



(12) **BẢN MÔ TẢ GIẢI PHÁP HỮU ÍCH THUỘC BẰNG ĐỘC QUYỀN
GIẢI PHÁP HỮU ÍCH**

(19) **Cộng hòa xã hội chủ nghĩa Việt Nam (VN)
CỤC SỞ HỮU TRÍ TUỆ**

(11)



2-0002433

(51) **G06F 16/27**
2020.01

(13) **Y**

(21) 2-2020-00273

(22) 25/06/2018

(67) 1-2018-02739

(45) 26/10/2020 391

(43) 25/09/2018 366A

(73) **TẬP ĐOÀN CÔNG NGHIỆP - VIỄN THÔNG QUÂN ĐỘI (VIETTEL) (VN)**
Số 1 Trần Hữu Dực, Mỹ Đình 2, quận Nam Từ Liêm, thành phố Hà Nội

(72) Lương Thị Chuyên (VN); Vũ Đức Chính (VN); Đoàn Khả Cường (VN); Phạm Văn Thông (VN); Nguyễn Hoàng Quân (VN); Lê Mạnh Cường (VN).

(74) Công ty Luật TNHH quốc tế BMVN (BMVN INTERNATIONAL LLC)

(54) **PHƯƠNG PHÁP ĐIỀU KHIỂN PHÂN TRANH TRONG HỆ THỐNG QUẢN LÝ DỮ
LIỆU PHÂN TÁN ĐA VI XỬ LÝ THÔNG QUA QUÁ TRÌNH TÍNH TOÁN NHẢN THỜI
GIAN LÔGIC**

(57) Giải pháp hữu ích đề cập đến phương pháp điều khiển phân tranh trong hệ thống quản lý dữ liệu phân tán đa vi xử lý khi thực hiện nhiều giao dịch đồng thời thông qua quá trình tính toán khoảng nhãn thời gian lôgic.

Lĩnh vực kỹ thuật được đề cập

Giải pháp hữu ích đề cập đến phương pháp điều khiển phân tranh phân tán trong hệ thống quản lý dữ liệu phân tán đa vi xử lý nhằm đảm bảo tính khả tuần tự - mức độ cao nhất của thuộc tính độc lập, trong quá trình xử lý các giao dịch (transaction) một cách đồng thời.

Tình trạng kỹ thuật của giải pháp hữu ích

Trong hệ thống quản lý dữ liệu phân tán đa vi xử lý, dữ liệu được phân chia, sao chép và lưu trữ tại các nhân thuộc các máy chủ khác nhau. Một giao dịch bao gồm nhiều tác vụ được thực hiện tuần tự thông qua pha đọc, pha chuẩn bị (prepare phase) và pha bàn giao (commit phase). Các tác vụ được thao tác với các dữ liệu ở các nhân (core) khác nhau của các máy chủ (server). Tuy nhiên, quá trình xử lý các giao dịch là bất đồng bộ, máy chủ sẽ thực thi các pha bất kì của các giao dịch bất kì. Do đó, quá trình xử lý đồng thời nhiều giao dịch được xem xét bởi quá trình chia sẻ nguồn tài nguyên và chia sẻ khoảng thời gian xử lý tại các nhân trong các máy chủ để đạt hiệu suất thực hiện cao và thời gian đáp ứng trung bình của giao dịch ngắn với độ trễ thấp. Tuy nhiên, khi thực hiện nhiều giao dịch đồng thời, thuộc tính độc lập có thể bị xâm phạm dẫn đến không đảm bảo được tính nhất quán của dữ liệu. Khái niệm khả tuần tự được đưa ra để đảm bảo tính nhất quán của dữ liệu ở mức cao nhất và phương thức điều khiển phân tranh được đề xuất để đảm bảo tính khả tuần tự trong trường hợp thực hiện nhiều giao dịch đồng thời.

Có nhiều phương pháp được đề xuất để thực hiện điều khiển phân tranh trong hệ thống phân tán, chẳng hạn như phương pháp khóa hai pha trung tâm, phương pháp khóa hai pha phân tán, phương pháp sắp xếp thứ tự dựa vào nhãn thời gian, phương pháp nhãn thời gian đa phiên bản, phương pháp điều khiển tương tranh lạc quan, phương pháp MaaT, phương pháp Sundial. Phương pháp dùng bộ phận trung tâm, chẳng hạn như phương pháp khóa hai pha trung tâm, phương pháp Sundial hoặc các phương pháp nhãn thời gian có các vấn đề về thất nút cổ chai khi số lượng giao dịch thực hiện lớn hoặc độ tin cậy không cao khi bộ phận quản lý bị hỏng. Phương pháp khóa hai pha phân tán có thể tránh được trường hợp thất nút cổ chai hoặc độ tin cậy cao nhưng lại gặp phải vấn đề khóa chết do quá trình phân chia và sao chép dữ liệu đến từng máy chủ. Đối với các phương pháp liên quan đến nhãn thời gian, chẳng hạn như phương pháp sắp xếp thứ tự dựa vào nhãn thời gian, phương pháp nhãn thời gian đa phiên bản, nếu sử dụng đồng hồ vật lý thì quá trình đồng bộ đồng hồ phải được thực hiện hoặc tồn tại trung tâm quản lý thời gian. Đối với phương pháp điều khiển tương tranh lạc quan hoặc

phương pháp MaaT, yêu cầu hệ thống phải có khả năng lưu trữ cao vì phải lưu trữ các tập đọc và tập ghi của tất cả các giao dịch. Khi có nhiều xung đột xảy ra dẫn đến quá trình tái thực hiện hoặc hủy bỏ giao dịch sẽ xảy ra nhiều, gây lãng phí nguồn tài nguyên đã sử dụng.

Bản chất kỹ thuật của giải pháp hữu ích

Do đó, để khắc phục các nhược điểm nêu trên, các tác giả giải pháp hữu ích đã nghiên cứu và đề xuất phương pháp điều khiển phân tranh đảm bảo tính khả tuần tự khi thực hiện nhiều giao dịch đồng thời trong hệ thống quản lý dữ liệu phân tán đa vi xử lý. Phương pháp theo giải pháp hữu ích thực hiện tính toán nhãn thời gian phù hợp cho từng giao dịch dựa vào nhãn thời gian của các dữ liệu ở các tác vụ trong các nhân của các máy chủ khác nhau. Đối với từng máy chủ, các dữ liệu ở các nhân khác nhau được tính toán một khoảng nhãn thời gian thực hiện hợp lý, từ đó khoảng giao của các dữ liệu ở các tác vụ của một giao dịch trong một máy chủ được quyết định làm khoảng nhãn thời gian thực hiện giao dịch tại máy chủ đó. Sau đó, đối với từng giao dịch, máy khách thực hiện tính toán khoảng giao của các máy chủ và quyết định giá trị nhãn thời gian thấp nhất là nhãn thời gian cho quá trình thực hiện giao dịch tương ứng. Tính khả tuần tự được đảm bảo thông qua nhãn thời gian thực hiện giao dịch trong hệ phân tán đa vi xử lý. Thêm vào đó, tại mỗi nhân trong máy chủ, phương pháp này sử dụng một khóa chốt cứng cho một dữ liệu ghi và một khóa chốt mềm cho dữ liệu đọc để thực hiện các giao dịch từ giai đoạn chuẩn bị thành công đến khi hoàn thành giao dịch (bàn giao (commit) hoặc hủy bỏ (abort)). Mỗi dữ liệu yêu cầu thao tác chỉ cần duy trì một giao dịch ghi, một danh sách các giao dịch đọc và cơ chế hàng đợi được sử dụng để tránh lãng phí nguồn tài nguyên.

