



(12) BẢN MÔ TẢ SÁNG CHẾ THUỘC BẰNG ĐỘC QUYỀN SÁNG CHẾ

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

(11)



1-0031258

(51)^{2018.01} G06F 7/00; G06F 17/30

(13) B

(21) 1-2019-03461

(22) 28/06/2019

(45) 25/03/2022 408

(43) 25/10/2019 379A

(73) Tập đoàn Công nghiệp - Viễn thông Quân Đội (VN)

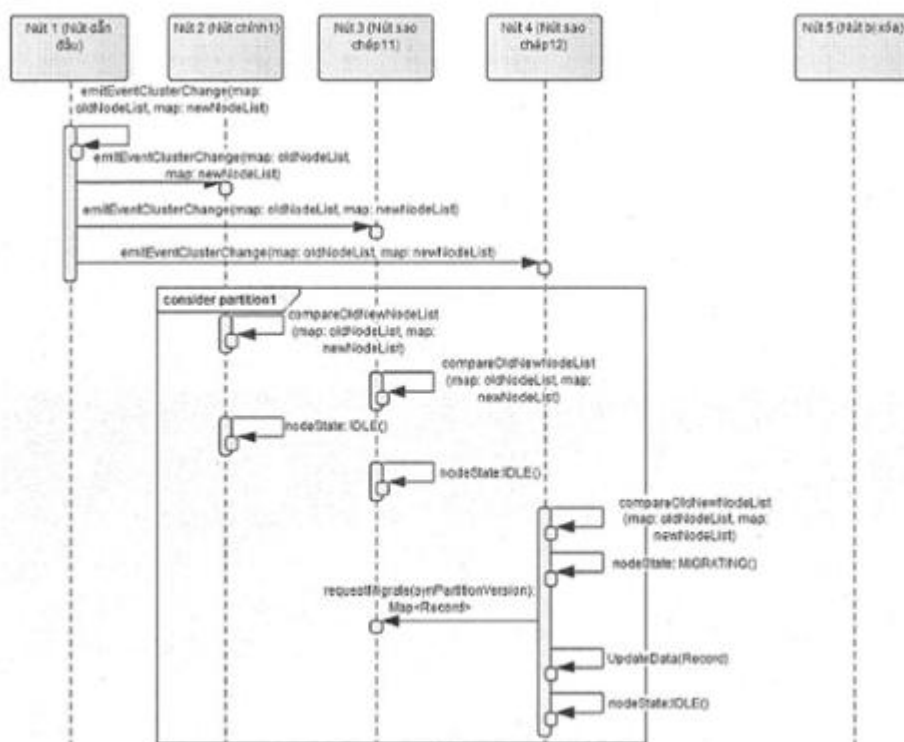
Số 1 đường Trần Hữu Dực, phường Mỹ Đình 2, quận Nam Từ Liêm, thành phố Hà Nội

(72) Lương Thị Chuyên (VN); Vũ Đức Chính (VN); Đoàn Khả Cường (VN); Nguyễn Hoàng Quân (VN).

(74) Công ty TNHH Tư vấn Quốc Dân (NACILAW)

(54) PHƯƠNG PHÁP CHUYỂN ĐỔI DỮ LIỆU VÀ ĐỒNG BỘ DỮ LIỆU ĐỊNH KÌ DỰA VÀO SỰ SAI KHÁC

(57) Sáng chế đề cập đến phương pháp chuyển đổi và đồng bộ dữ liệu định kì trong hệ thống quản lý dữ liệu phân tán đa vi xử lý khi cấu hình cụm thay đổi hoặc không thể sao chép dữ liệu do có lỗi trên đường truyền mạng để đảm bảo tính nhất quán dữ liệu cuối cùng với dung lượng truyền qua mạng tối thiểu dựa trên quá trình đánh dấu dữ liệu đã được đồng bộ thành công và dựa trên sự sai khác siêu dữ liệu của dữ liệu. Phương pháp này bao gồm các bước: thực hiện so sánh phiên bản của các phân vùng; thực hiện lưu trữ dữ liệu đối với từng nhóm sao chép; thực hiện quá trình chuyển đổi dữ liệu và đồng bộ dữ liệu định kì.



