_do(ma_vung, ban_do). Tại CSDL tổng đài có các quan hệ so_dien_thoai_trong(ma_tong_dai, so_dt_trong). Tại CSDL quản lý cáp có các quan hệ ket_cuoi(ma_kc, dia_chi_kc),doi_so(ma_kc, doi_so), cap(ma_kc_vao, ma_kc_ra, ma_cap, dung_luong). Tại csdl Ngân hàng có các quan hệ tai_khoan(ma_tai_khoan, ma_khach_hang, so_tien).

Giao tác mà APP⁰ đưa ra cho PA⁰ như sau:

Begin

```
SELECT ma_dv, ten_dv, mo_ta
FROM dich_vu;

SELECT ma_vung, ban_do
FROM ban_do

WHERE ma_vung = gia_tri_kh_chon_ma_vung;

SELECT ma_kc INTO val_ma_kc
FROM ket_cuoi

WHERE dia_chi_kc = gia_tri_kh_chon_ma_kc;

SELECT dung_luong INTO var_dung_luong_1_kc
FROM cap

WHERE ma_kc_ra = val_ma_kc;

SELECT count(*) INTO var_cnt_dung_luong_1_kc
```

```

FROM doi_so
WHERE ma_kc = val_ma_kc;
IF (var_dung_luong_1_kc - var_cnt_dung_luong_1_kc) > 0
Then
    Message('Con cap');
Else
    Message('Khong con cap');
EndIf
SELECT count(*) INTO ts_so_dt
FROM so_dien_thoai_trong;
IF ts_so_dt > 0
Then
    Message('Con so');
Else
    Message('Khong con so');
EndIf
● SELECT so_tien INTO so_tien_tk
FROM tai_khoan
WHERE ma_tai_khoan = tai_khoan_kh
IF so_tien_tk > so_tien_dv_kh
Then
/* Chuyen_khoan(so_tien_tk-so_tien_dv_kh) */
    UPDATE tai_khoan SET so_tien = so_tien-so_tien_dv_kh
    WHERE ma_tai_khoan = ma_tai_khoan_kh;
    UPDATE tai_khoan SET so_tien = so_tien + so_tien_dv_kh
    WHERE ma_tai_khoan = ma_tai_khoan_nha_cc
Else
    Message('Khach hang khong co du tien de dat dich vu')
EndIf
INSERT INTO khach_hang_yeu_cau
VALUES(max(ma_khach_hang)+1,ten_kh,dia_chi_kh,tai_khoan,ma_kc)
End

```

Ở đây những giá trị như: `tai_khoan_kh`, `gia_tri_kh_chon_ma_vung`, `gia_tri_kh_chon_ma_kc` là những giá trị mà khách hàng chọn và điền vào form của chương trình còn `so_tien_dv_kh` ứng dụng sẽ tính ra khi khách hàng chọn dịch vụ. Vậy, có thể khái quát giao tác trên như sau:

$$T_0 = \{R(dich_vu), R(ban_do), W(khach_hang_yeu_cau), R(so_dien_thoai_trong), R(ket_cuoi), R(cap), R(ket_cuoi), R(tai_khoan), W(tai_khoan), W(tai_khoan)\}$$

Khi PA^0 nhận được T_0 sẽ gửi T_0 đến bộ lập lịch trình đặt tại C và nhận được 4 giao tác con hoạt động tại 4 trạm như sau:

$$T_0 = \{T_{01}, T_{02}, T_{03}, T_{04}\},$$

$$T_{01} = \{R(dich_vu), R(ban_do), W(khach_hang_yeu_cau)\},$$

$$T_{02} = \{R(so_dien_thoai_trong)\},$$

$$T_{03} = \{R(ket_cuoi), R(cap), R(doi_so)\},$$

$$T_{04} = \{R(tai_khoan), W(tai_khoan), W(tai_khoan)\}.$$

Lần lượt PA^0 gửi các thao tác của các giao tác con $T_{01}, T_{02}, T_{03}, T_{04}$, đến các thành viên tương ứng P^1, P^2, P^3, P^4 . Trong quá trình thực hiện giao tác T_{01} sau khi người sử dụng thực hiện 2 thao tác là `R(dich_vu)`, `R(ban_do)` thì người sử dụng ngắt kết nối ra khỏi mạng và thực hiện xử lý ngoại tuyến (off line) như khai báo tên, địa chỉ, tài khoản, vị trí sẽ lắp đặt điện thoại (điều này sẽ không thực hiện được với giao thức 2PC đã được cài đặt trong các hệ quản trị cơ sở dữ liệu thương mại như Oracle, SQL Server, DB2, ... Vì nếu ngắt ra khỏi mạng khi kết nối trở lại giao tác sẽ bị huỷ bỏ), ... Sau đó, PA^0 sẽ ghi nhật ký từng thao tác và gửi tiếp các thao tác của các giao tác con đến các thành viên tương ứng để thực hiện. Trạng thái thực hiện mỗi thao tác sẽ trả lại cho PA^0 (điều này có thể thấy khi thao tác với môi trường CSDL phân tán thông thường). Khi PA^0 nhận được hết các xác nhận của các thao tác trong giao tác T_0 , đến thời điểm này tính chất ACI của giao tác đã được đảm bảo. Để đảm bảo nốt tính chất D của giao tác thì PA^0 sẽ gửi nhật ký và lệnh hợp thức đến C (máy chủ cung cấp dịch vụ) để C ghi nhật ký vào ổ cứng máy chủ và hoàn tất giao tác.

Trên đây chúng tôi vừa mô tả về một ứng dụng trong Viễn thông và thấy giao thức CPM có thể áp dụng tốt trong việc xử lý ngoại tuyến, di chuyển và tạm ngắt thành viên ra khỏi mạng.

4. KẾT LUẬN

Trong bài báo này tác giả đã trình bày đề xuất liên quan đến giao thức CPM dưới dạng các thuật toán. Tác giả đã phân tích và chỉ ra sự phù hợp với nhiệm vụ quản lý giao tác phân tán di động thông qua việc phân tích áp dụng vào một bài toán tiêu biểu của viễn thông và chỉ ra rằng CPM là một giao thức quản lý sự kết thúc các giao tác di động thoả mãn các yếu tố sau:

- Hỗ trợ các giao tác ngoại tuyến và giảm thiểu việc huỷ bỏ giao tác tại thời điểm kết nối lại;

- Có thể hợp thức một giao tác khi thành viên đang di chuyển;
- Hỗ trợ việc hợp thức các giao tác trong thời điểm bị tạm ngắt (disconnect);
- Áp dụng được cho bất kì loại server nào, sử dụng không lãng phí tài nguyên của các máy chủ di động gọn nhẹ.

TÀI LIỆU THAM KHẢO

1. Bernstein, P.A., Hadzilacos, V., Goodman, N.- Concurrency control and recovery in database systems, book, Addison Wesley Publishers, 1987.
2. Gray, J. - Notes on DataBase Operating Systems, Lecture Notes in Computer Science, Vol. 60, Springer Verlag, 1978.
3. IBM Corp - DB2 Everywhere for Windows CE and Palm OS Software, Administration and Application Programming Guide, SC26967500, 1999.
4. Nguyễn Khắc Lịch - Quản lí giao tác và điều khiển tương tranh trong hệ cơ sở dữ liệu phân tán, Luận án Thạc sĩ, Đại học quốc gia Hà Nội, 1999.
5. Nguyễn Khắc Lịch - Thiết kế và xây dựng ngân hàng dữ liệu Bưu điện, Đề tài KHCN mã số: 210-00-TCT-AP-VT-62, giai đoạn 2000-2001.
6. Nguyen Khac Lich - Realtime Database Model (in Vietnamese), Results of The Science & Technology Researching of RIPT, 2000, pp. 382-390.
7. Nguyen Khac Lich - System Aided Management for Post and Telecom (SAMPT), Paper of The 25th AIC Conference, May 2001, pp. 343-349.
8. Pitoura, E., Samaras, G. - Data Management for Mobile Computing', book, Kluwer Academic Publishers, 1998.
9. Walborn, G., Chrysanthis, P. K. - Using the Escrow Transactional Method to Manage Replicated Data in Disconnected Mobile Operations', CS Technical Report 9432, University of Pittsburgh, 1994.
10. Walborn, G., Chrysanthis, P.K. - PROMOTION: Support for Mobile Database Access, Personal Technologies Journal, September 1997.
11. Lê Hữu Lập, Nguyễn Khắc Lịch - Giao thức hợp thức cho quản lí giao tác phân tán di động, Tạp chí Tin học và Điều khiển học **19** (4) (2003) 391-398.
12. <http://www.pocketpc.com>
13. <http://www.palm.com>

SUMMARY

THE CPM AND A PROPOSAL FOR TELECOMMUNICATION APPLICATION

With the rapid growth and convenience of wireless technology, nowadays, it opens a new era in mobile computing and a new orientation in communication and processing data. Mobile databases are hosted on the lightweight devices like palmtops, PDA, Pocket PC, Cellular phones