Phương pháp điều khiển phân tranh trong hệ thống quản lý dữ liệu phân tán đa vi xử lý khi thực hiện nhiều giao dịch đồng thời thông qua quá trình tính toán nhãn thời gian lôgic. Một giao dịch được thực hiện tuần tự thông qua pha đọc, pha chuẩn bị và pha bàn giao. Để thực hiện nhiều giao dịch đồng thời, hệ thống quản lý dữ liệu phân tán đa vi xử lý thực hiện các pha của các giao dịch khác nhau một cách bất đồng bộ. Quá trình thực hiện một giao dịch được thực hiện thông qua các bước sau:

- bước 1: thực hiện pha đọc

máy khách thực hiện gửi tác vụ đọc dữ liệu đến các máy chủ, các máy chủ thực hiện điều hướng các tác vụ đến nhân chứa dữ liệu để thực lấy dữ liệu trong bộ lưu trữ trả về cho máy khách.

- bước 2: thực hiện pha chuẩn bị

máy khách sau khi thực hiện nhận hết dữ liệu đọc, máy khách thực hiện gửi tác vụ chuẩn bị của các dữ liệu đọc và ghi của giao dịch đến các máy chủ chứa dữ liệu tương ứng. Máy chủ thực hiện điều hướng đến các nhân chứa dữ liệu tương ứng để thực hiện thẩm định quá trình chuẩn bị và tính toán khoảng nhãn thời gian bàn giao của từng giao dịch. Sau đó máy chủ

thực hiện phản hồi lại cho máy khách kết quả thực hiện pha bàn giao và khoảng nhân thời gian bàn giao.

- bước 3: thực hiện pha bàn giao

sau khi nhận được bản tin phản hồi của pha bàn giao, máy khách thực hiện quyết định kết quả bàn giao thông qua kết quả nhận được và tính toán nhân thời gian bàn giao dựa vào khoảng nhân thời gian nhận được. Tiếp theo, máy khách thực hiện gửi bản tin bàn giao đến các máy chủ thực hiện giao dịch cùng với nhân thời gian tìm được để thực hiện bàn giao hoặc hủy bỏ giao dịch. Máy chủ thực hiện bản tin bàn giao đến các nhân thực hiện pha chuẩn bị trước đó để thực hiện bàn giao hoặc hủy bỏ giao dịch.

Giải pháp hữu ích đề xuất phương pháp điều khiển phân tranh trong hệ thống quản lý dữ liệu phân tán đa vi xử lý khi thực hiện nhiều giao dịch đồng thời thông qua quá trình tính toán nhân thời gian logic, bao gồm:

- thực hiện điều khiển phân tranh tại các nhân của các máy chủ; máy khách thực hiện điều hướng các tác vụ đến các máy chủ tương ứng và các máy chủ nhận được yêu cầu thực hiện tác vụ sẽ thực hiện điều hướng đến nhân chứa dữ liệu để xử lý;
- thực hiện khóa chốt cứng cho dữ liệu ghi và khóa chốt mềm cho dữ liệu đọc từ lúc dữ liệu được thực hiện pha chuẩn bị thành công đến lúc hoàn thành giao dịch trong đó:
 - + tại một thời điểm, có thể có một hoặc nhiều giao dịch đọc một dữ liệu nhưng chỉ có nhiều nhất một giao dịch thực hiện ghi một dữ liệu;
 - + khi một dữ liệu ghi bị khóa chốt cứng, bất kỳ một tác vụ đọc nào của giao dịch khác chứa dữ liệu sẽ được đưa vào hàng đợi giúp cho giao dịch ghi có thể được thực hiện trong một thời gian trễ trọn vòng; khi một dữ liệu bị khóa chốt cứng hoặc chốt mềm, bất kỳ một tác vụ ghi chứa dữ liệu nào của giao dịch khác thực hiện tác vụ sẽ được thực hiện theo cơ chế đợi-hủy bỏ (wait-die) dựa vào định danh của các giao dịch để tránh khóa chết trong các nhân của các máy chủ; cơ chế wait-die được thực hiện xem xét nếu giao dịch đang xem xét có định danh nhỏ hơn giao dịch đang thực hiện khóa dữ liệu thì thực hiện đưa giao dịch vào trong một hàng đợi ưu tiên, ngược lại thì máy chủ sẽ trả về mã lỗi cho máy khách để yêu cầu thực hiện lại giao dịch;
- thực hiện điều khiển phân tranh và sắp xếp trình tự giữa các giao dịch thông qua quá trình tính toán khoảng nhân thời gian logic tại từng nhân của máy chủ và tại từng máy chủ trong khi máy khách sẽ thực hiện quá trình quyết định nhân thời gian hợp lý để thực hiện giao dịch hoặc hủy bỏ giao dịch;
- quá trình tính toán khoảng nhân thời gian tại các nhân của máy chủ được thực hiện trong pha chuẩn bị bao gồm các bước sau:
 - + đối với dữ liệu đọc của giao dịch, thực hiện tính toán cận dưới bằng giá trị nhân thời gian ghi tại thời điểm dữ liệu đã được đọc để thực hiện sắp xếp giao dịch đang được xem xét sau các giao dịch đã thực hiện thành công trước đó; nếu dữ liệu bị

khóa ghi bởi một giao dịch khác thì thực hiện tính toán cận trên bằng giá trị nhỏ nhất của cận trên hiện tại với giá trị liền trước giá trị cận dưới của giao dịch khóa ghi dữ liệu đó để thực hiện sắp xếp giao dịch đọc đứng trước giao dịch ghi đã được chuẩn bị thành công; nếu có các giao dịch khác thực hiện ghi dữ liệu thành công từ lúc giao dịch hiện tại thực hiện pha đọc đến lúc giao dịch hiện tại thực hiện pha chuẩn bị thì thực hiện cập nhật giá trị cận trên bằng giá trị nhỏ nhất của giá trị cận trên hiện tại và giá trị liền trước của nhãn thời gian bàn giao của giao dịch thành công cuối cùng trong các giao dịch thành công đó để thực hiện sắp xếp giao dịch hiện tại liền trước các giao dịch thực hiện ghi dữ liệu thành công sau khi dữ liệu được đọc;