Lĩnh vực kỹ thuật được đề cập

Sáng chế đề cập đến phương pháp chuyển đổi dữ liệu giữa các nút khi cấu hình cụm thay đổi và thực hiện đồng bộ dữ liệu (synchronize) định kì khi cấu hình cụm ổn định theo sự sai khác dữ liệu (delta) để đảm bảo được tính nhất quán của dữ liệu trong hệ thống khi thực hiện thêm, bớt nút hoặc đường truyền mạng phát sinh lỗi với lượng dữ liệu gửi đi tối thiểu trong hệ thống dữ liệu phân tán đa vi xử lý.

Tình trạng kỹ thuật của sáng chế

Trong hệ thống quản lý dữ liệu phân tán, dữ liệu được phân chia, lưu trữ và sao chép tại các phân vùng (partition) thuộc nút khác nhau trong cụm. Hiện nay, mỗi nút trong cụm đều chứa một lượng lớn dữ liệu. Khi hệ thống cơ sở dữ liệu bị thay đổi cấu hình như quá trình mất nút hoặc thêm nút, dữ liệu sẽ được phân bổ lại. Ngoài ra, khi cấu hình hệ thống ổn định nhưng do lỗi đường truyền gây ra dữ liệu không nhất quán trong cụm. Do đó, cần phải có các phương pháp chuyển đổi dữ liệu phù hợp để lượng dữ liệu gửi đi tối thiểu tránh tắc nghẽn đường truyền và đảm bảo được tính nhất quán của dữ liệu trong cụm.

Hiện nay, để thực hiện chuyển đổi dữ liệu tối ưu giữa các nút dựa vào phiên bản của các phân vùng (partition version) như trên Aerospike (Aerospike là một cơ sở dữ liệu Key Value Store (KVS là kiểu lưu trữ dữ liệu đơn giản dưới dạng cặp khóa (key) liên kết với giá trị (value)) có tốc độ rất cao và tốc độ nhanh gấp nhiều lần các KVS thông thường hay được sử dụng trên thị trường. Nếu phiên bản của các phân vùng của nút nhỏ hơn phiên bản của các phân vùng của nút khác thì chỉ có chiều chuyển dữ liệu thuộc phân vùng từ các nút khác đến nút đó. Tuy nhiên, toàn bộ dữ liệu của phân vùng sẽ được gửi đi. Trong toàn bộ dữ liệu gửi chứa cả dữ liệu có trong nút cần cập nhật và dữ liệu mới. Do đó, lượng dữ liệu dư thừa vẫn được gửi đi giữa các nút trong cùng một phân vùng. Nếu phiên bản của các phân vùng giống nhau và cần thực hiện chuyển đổi dữ liệu (migrate) thì thực hiện gửi và so sánh toàn bộ siêu dữ liệu (metadata) của dữ liệu trong nút cần chuyển đổi dữ liệu và nút thực hiện chuyển đổi dữ liệu của phân vùng tương ứng để xác định dữ liệu cần chuyển đổi dữ liệu. Giá trị của phiên bản của các phân vùng được tính toán khi cấu hình cụm có sự thay đổi. Đặc biệt, khi dữ liệu không được sao chép thành công do lỗi đường truyền mạng, dữ liệu trong cả cụm của

phiên bản của các phân vùng sẽ không đồng nhất ở một thời điểm nào đó. Một số cơ sở dữ liệu (database) khác như Cassandra, Dynamo, Riak thực hiện quá trình chuyển đổi dữ liệu thông qua hai bước xây dựng cây Merkle (Merkle Tree) và truyền nhận dữ liệu. Cây Merkle được xây dựng để làm giảm số lượng dữ liệu được truyền khi kiểm tra tính nhất quán giữa các nhóm sao chép và giảm quá trình đọc dữ liệu từ ổ đĩa khi thực hiện quá trình đồng bộ. Mỗi nút trong cụm xây dựng một cây Merkle với các nhánh khoảng khóa bao gồm tập các khóa của từng phân vùng. Khi nút được thêm hoặc rời khỏi cụm, khoảng khóa thay đổi nên cây Merkle phải được tính toán lại.

Bản chất kỹ thuật của sáng chế

Để khắc phục các nhược điểm nêu trên, sáng chế đề xuất phương pháp chuyển đổi và đồng bộ dữ liệu định kì dựa vào sự sai khác (cơ chế delta) về dữ liệu dựa vào việc lưu trữ và so sánh giá trị phiên bản của các phân vùng của dữ liệu trên các nút. Theo phương pháp này, chỉ những dữ liệu sai khác dựa vào sự so sánh của phiên bản của các phân vùng và sự so sánh phiên bản (version) của dữ liệu được di chuyển. Phương pháp đề xuất những nội dung sau: định dạng và cách thức tự động cập nhật phiên bản của các phân vùng, cách thức thực hiện chuyển đổi dữ liệu trong các phân vùng khi cấu hình cụm thay đổi và cách thức thực hiện đồng bộ dữ liệu định kì.

Phương pháp chuyển đổi dữ liệu và đồng bộ dữ liệu định kì dựa vào sự sai khác như sau:

Tính toán phiên bản của các phân vùng; phiên bản của các phân vùng là một số M bit, trong đó N ($N < M$) bit đầu tiên là chỉ số đánh dấu sự thay đổi của cụm và M-N bit còn lại là chỉ số đồng bộ đánh dấu cho mỗi chu kì đồng bộ tại chỉ số N bit. Giá trị N thay đổi khi cấu hình cụm thay đổi như quá trình thêm nút, mất nút hoặc khi trạng thái cụm thay đổi khi thay đổi nút dẫn đầu (leader). Chỉ số đồng bộ được tăng dần sau mỗi chu kì. Khi chỉ số đánh dấu cụm N bit thay đổi thì chỉ số đồng bộ được thiết lập về 0. Nếu M-N bit bị tràn số thì tăng giá trị N bit thêm một đơn vị và chỉ số đồng bộ cũng thực hiện thiết lập lại về 0.

Thực hiện lưu trữ dữ liệu đối với từng nhóm sao chép theo định dạng `<synPartitionVersion <map<partitionVersion, map<metadata>>>>`. Trong đó synPartitionVersion là giá trị của phiên bản của các phân vùng đã được đồng bộ tại thời điểm gần nhất với thời điểm hiện tại, partitionVersion là giá trị phiên bản của các

phân vùng tại thời điểm dữ liệu được ghi, siêu dữ liệu (metadata) là thông tin bảng, khóa và giá trị phiên bản dữ liệu.

Nút đầu tiên trong danh sách nút thuộc phân vùng được coi là nút chính, các nút còn lại sẽ được coi là nút sao chép.

Đối với quá trình chuyển đổi dữ liệu thực hiện với từng phân vùng như sau:

- Nút mới trong nhóm sao chép thực hiện gửi yêu cầu chuyển đổi dữ liệu đến một trong các nút cũ tồn tại trong cả danh sách nút trước khi thay đổi cấu hình và sau khi thay đổi cấu hình kèm theo phiên bản của các phân vùng đã được đồng bộ gần nhất của nút.
- Nút nhận được yêu cầu chuyển đổi dữ liệu thực hiện nhận những siêu dữ liệu có phiên bản của các phân vùng lớn hơn phiên bản của các phân vùng trong bản tin nhận được, thực hiện truy cập vào trong nơi lưu trữ dữ liệu để lấy dữ liệu với siêu dữ liệu tương ứng để gửi trả lại nút cần được chuyển đổi dữ liệu.
- Nút cần được chuyển đổi dữ liệu sau khi nhận được dữ liệu, thực hiện cập nhật vào nơi lưu trữ dữ liệu.