- + đối với dữ liệu ghi của giao dịch, thực hiện cập nhật cận dưới của giao dịch bằng giá trị nhỏ nhất của cận dưới hiện tại, giá trị liền sau giá trị cận trên của tất cả các giao dịch đã thực hiện pha chuẩn bị đọc dữ liệu thành công và giá trị liền sau giá trị cận trên của tất cả các giao dịch đã hoàn thành đọc dữ liệu nhằm mục đích sắp xếp giao dịch ghi đứng sau các giao dịch đọc đã thực hiện pha chuẩn bị thành công hoặc hoàn thành giao dịch đọc thành công;
- quá trình tính toán khoảng nhãn thời gian tại các máy chủ được thực hiện trong pha chuẩn bị như sau:
 - + sau khi tất cả các dữ liệu của một giao dịch thực hiện pha chuẩn bị thành công, máy chủ thực hiện tìm khoảng giao của giao dịch ở tất cả các nhân và gửi khoảng giao này về cho máy khách; nếu máy chủ không tìm được khoảng giao nào phù hợp cho giao dịch thì máy chủ thực hiện gửi về cho máy khách một mã lỗi;
- quá trình tính toán nhãn thời gian duy nhất cho một giao dịch để hoàn thành giao dịch bao gồm:
 - + đối với từng dữ liệu, sau khi đa số các máy chủ thực hiện dữ liệu đó trả về giá trị khoảng thời gian bàn giao, máy khách thực hiện tìm khoảng giao của đa số các máy chủ chứa dữ liệu làm khoảng giao của dữ liệu đó;
 - + sau đó, máy khách thực hiện tìm khoảng giao của tất cả các dữ liệu; nếu bất kì dữ liệu nào mà máy khách không tìm được khoảng giao thì máy khách thực hiện hủy bỏ giao dịch đến tất cả các máy chủ thực hiện và chứa đựng dữ liệu của giao dịch đó thông qua pha bàn giao; ngược lại, nếu máy khách tìm được khoảng giao của tất cả các dữ liệu, máy khách sẽ quyết định lấy cận dưới của khoảng giao làm nhãn thời gian bàn giao cho giao dịch và thực hiện hoàn thành giao dịch cùng với nhãn thời gian tìm được thông qua pha bàn giao.

Mô tả vắn tắt các hình vẽ

Hình 1 là hình vẽ thể hiện quá trình thực hiện pha đọc của giao dịch A trong hệ thống quản lý dữ liệu có sử dụng phương pháp điều khiển phân tranh theo giải pháp hữu ích.

Hình 2 là hình vẽ thể hiện quá trình thực hiện pha chuẩn bị của giao dịch A trong hệ thống quản lý dữ liệu có sử dụng phương pháp điều khiển phân tranh theo giải pháp hữu ích.

Hình 3 là hình vẽ thể hiện quá trình thực hiện pha bàn giao của giao dịch A trong hệ thống quản lý dữ liệu có sử dụng phương pháp điều khiển phân tranh theo giải pháp hữu ích.

Mô tả chi tiết giải pháp hữu ích

Quá trình xử lý giao dịch trong hệ thống quản lý dữ liệu phân tán đa vi xử lý áp dụng phương pháp điều khiển phân tranh theo giải pháp hữu ích được thực hiện thông qua các pha đọc, pha chuẩn bị và pha bàn giao như được mô tả dưới đây. Khi một giao dịch bắt đầu ở phía máy khách, máy khách này sẽ thực hiện phân tích các tác vụ và gửi đến máy chủ tương ứng, sau đó tác vụ được điều hướng xử lý đến từng nhân chứa dữ liệu tương ứng. Tại mỗi nhân của máy chủ, mỗi giao dịch T_i luôn bao gồm một định danh $txnID$, một khoảng nhãn thời gian bàn giao từ $lowerTS$ đến $upperTS$, một trong bốn trạng thái: được khởi tạo (Initialized), được chuẩn bị (Prepared), được bàn giao (Committed), được hủy bỏ (Aborted) và danh sách các thông tin dữ liệu của các tác vụ ($opMap$). Mỗi thông tin dữ liệu ($keyInfo$) của tác vụ bao gồm dữ liệu (key), một cờ $isReadOp$, một cờ $isWriteOp$, một nhãn thời gian ghi ($write_{ts}$). Mỗi dữ liệu đều được lưu trữ trong bộ nhớ kèm theo nhãn thời gian đọc (rts) và nhãn thời gian ghi (wts). Tại một thời điểm có thể có nhiều giao dịch cùng đọc một dữ liệu. Dữ liệu sẽ được chuẩn bị hoặc đã được chuẩn bị (prepare) nhưng chưa bàn giao (commit) hoặc hủy bỏ (abort) sẽ được lưu trữ trong bộ nhớ đệm ($cache$) với các thông số một tập các giao dịch đọc đã được chuẩn bị thành công (RS), một giao dịch đọc (T_r) thuộc RS với giá trị cận trên lớn nhất và chỉ có nhiều nhất một giao dịch ghi (T_w) được chuẩn bị thành công nhưng chưa thực hiện bàn giao, một nhãn thời gian đọc ($readTS$), một nhãn thời gian ghi ($writeTS$) và một hàng đợi ưu tiên ($pendingTxn$).

a. Pha đọc:

Máy khách thực hiện yêu cầu tác vụ đọc dữ liệu x của giao dịch T_i đến nhóm sao chép. Máy chủ s thuộc nhóm sao chép sau khi nhận yêu cầu đọc dữ liệu x sẽ thực hiện định tuyến đến nhân chứa dữ liệu này. Tại nhân chứa dữ liệu, khối giao dịch sẽ thực hiện kiểm tra trong bộ đệm ($cache$) xem có giao dịch nào đang trong pha chuẩn bị dữ liệu đó hay không.

- Nếu tồn tại $cache.x.T_w$, chứng tỏ đang có một giao dịch đang thực hiện chuẩn bị dữ liệu thành công. Để dữ liệu nhận về là mới nhất, khối giao dịch thực hiện đưa T_i vào trong danh sách $cache.x.pendingTxn$.

- Nếu không tồn tại $cache.x.T_w$, thực hiện truy vấn dữ liệu x trong bộ nhớ và sau đó trả về giá trị của dữ liệu và nhãn thời gian $x.wts$ cho máy khách đồng thời cập nhật thông tin dữ liệu vào trong $cache$.

Quá trình đọc dữ liệu của giao dịch được mô tả chi tiết trong giải thuật 1 (algorithm1) như sau:

Giải thuật 1: Pha đọc

Dữ liệu: Giao dịch T_i với định danh $txnID$, dữ liệu đọc x , bộ đệm cache, bộ lưu trữ str

```

1       $info = cache.x$ 
2       $if(info)$ 
4           $if(info.T_w)$ 
5               $if(x.txnID < info.writeTS)$ 
6                   $x.pendingList.add(T_i)$ 
                   $return$ 
7               $else$ 
8                   $return ERRORCODE\_TXN\_WAIT\_DIE$ 
9               $end$ 
10          $end$ 
11      $else$ 
12          $initialize\ cache.x\ with\ readTS = str.x.rts, writeTS = str.x.wts$ 
13      $end$ 
14          $cache.x.writeTS = str.x.wts$ 
15      $initialize\ keyInfo(x)\ with\ isReadOp = true, isWriteOp = false,$ 
         $write_{ts} = str.x.wts$ 
16      $update\ T_i\ with\ T_i.lowerTS = str.x.wts, T_i.upperTS = +\infty, state =$ 
         $Initialized, add\ keyInfo\ to\ opMap$ 
17      $return\ str.x.value\ and\ str.x.wts$ 

```

b. Pha chuẩn bị:

- Sau khi tất cả các tác vụ đọc của giao dịch T_i được trả về, máy khách gửi yêu cầu chuẩn bị đến tất cả các nhóm máy chủ thực hiện tác vụ đọc các dữ liệu trước đó và nhóm máy chủ thực hiện tác vụ ghi. Tại từng nhân chứa dữ liệu ghi y của giao dịch, thực hiện kiểm tra xem có tồn tại giao dịch đã khóa bởi giao dịch khác (tồn tại $cache.y.T_w$ hoặc $cache.y.T_r$ với $cache.y.T_r.upperTS = +\infty$) hay không. Nếu có sử dụng phương pháp wait-die để tránh khóa chết. Nếu giao dịch T_i có $txnID$ nhỏ hơn $txnID$ của giao dịch đang thực hiện khóa dữ liệu thì thực hiện thêm giao dịch T_i tới hàng đợi ưu tiên. Ngược lại sẽ thực hiện trả về mã lỗi TRANSACTION_WAIT_DIE cho máy khách để yêu cầu thực hiện lại giao dịch. Nếu không có bất kì dữ liệu nào của giao dịch T_i bị khóa bởi giao dịch khác thì thực hiện thăm tra xung đột và tính toán khoảng nhãn thời gian bàn giao như sau:
- Đối với các tác vụ đọc dữ liệu x mà không thuộc tác vụ ghi, thực hiện chuẩn bị:
 - Thực hiện tính toán $lowerTS = \max(lowerTS, keyInfo(x).write_{ts})$
 - Thực hiện kiểm tra xem có giao dịch nào đang khóa ghi x hay không (đã chuẩn bị nhưng chưa bàn giao) thông qua việc kiểm tra sự tồn tại của $x.T_w$ thì thực hiện cập nhật $upperTS = \min(upperTS, cache.x.T_w.lowerTS - 1)$.
 - Thực hiện kiểm tra xem $keyInfo(x).write_{ts} < cache.x.writeTS$ thì chứng tỏ có giao dịch khác đã biến đổi dữ liệu x sau khi thực hiện tác vụ đọc, thực hiện cập nhật $upperTS = \min(upperTS, cache.x.writeTS - 1)$.
- Đối với các tác vụ ghi y , thực hiện chuẩn bị:
 - Nếu dữ liệu key y chưa tồn tại trong bộ đệm $cache$, thực hiện cập nhật trong bộ đệm $readTS = storage.y.rts, writeTS = storage.y.wts$.
 - Thực hiện cập nhật $lowerTS = \max(lowerTS, y.readTS + 1)$
 - Thực hiện cập nhật $lowerTS = \max(lowerTS, y.T_r.upperTS + 1)$
 - Thực hiện kiểm tra tính hợp lệ của khoảng bàn giao. Nếu $upperTS < lowerTS$ thì thực hiện trả về mã lỗi TRANSACTION_CONFLICT.
- Thực hiện khóa các dữ liệu ghi của giao dịch.

Đối với các tác vụ đọc mà ko thuộc tác vụ ghi thì thêm giao dịch đọc vào trong RS và cập nhật giao dịch T_r với $upperTS$ cao nhất. Thực hiện khóa các dữ liệu của tác vụ đọc nếu cận trên là $+\infty$.

Thực hiện chuẩn bị tác vụ ghi của giao dịch xuống vùng nhớ đệm. Sau đó trả về kết quả chuẩn bị và khoảng nhãn thời gian bàn giao từ $lowerTS$ đến $upperTS$ tại từng nhân về cho máy chủ. Máy chủ thực hiện tìm kiếm khoảng giao giữa các khoảng nhãn thời gian bàn giao. Nếu máy chủ không tìm được khoảng giao nhãn thời gian bàn giao phù hợp, máy chủ

sẽ trả về cho máy khách mã lỗi TRANSACTION_CONFLICT, ngược lại sẽ trả về khoảng nhãn thời gian bàn giao tìm được cho máy khách. Máy khách thực hiện tìm giao giữa các khoảng nhãn thời gian bàn giao của các máy chủ và quyết định xem hoàn thành hay hủy bỏ giao dịch. Nếu máy khách thực hiện tìm được khoảng giao duy nhất và thực hiện bàn giao giao dịch với nhãn thời gian bàn giao bằng cận dưới của khoảng giao tìm được.

Quá trình chuẩn bị dữ liệu của giao dịch được mô tả chi tiết trong giải thuật 2 (algorithm2) như sau:

Giải thuật 2 Pha chuẩn bị

Dữ liệu: Giao dịch T_i với $txnID$, tập đọc WS, bộ đệm c, bộ lưu trữ str

#bước 1: kiểm tra dữ liệu đã bị khoá

```

1   for wop in WS
2       keyCacheInfo1 = c.KeyCacheInfo(wop)
3       if(keyCacheInfo1. $T_w$ )
4           if(keyCacheInfo1. $T_w.txnID > T_i.txnID$ )
5               pendingList.add( $T_i.txnID$ )
6               return
7           else
8               return ERRORCODE_TXN_WAIT_DIE
9           end
10      else
11          if(keyCacheInfo1. $T_r$  and keyCacheInfo1. $T_r.upperTS == +\infty$ )
12              if(keyCacheInfo1. $T_r.txnID > T_i.txnID$ )
13                  add to pending list
14                  return
15              else
16                  return ERRORCODE_TXN_WAIT_DIE
17              end
18      end

```

```

19     end
20 end

#bước 2: tính toán nhãn khoảng thời gian bàn giao của giao dịch với các dữ liệu trong
tác vụ đọc.
21     for rop in opMap
22         keyCacheInfo2 = c.KeyCacheInfo(rop.key)
23          $lowerTS = \max(lowerTS, keyInfo(rop).write_{ts})$ 
24         if(keyCacheInfo2.writeTS > keyInfo(rop).writets)
25              $upperTS = \min(upperTS, keyCacheInfo2.writeTS - 1)$ 
26         end
27         if(keyCacheInfo2.writeTS)
28              $upperTS = \min(upperTS, keyCacheInfo2.T_w.lowerTS - 1)$ 
29         end
30     end

#Bước 3: Tính toán khoảng nhãn thời gian bàn giao của giao dịch với các dữ liệu thuộc
các tác vụ ghi
32     for wop1 in WS
33         keyCacheInfo3 = c.KeyCacheInfo(wop1.key)
34         if(!keyCacheInfo3)
35             item = str(wop1.key)
36             if(item)
37                 keyCacheInfo3.writeTS= str(wop1.key).wts
38                 keyCacheInfo3.readTS = str(wop1.key).rts
39             else
40                 keyCacheInfo3.writeTS = 0
41                 keyCacheInfo3.readTS = 0
42             end
43     end