Khi cụm ở trạng thái ổn định, quá trình đồng bộ được thực hiện định kì như sau:

- Nút dẫn đầu định kì thực hiện yêu cầu đồng bộ.
- Các nút khi nhận được yêu cầu đồng bộ sẽ thực hiện đối với từng phân vùng. Nút sao chép thực hiện gửi siêu dữ liệu chưa được đồng bộ về cho nút chính tương ứng.
- Nút chính sau khi nhận được toàn bộ siêu dữ liệu từ các nút sao chép thực hiện so sánh và tìm ra dữ liệu có phiên bản mới nhất, nút chứa dữ liệu mới nhất và cập các siêu dữ liệu và nút cần cập nhật dữ liệu tương ứng với siêu dữ liệu đó. Sau đó, nút chính thực hiện gửi bản tin pushData đến các nút sao chép kèm theo các thông tin sau:
 - + Dữ liệu mới nhất thuộc nút chính mà nút sao chép cần cập nhật.
 - + Các nút và danh sách siêu dữ liệu tương ứng với dữ liệu mới nhất mà nút sao chép cần phải di chuyển dữ liệu đến nút tương ứng.
- Nút nhận được bản tin pushData sẽ thực hiện:
 - + Nếu tồn tại dữ liệu cần cập nhật trong bản tin nhận được, nút thực hiện cập nhật dữ liệu vào nơi chứa dữ liệu.

+ Nếu tồn tại các cặp nút và danh sách siêu dữ liệu trong bản tin nhận được, nút thực hiện đọc các dữ liệu từ nơi lưu trữ dữ liệu tương ứng với các siêu dữ liệu nhận được và gửi đến nút cần cập nhật dữ liệu mới nhất này thông qua bản tin sendData.

- Nút nhận được bản tin sendData, thực hiện cập nhật dữ liệu mới nhất này vào trong nơi lưu trữ dữ liệu.
- Sau khi tất cả các nút trong phân vùng nhận được dữ liệu mới nhất, quá trình đồng bộ cho phân vùng hoàn thành, nút chính thực hiện cập nhật phiên bản của các phân vùng đã được đồng bộ mới nhất cho chính nó và cho các nút sao chép. Khi tất cả các phân vùng hoàn thành đồng bộ dữ liệu, quá trình đồng bộ dữ liệu của cụm sẽ được hoàn thành.

Mô tả vắn tắt các hình vẽ

Hình 1 là hình vẽ thể hiện quá trình thực hiện quá trình chuyển đổi dữ liệu khi nút sao chép 3 được thêm vào phân vùng 1.

Hình 2 là hình vẽ thể hiện quá trình thực hiện đồng bộ dữ liệu của phân vùng 1 khi hệ thống ổn định.

Mô tả chi tiết sáng chế

Tối ưu quá trình di chuyển dữ liệu đảm bảo tính nhất quán của dữ liệu được chia thành hai phần bao gồm quá trình chuyển đổi dữ liệu và quá trình đồng bộ dữ liệu định kì. Một nút tồn tại ba trạng thái:

Trạng thái	Ý nghĩa
IDLE	Khi nút bắt đầu được khởi tạo hoặc nút đã hoàn thành chuyển đổi dữ liệu hoặc hoàn thành xong quá trình đồng bộ.
MIGRATING	Nút đang thực hiện quá trình chuyển đổi dữ liệu.
SYNCHRONIZING	Nút đang thực hiện quá trình đồng bộ.

Định dạng và cách thức tự động cập nhật phân vùng được thể hiện như sau:

Phiên bản của các phân vùng là một số M bit, trong đó N ($N < M$) bit đầu tiên là chỉ số đánh dấu sự thay đổi của cụm và M-N bit còn lại là chỉ số đồng bộ đánh dấu cho mỗi chu kì đồng bộ tại chỉ số N bit. Giá trị N bit cập nhật giá trị khi xảy ra quá trình thêm nút, mất nút hoặc khi thay đổi nút dẫn đầu. Chỉ số đồng bộ được tăng dần sau

mỗi chu kì. Khi chỉ số đánh dấu cụm N bit thay đổi thì chỉ số đồng bộ được thiết lập về 0. Nếu M-N bit bị tràn số thì tăng giá trị N bit và chỉ số đồng bộ cũng thực hiện thiết lập lại về 0.

Thực hiện lưu trữ dữ liệu đối với từng nhóm sao chép theo dạng `<synPartitionVersion <map<partitionVersion, map<metadata>>>>`. Trong đó `synPartitionVersion` là giá trị của phiên bản của các phân vùng đã được đồng bộ tại thời điểm gần nhất với thời điểm hiện tại, `partitionVersion` là giá trị phiên bản của các phân vùng tại thời điểm dữ liệu được ghi, siêu dữ liệu là thông tin bảng, khóa và giá trị phiên bản dữ liệu.

a. Quá trình chuyển đổi dữ liệu.

Khi nút được thêm vào cụm hoặc nút rời ra khỏi cụm dẫn đến cấu hình của cụm thay đổi. Do đó, danh sách các nút trong phân vùng có thể bị thay đổi. Khi đó, dữ liệu trong một phân vùng cần phải được nhất quán với nhau. Những nút mới được thêm vào phân vùng phải lấy dữ liệu từ các nút tồn tại trong phân vùng hiện tại và trước khi cấu hình phân vùng thay đổi. Quá trình chuyển đổi dữ liệu được thực hiện như sau:

- Các nút hiện tại trong phân vùng thực hiện so sánh cấu hình cũ trước khi thay đổi và cấu hình mới sau khi thay đổi. Nếu nút hiện tại có trong cấu hình cũ và không có trong cấu hình mới thì chúng tỏ nút mới được thêm vào phân vùng. Nếu nút ở trạng thái IDLE hoặc SYNCHRONIZING, nút bắt đầu thực hiện quá trình chuyển đổi dữ liệu, kết thúc trạng thái hiện tại và chuyển trạng thái của nút sang trạng thái MIGRATING.
- Nút thêm mới vào phân vùng thực hiện gửi bản tin yêu cầu chuyển đổi dữ liệu đến nút thuộc cả cấu hình cũ và cấu hình mới kèm theo phiên bản của các phân vùng được đồng bộ thành công gần nhất `synPV` của nút.
- Nút nhận được bản tin chuyển đổi dữ liệu thực hiện lấy toàn bộ siêu dữ liệu có `partitionVersion` lớn hơn `synPV` và đọc dữ liệu từ nơi lưu trữ dữ liệu với siêu dữ liệu tương ứng gửi phản hồi lại nút cần chuyển đổi dữ liệu.
- Nút cần đồng bộ khi nhận được dữ liệu sẽ thực hiện ghi dữ liệu vào trong nơi lưu trữ dữ liệu và đánh dấu phân vùng đã được đồng bộ. Khi hoàn thành đồng bộ cho toàn bộ các phân vùng thuộc nút, nút thực hiện hoàn thành quá trình đồng bộ và chuyển sang trạng thái IDLE.

b. Quá trình đồng bộ dữ liệu định kì.

Khi hệ thống ổn định, quá trình đồng bộ được thực hiện định kì như sau:

- Nút dẫn đầu thực hiện gửi bản tin yêu cầu đồng bộ với phiên bản trước đó đến tất cả các nút trong hệ thống.
- Nút nhận được yêu cầu đồng bộ có trạng thái MIGRATING, nút thực hiện trả về kết quả đồng bộ không thành công về LEADER và kết thúc quá trình đồng bộ của nút. Nếu xem xét trạng thái có đang ở trạng thái IDLE hoặc SYNCRONIZING, nút nhận được yêu cầu đồng bộ thực hiện đồng bộ tiếp tục với quá trình xử lý bên dưới.
- Nếu nút nhận được yêu cầu đồng bộ là nút sao chép, nút thực hiện gửi siêu dữ liệu cần xem xét đồng bộ đến nút chính. Nếu nút nhận được yêu cầu đồng bộ là nút chính, sau khi nhận được toàn bộ siêu dữ liệu từ các nút sao chép và chính nó, nút chính thực hiện so sánh các siêu dữ liệu của chính nó và các siêu dữ liệu nhận được thông qua phiên bản chứa trong siêu dữ liệu để chọn ra những siêu dữ liệu tương ứng với dữ liệu mới nhất (siêu dữ liệu có giá trị phiên bản lớn nhất), nút chứa đựng dữ liệu mới nhất và nút chứa giá trị siêu dữ liệu thấp hơn cần cập nhật dữ liệu mới nhất đó.
- Nút chính thực hiện gửi bản tin pushData đến tất cả nút sao chép kèm theo:
 - + Dữ liệu mới nhất trong nơi lưu trữ dữ liệu thuộc nút chính mà nút sao chép cần được cập nhật.
 - + Danh sách map<node, map<metadata>> với siêu dữ liệu của dữ liệu mới nhất cần cập nhật cho nút tương ứng.
- Nút nhận được bản tin pushData thực hiện cập nhật dữ liệu mới nhất nếu có. Nếu tồn tại danh sách nút cần gửi dữ liệu mới nhất, nút thực hiện đọc dữ liệu từ nơi lưu trữ dữ liệu ứng với siêu dữ liệu trong bản tin và thực hiện gửi dữ liệu mới nhất đó đến nút tương ứng thông qua bản tin sendData.
- Nút nhận được bản tin sendData, nếu có dữ liệu trong bản tin thực hiện cập nhật dữ liệu mới nhất này vào trong nơi lưu trữ dữ liệu. Sau đó trả về kết quả cập nhật dữ liệu thành công.
- Khi tất cả các nút thực hiện trả về bản tin pushData thành công cho nút chính, nút chính thực hiện cập nhật phiên bản của các phân vùng đồng bộ thông qua quá trình gửi bản tin cập nhật phiên bản của phân vùng (updatePartitionVersion) bằng phiên bản của các phân vùng vừa đồng bộ.