```

```

44         lowerTS = max(lowerTS, keyCacheInfo3.readTS + 1)
45         if(keyCacheInfo3.  $T_r$ )
46             lowerTS = max(lowerTS, keyCacheInfo3.  $T_r$ .upperTS + 1)
47         end
48         if(opMap(wop1.key))
49             opMap(wop1.key).isWriteOp = true
50         else
51             initialize KeyInfo with key = x, isReadOp = true, isWriteOp = false,
            wts = keyCacheInfo3.writeTS
52             opMap.add(KeyInfo)
53         end
54     end
#Bước 4 kiểm tra khoảng nhãn thời gian bàn giao
55     if(upperTS < lowerTS)
56         abort()
57     end
#Bước 5: thực hiện khoá dữ liệu của giao dịch
58     for opInfo in opMap
59         keyCacheInfo4 = c.KeyCacheInfo(opInfo.key)
        #Bước 5.1 khoá dữ liệu ghi
60         if(opInfo.key.isWriteOp)
61              $T_w = T_i$ 
62             RS.clear
63         else
        #Bước 5.2 khoá dữ liệu đọc
64             if(opInfo.key.isReadOp)
65                 keyCacheInfo4.RS.add( $T_i$ )

```

```

66          end
67          if((!keyCacheInfo4.Tr) or (keyCacheInfo4.Tr.upperTS <
          Ti.upperTS))
68              Tr = Ti
69          end
70      end
71  end

```

c. Pha bàn giao (commit phase):

Máy khách thực hiện bàn giao giao dịch với nhãn thời gian bàn giao tìm được hoặc thực hiện hủy bỏ giao dịch đến tất cả các máy chủ thực hiện các tác vụ của giao dịch trong giải thuật 3 (algorithm 3).

Nếu khối giao dịch của máy chủ nhận được bản tin bàn giao thì thực hiện bàn giao dữ liệu đã chuẩn bị trước đó trong từng nhân và thực hiện cập nhật nhãn thời gian bàn giao (commitTS) vào trong khối lưu trữ (storage).

- Nếu tác vụ thực hiện đọc dữ liệu thì $\text{storage.key.rts} = \max(\text{rts}, \text{commitTS})$.
- Nếu tác vụ thực hiện ghi dữ liệu thì $\text{storage.key.wts} = \text{storage.key.rts} = \text{commitTS}$

Sau đó thực hiện giải phóng các dữ liệu đã được khóa trước đó như sau:

- Đối với những tác vụ ghi của giao dịch, thực hiện cập nhật các thông số trong KeyCacheInfo với $T_w = \text{null}$, $\text{readTS} = \text{writeTS} = \text{commitTS}$.
- Đối với những dữ liệu đọc trong tác vụ đọc của giao dịch, thực hiện cập nhật các thông số trong KeyCacheInfo $\text{readTS} = \max(\text{readTS}, \text{commitTS})$, xóa giao dịch ra khỏi RS. Nếu giao dịch hiện tại trùng với T_r thì thực hiện cập nhật lại T_r thuộc RS với cận trên lớn nhất.

Nếu khối giao dịch của máy chủ nhận được bản tin yêu cầu hủy bỏ giao dịch thì thực hiện hủy bỏ dữ liệu đã chuẩn bị trước đó và thực giải phóng các dữ liệu đã khóa của giao dịch như sau:

- Đối với những dữ liệu trong tác vụ ghi của giao dịch, thực hiện cập nhật các thông số trong KeyCacheInfo với $T_w = \text{null}$.
- Đối với những dữ liệu trong tác vụ đọc của giao dịch, thực hiện cập nhật các thông số trong KeyCacheInfo thông qua xóa giao dịch ra khỏi RS. Nếu giao dịch hiện tại trùng với T_r thì thực hiện cập nhật lại T_r thuộc RS với cận trên lớn nhất.

Sau đó tiếp tục thực hiện các lệnh trong danh sách chờ.

Giải thuật 3 Pha bàn giao

Dữ liệu: giao dịch T_i với nhãn thời gian bàn giao commitTS, bộ đếm c, bộ lưu trữ str

```

1      str.commit( $T_i$ .txnID, commitTS)
2      for op in opMap
          # $T_i$  được huỷ bỏ với commitTS = 0
3      if(commitTS == 0)
4          if(op.key.isWriteOp)
5               $T_w$  = null
6          else
7              if(op.key.isReadOp)
8                  RS.erase( $T_r$ )
9              end
10         end
11     else
          # $T_i$  được bàn giao với commitTS#0
12     if(op.key.isWriteOp)
13         readTS = writeTS = commits
14          $T_w$  = null
15     else
16         if(op.key.isReadOp)
17             readTS = max(readTS, commitTS)
18             RS.erase( $T_i$ )
19         end
20     end

```

```

21          if( $T_r == T_i$ )
22               $T_r = \text{RS.maxUpper}()$ ;
23          end
24      end
25  end

```

Ví dụ thực hiện giải pháp hữu ích

Các tác giả giải pháp hữu ích đã thực hiện thử nghiệm một giao dịch A trong hệ thống quản lý dữ liệu phân tán đa vi xử lý có sử dụng phương pháp điều khiển phân tranh theo giải pháp hữu ích.

Hình 1 thể hiện quá trình thực hiện pha đọc của giao dịch A trong hệ thống quản lý dữ liệu có sử dụng phương pháp điều khiển phân tranh theo giải pháp hữu ích. Như được thể hiện trên Hình 1, pha đọc được thực hiện như sau:

- Máy khách thực hiện đọc dữ liệu x đến máy chủ 1 và máy chủ 2.
- Máy chủ 1 thực hiện điều hướng đến nhân 1, máy chủ 2 thực hiện điều hướng đến nhân 3 để thực hiện thăm tra và trả về dữ liệu bao gồm key x, giá trị và nhãn thời gian 2 cho máy khách.

Hình 2 và Hình 3 thể hiện quá trình thực hiện pha chuẩn bị và pha bàn giao của giao dịch A trong hệ thống quản lý dữ liệu có sử dụng phương pháp điều khiển phân tranh theo giải pháp hữu ích.

Cụ thể, pha chuẩn bị của giao dịch A được thực hiện như sau:

- + Sau khi nhận dữ liệu từ pha đọc của máy chủ 1 và máy chủ 2, máy khách thực hiện chuẩn bị dữ liệu x để gửi đến máy chủ 1 và máy chủ 2, thực hiện chuẩn bị dữ liệu y để gửi đến máy chủ 1 và máy chủ 3.
- + Máy chủ 1 thực hiện điều hướng tác vụ với dữ liệu x đến nhân 1, tác vụ với dữ liệu y đến nhân 2. Máy chủ 2 thực hiện điều hướng tác vụ với dữ liệu x đến nhân 3. Máy chủ 3 thực hiện điều hướng tác vụ với dữ liệu y đến nhân 2.
- + Tại máy chủ 1 nhân 1 và máy chủ 2 nhân 3, giao dịch A với định danh 1004 sẽ thực hiện cập nhật $lowerTS = \max(0, 2) = 2$. Trong quãng thời gian từ lúc giao dịch A thực hiện đọc và giao dịch A thực hiện chuẩn bị, dữ liệu x đã bị biến đổi bởi một giao dịch khác với nhãn thời gian bàn giao 9 do giá trị $writeTS$ trong bộ đệm bị thay đổi do đó thực hiện cập nhật $upperTS = \min(+\infty, 9 - 1) = 8$.

- + Tại máy chủ 1 nhân 2 và máy chủ 3 nhân 2, giao dịch với định danh 1006 đã thực hiện bàn giao với nhãn thời gian 6 do đó giá trị rts trong khối lưu trữ và $readTS$ trong bộ đệm được cập nhật thành 6. Tại thời điểm này giao dịch A thực hiện pha chuẩn bị cho dữ liệu y thông qua quá trình cập nhật $lowerTS = \max(0, 6 + 1) = 7$ và $lowerTS = \max(7, 7 + 1) = 8$.

Pha bàn giao của giao dịch A được thực hiện như sau:

- + Máy chủ 1 tính khoảng giao nhãn thời gian bàn giao cho dữ liệu x $[2, 8]$ và khoảng nhãn thời gian bàn giao cho dữ liệu y $[8, +\infty]$ là $[8, 8]$ và sau đó trả về khoảng tính toán nhãn thời gian bàn giao $[8, 8]$ của giao dịch A cho máy khách.
- + Máy chủ 2 thực hiện trả về khoảng bàn giao $[2, 8]$ của giao dịch A cho máy khách.
- + Máy chủ 3 thực hiện trả về khoảng bàn giao $[8, +\infty]$ của giao dịch A cho máy khách.
- Pha bàn giao được biểu diễn trong hình 3 như sau:
- + Máy khách sau khi nhận được khoảng bàn giao $[8, 8]$ từ máy chủ 1, $[2, 8]$ từ máy chủ 2, $[8, +\infty]$ từ máy chủ 3, sẽ tính toán khoảng giao là $[8, 8]$ và chọn cận dưới 8 là nhãn thời gian bàn giao cho giao dịch A.
- + Máy khách thực hiện gửi bản tin bàn giao với nhãn thời gian bàn giao 8 đến các máy chủ 1, 2, 3.
- + Các máy chủ thực hiện điều hướng đến các nhân chứa dữ liệu x, y tương ứng và thực hiện cập nhật giá trị trong khối lưu trữ và trong bộ đệm, xóa bỏ giao dịch như được thể hiện trong hình 3.

Hiệu quả đạt được của giải pháp hữu ích

Phương pháp điều khiển phân tranh theo giải pháp hữu ích đảm bảo được tính khả tuần tự trong hệ thống quản lý dữ liệu phân tán đa vi xử lý thông qua quá trình tính toán nhãn thời gian logic dựa vào phương pháp điều khiển phân tranh lạc quan và phương pháp khóa hai pha phân tán. Các hiệu quả đạt được khi sử dụng phương pháp này bao gồm:

- quá trình thực hiện giao dịch có thể thực hiện trong một thời gian trễ trọn vòng trong hệ thống phân tán đa xử lý;
- tránh được thắt nút cổ chai trong quá trình cấp phát nhãn thời gian do không sử dụng bộ phận điều phối; tránh được quá trình phải đồng bộ thời gian giữa các máy chủ vì sử dụng nhãn thời gian logic thay vì sử dụng nhãn thời gian vật lý;
- tránh được quá trình khóa chết (deadlock) là hiện tượng hay gặp trong hệ thống phân tán vì dữ liệu được lưu trữ và sao chép ở các nhân thuộc máy chủ khác nhau do áp dụng cơ chế wait-die;

- giảm thiểu được quá trình hủy bỏ giao dịch do cơ chế cấp phát và tính toán các khoảng nhần thời gian trong các máy chủ là hợp lý làm tăng hiệu suất khi số lượng lớn giao dịch cùng tác động đến một mục dữ liệu.

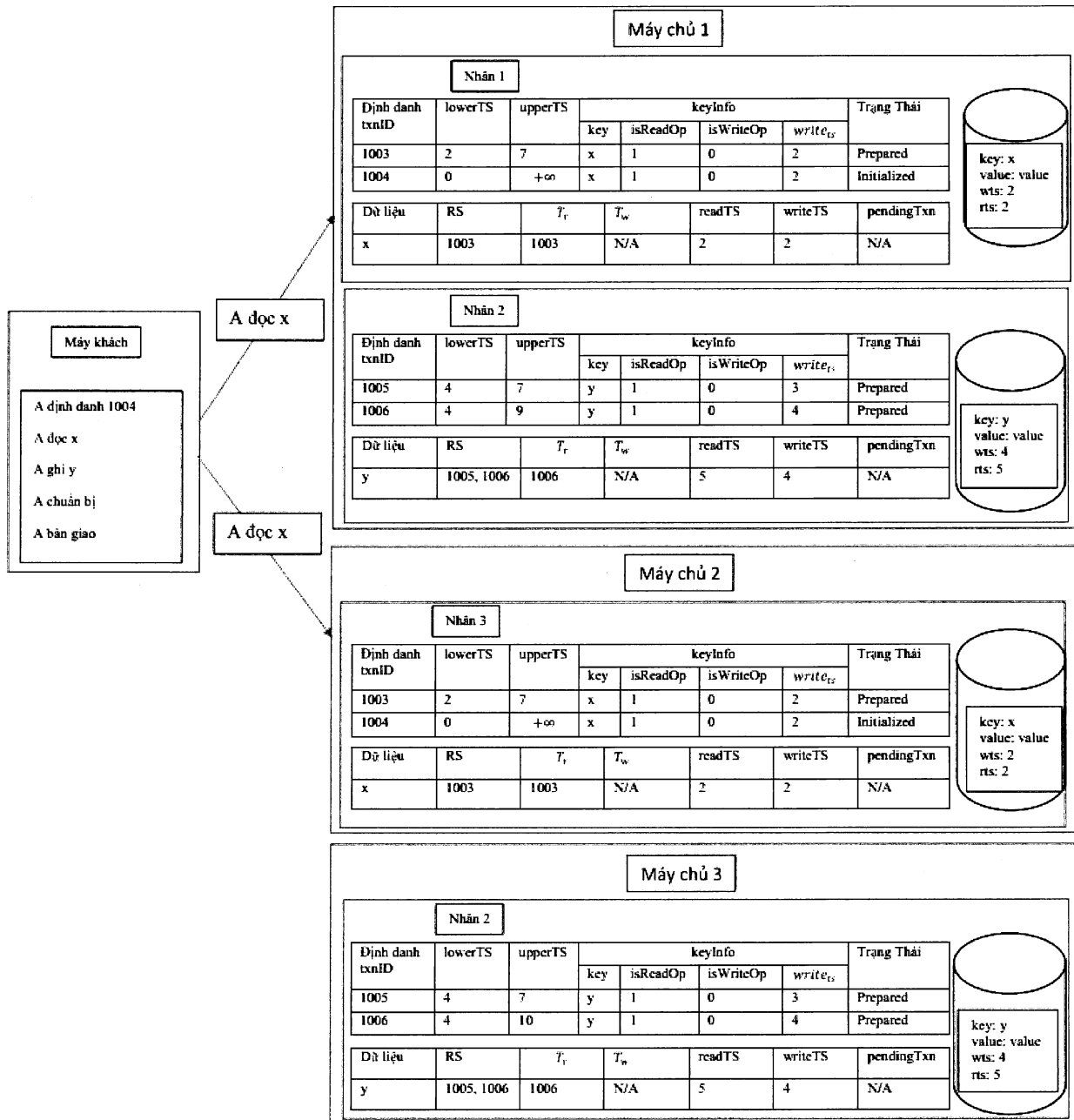
YÊU CẦU BẢO HỘ

1. Phương pháp điều khiển phân tranh trong hệ thống quản lý dữ liệu phân tán đa vi xử lý khi thực hiện nhiều giao dịch đồng thời thông qua quá trình tính toán nhân thời gian logic, bao gồm:

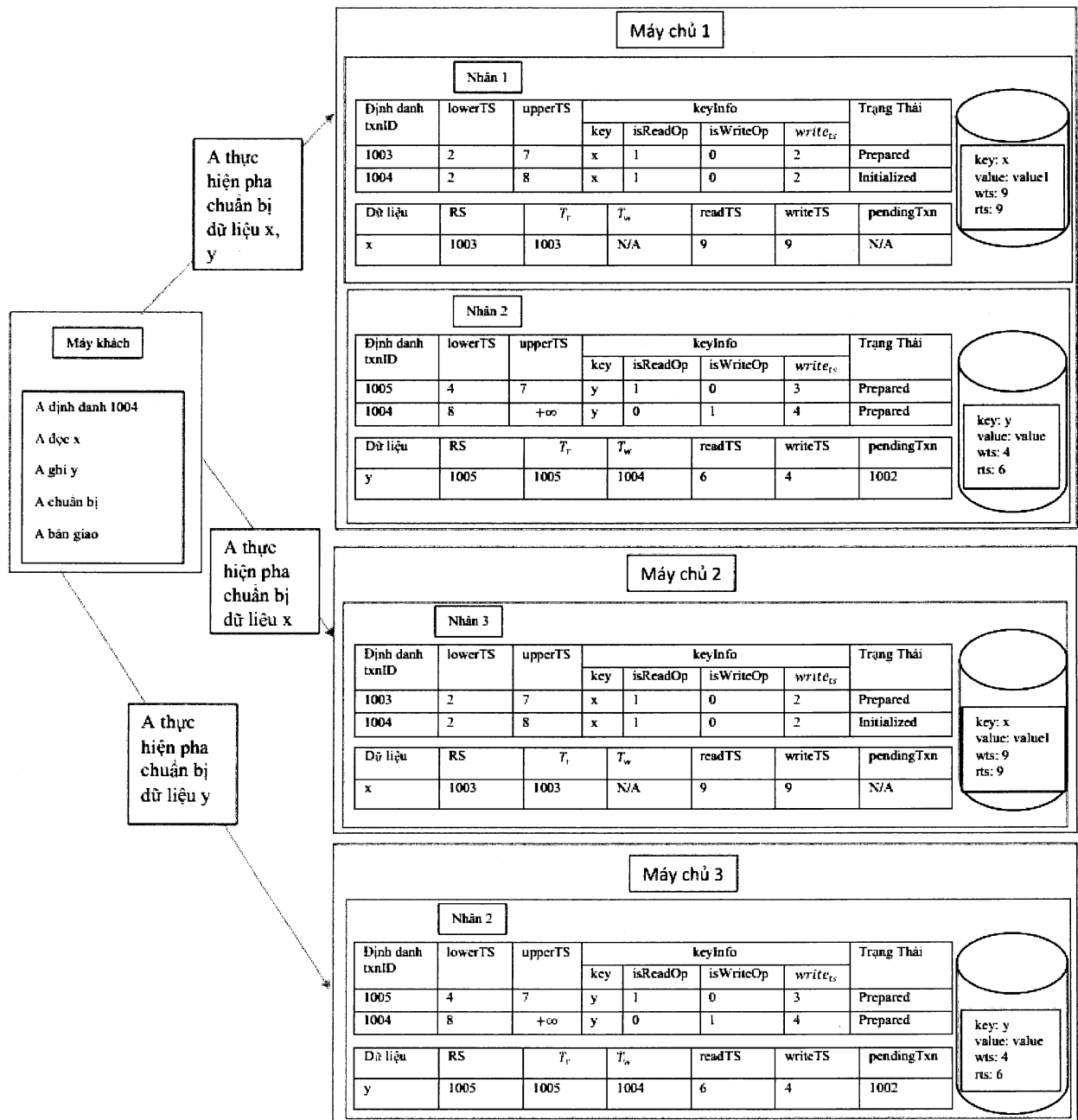
- thực hiện điều khiển phân tranh tại các nhân của các máy chủ; máy khách thực hiện điều hướng các tác vụ đến các máy chủ tương ứng và các máy chủ nhận được yêu cầu thực hiện tác vụ sẽ thực hiện điều hướng đến nhân chứa dữ liệu để xử lý;
- thực hiện khóa chốt cứng cho dữ liệu ghi và khóa chốt mềm cho dữ liệu đọc trong pha chuẩn bị để khóa dữ liệu từ lúc thực hiện pha chuẩn bị thành công đến lúc hoàn thành giao dịch trong đó:
 - + tại một thời điểm, có thể có một hoặc nhiều giao dịch đọc một dữ liệu nhưng chỉ có nhiều nhất một giao dịch thực hiện ghi một dữ liệu;
 - + khi một dữ liệu ghi bị khóa chốt cứng, bất kì một tác vụ đọc nào của giao dịch khác chứa dữ liệu sẽ được đưa vào hàng đợi giúp cho giao dịch ghi có thể được thực hiện trong một thời gian trễ tròn vòng; khi một dữ liệu bị khóa chốt cứng hoặc chốt mềm, bất kì một tác vụ ghi chứa dữ liệu nào của giao dịch khác thực hiện tác vụ sẽ được thực hiện theo cơ chế wait-die dựa vào định danh của các giao dịch để tránh khóa chết trong các nhân của các máy chủ; cơ chế wait-die được thực hiện xem xét nếu giao dịch đang xem xét có định danh nhỏ hơn giao dịch đang thực hiện khóa dữ liệu thì thực hiện đưa giao dịch vào trong một hàng đợi ưu tiên, ngược lại thì máy chủ sẽ trả về mã lỗi cho máy khách để yêu cầu thực hiện lại giao dịch;
- thực hiện điều khiển phân tranh và sắp xếp trình tự giữa các giao dịch thông qua quá trình tính toán khoảng nhân thời gian logic tại từng nhân của máy chủ, và tại từng máy chủ trong pha chuẩn bị trong khi máy khách sẽ thực hiện quá trình quyết định nhân thời gian hợp lý để thực hiện giao dịch hoặc hủy bỏ giao dịch trong pha bàn giao;
- quá trình tính toán khoảng nhân thời gian tại các nhân của máy chủ được thực hiện trong pha chuẩn bị bao gồm các bước sau:
 - + đối với dữ liệu đọc của giao dịch, thực hiện tính toán cận dưới bằng giá trị nhân thời gian ghi tại thời điểm dữ liệu đã được đọc để thực hiện sắp xếp giao dịch đang được xem xét sau các giao dịch đã thực hiện thành công trước đó; nếu dữ liệu bị khóa ghi bởi một giao dịch khác thì thực hiện tính toán cận trên bằng giá trị nhỏ nhất của cận trên hiện tại với giá trị liên trước giá trị cận dưới của giao dịch khóa ghi dữ liệu đó để thực hiện sắp xếp giao dịch đọc đứng trước giao dịch ghi đã được chuẩn bị thành công; nếu có các giao dịch khác thực hiện ghi dữ liệu thành công từ lúc giao dịch hiện tại thực hiện pha đọc đến lúc giao dịch hiện tại thực hiện pha chuẩn bị thì thực hiện cập nhật giá trị cận trên bằng giá trị nhỏ nhất của giá trị cận

trên hiện tại và giá trị liền trước của nhãn thời gian bàn giao của giao dịch thành công cuối cùng trong các giao dịch thành công đó để thực hiện sắp xếp giao dịch hiện tại liền trước các giao dịch thực hiện ghi dữ liệu thành công sau khi dữ liệu được đọc;