Quá trình cập nhật phiên bản của các phân vùng ở các nút thuộc phân vùng thành công, nút chính thực hiện gửi trả thông báo đồng bộ cho phân vùng thành công đến nút dẫn đầu.

- Nút thực hiện đồng bộ cho tất cả các phân vùng hiện tại trong nút thành công sẽ chuyển sang trạng thái IDLE và kết thúc quá trình đồng bộ.

Ví dụ thực hiện sáng chế

Giả sử một cụm bao gồm năm nút (nút 1, nút 2, nút 3, nút 4, nút 5) với hệ số sao chép là 3, nút dẫn đầu (leader) là nút 1. Phân vùng 1 bao gồm nút 2, nút 3 và nút 5. Phiên bản của các phân vùng là một số gồm 64 bit có giá trị (4, 0), trong đó 44 bit đầu có giá trị 4 là chỉ số định danh cho cấu hình và trạng thái của cụm, 20 bit còn lại định danh cho quá trình đồng bộ định kì có giá trị 0 với chu kì đồng bộ 1 phút. Nhưng sau một thời gian nút 5 bị chết nên danh sách các nút của phân vùng 1 bị thay đổi thành nút 2, nút 3, nút 4. Phiên bản của các phân vùng có giá trị là (6, 0) bao gồm 44 bit đầu có giá trị 6, 20 bit sau có giá trị là 0. Trước đó, nút 4 có chứa phân vùng 1 với phiên bản của các phân vùng có giá trị (2, 100). Do đó, quá trình thực hiện chuyển đổi dữ liệu được xem xét trên phân vùng 1 được minh họa trong Hình 1 như sau:

- Nút dẫn đầu thực hiện phát ra thông báo thay đổi cấu hình emitEventClusterChange đến tất cả các nút trong cụm kèm theo cấu hình trước đó oldNodeList (nút 1, nút 2, nút 3, nút 4, nút 5) và cấu hình cụm hiện tại newNodeList (nút 1, nút 2, nút 3, nút 4).
- Khi nút trong cụm nhận được thông báo thay đổi cấu hình cụm, nút thực hiện xem xét tất cả các phân vùng thuộc nút và chuyển sang trạng thái MIGRATING. Quá trình chuyển đổi dữ liệu phân vùng 1 được mô tả chi tiết còn các phân vùng khác có sự thay đổi về mặt cấu hình được thực hiện tương tự, các phân vùng không có sự thay đổi về mặt cấu hình sẽ không cần thực hiện quá trình chuyển đổi dữ liệu.
- Phân vùng 1 trên các nút 2, nút 3, nút 4 thực hiện so sánh cấu hình phân vùng mới (nút 2, nút 3, nút 4) và cũ (nút 2, nút 3, nút 5) thông qua quá trình thực hiện hàm compareOldeNewNodeList để biết được nút 4 (nút sao chép 12) là nút mới của phân vùng 1 và nút 2 (nút chính 1), nút 3 (nút sao chép 11) là các nút cũ thuộc phân vùng.