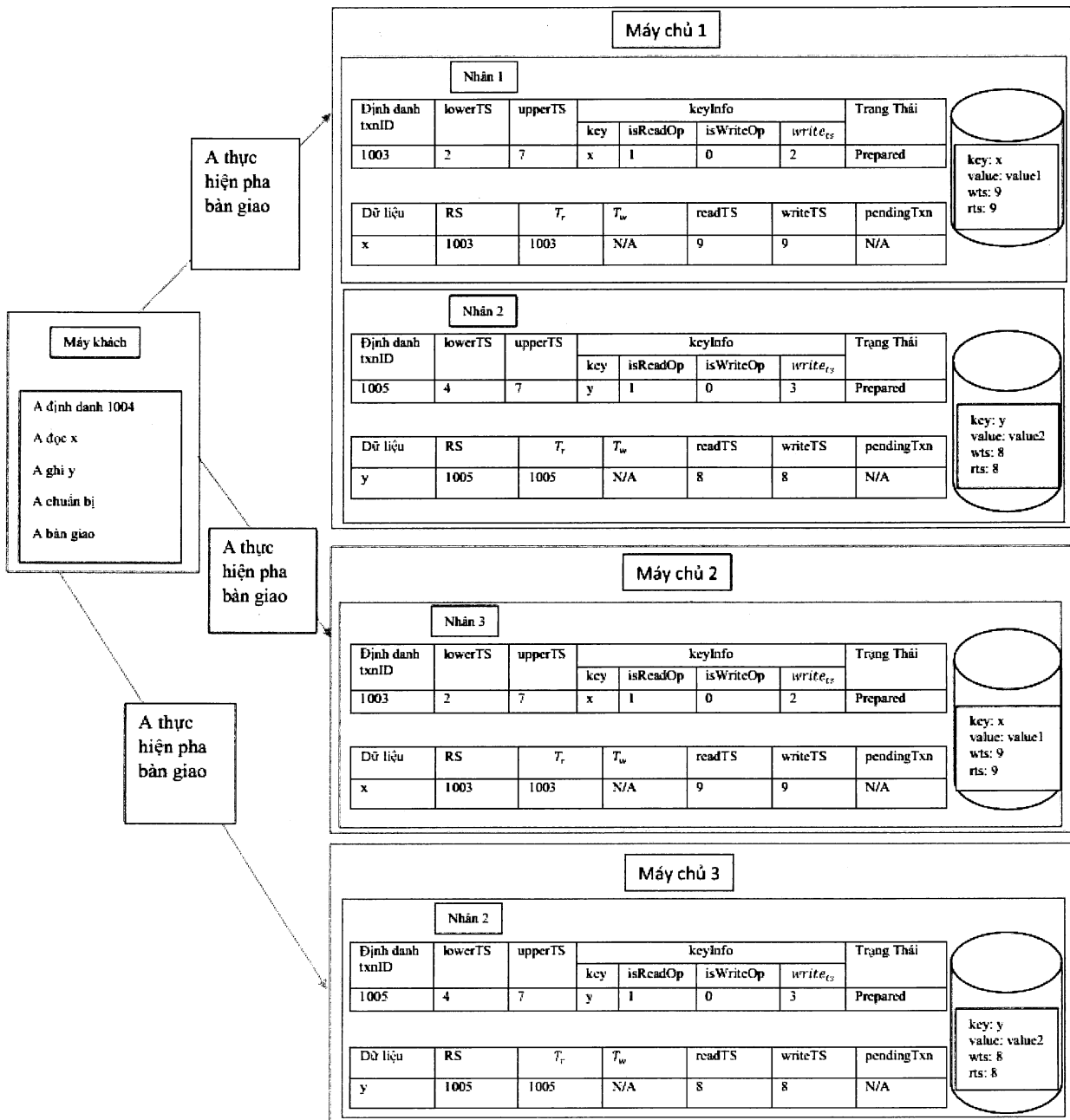
- + đối với dữ liệu ghi của giao dịch, thực hiện cập nhật cận dưới của giao dịch này bằng giá trị nhỏ nhất của cận dưới hiện tại, giá trị liền sau giá trị cận trên của tất cả các giao dịch đã thực hiện pha chuẩn bị đọc dữ liệu thành công và giá trị liền sau giá trị cận trên của tất cả các giao dịch đã hoàn thành đọc dữ liệu nhằm mục đích sắp xếp giao dịch ghi đứng sau các giao dịch đọc đã thực hiện pha chuẩn bị thành công hoặc hoàn thành giao dịch đọc thành công;
- quá trình tính toán khoảng nhãn thời gian tại các máy chủ được thực hiện trong pha chuẩn bị như sau:
 - + sau khi tất cả các dữ liệu của một giao dịch thực hiện pha chuẩn bị thành công, máy chủ thực hiện tìm khoảng giao của giao dịch ở tất cả các nhân và gửi khoảng giao này về cho máy khách; nếu máy chủ không tìm được khoảng giao nào phù hợp cho giao dịch thì máy chủ thực hiện gửi về cho máy khách một mã lỗi;
- quá trình tính toán nhãn thời gian duy nhất cho một giao dịch để hoàn thành giao dịch trong pha bàn giao bao gồm:
 - + đối với từng dữ liệu, sau khi đa số các máy chủ thực hiện dữ liệu đó trả về giá trị khoảng thời gian bàn giao, máy khách thực hiện tìm khoảng giao của đa số các máy chủ chứa dữ liệu làm khoảng giao của dữ liệu đó;
 - + sau đó, máy khách thực hiện tìm khoảng giao của tất cả các dữ liệu; nếu bất kì dữ liệu nào mà máy khách không tìm được khoảng giao thì máy khách thực hiện hủy bỏ giao dịch đến tất cả các máy chủ thực hiện và chứa đựng dữ liệu của giao dịch đó thông qua pha bàn giao; ngược lại, nếu máy khách tìm được khoảng giao của tất cả các dữ liệu, máy khách sẽ quyết định lấy cận dưới của khoảng giao làm nhãn thời gian bàn giao cho giao dịch và thực hiện hoàn thành giao dịch cùng với nhãn thời gian tìm được thông qua pha bàn giao.



Hình 1



Hình 2



Hình 3