- Nút sao chép 12 thực hiện gửi giá trị phiên bản của các phân vùng đã được đồng bộ ở chu kì gần nhất (2, 100) đến nút sao chép 11 thông qua bản tin yêu cầu chuyển đổi dữ liệu (requestMigrate).
- Nút sao chép 11 sau khi nhận được bản tin requestMigrate từ nút sao chép 12, thực hiện lấy hết tất cả các khóa thuộc phân vùng 1 có phiên bản của các phân vùng lớn hơn (2, 100) gửi sang nút sao chép 12.
- Nút sao chép 12 khi nhận được dữ liệu, thực hiện cập nhật (update) dữ liệu vào trong nơi lưu trữ dữ liệu thông qua quá trình gọi hàm cập nhật dữ liệu (UpdateData), thông báo hoàn thành xong quá trình chuyển đổi dữ liệu tại phân vùng 1.
- Nếu nút thực hiện xong toàn bộ quá trình chuyển đổi dữ liệu cho tất cả các phân vùng của nút, nút thực hiện chuyển sang trạng thái IDLE để hoàn thành quá trình chuyển đổi dữ liệu.

Một phút sau khi hệ thống chạy ổn định với cấu hình không đổi gồm nút 1, nút 2, nút 3, nút 4, nút dẫn đầu (nút 1) thực hiện gửi yêu cầu đồng bộ hệ thống tại partitionVersion (6, 1) đến chính nó và các nút còn lại trong hệ thống thông qua bản tin bắt đầu đồng bộ (startSynchronization). Các nút sau khi nhận được bản tin yêu cầu đồng bộ từ nút dẫn đầu thực hiện chuyển sang trạng thái SYNCHRONIZING và bắt đầu thực hiện đồng bộ dữ liệu của các phân vùng trên chính nó. Quá trình đồng bộ của phân vùng 1 được minh họa chi tiết ở Hình 2 dưới đây, các phân vùng khác trong hệ thống được thực hiện tương tự. Các nút thực hiện xem xét phân vùng 1 với thông tin các bản ghi cần đồng bộ nút chính 1 với

(synPartitionVersion (5,600), < (partition version (5,800), < (bảng1 – key1, version 10) >), (partition version (5,900), < (bảng1 – key2, version 7) >) >)

, nút sao chép 11 với

(synPartitionVersion (5,600), < (partition version (5,800), < (bảng1 – key1, version 10) >), (partition version (5,1000), < (bảng1 – key2, version 8) >) >)

, nút sao chép 12 với

(synPartition (5,600), < (partitionVersion (5,700), < (bảng1 – key1, version 9) >), (partitionVersion (5,1000), < (bảng1 – key2, version 8) >) >)

:

- Nút sao chép 11 thực hiện gửi phản hồi danh sách siêu dữ liệu bao gồm $\langle (bảng1 - key1, version\ 10), (bảng1 - key2, version\ 8) \rangle$ cho nút chính 1 thông qua quá trình gửi bản tin requestSynchronization.
- Nút sao chép 12 thực hiện gửi phản hồi danh sách siêu dữ liệu bao gồm $\langle (bảng1 - key1, version\ 9), (bảng1 - key2, version\ 8) \rangle$ cho nút chính 1 thông qua quá trình gửi bản tin requestSynchronization.
- Nút chính 1 thực hiện so sánh danh sách siêu dữ liệu của chính nó $\langle (bảng1 - key1, version\ 10), (bảng1 - key2, version\ 7) \rangle$ danh sách siêu dữ liệu nhận được từ nút sao chép 11 và nút sao chép 12 thông qua quá trình thực hiện hàm compareRecordMetadata. Nút chính 1 chứa dữ liệu mới nhất của *bảng1 - key1*, nút sao chép 11 chứa dữ liệu mới nhất của *bảng1 - key2*, nút chính 1 cần cập nhật dữ liệu mới nhất *bảng1 - key2* và nút sao chép 12 cần cập nhật dữ liệu *bảng1 - key1*. Khi đó nút chính 1 thực hiện đọc dữ liệu *bảng1 - key1* từ nơi lưu trữ dữ liệu và gửi bản tin pushData tới nút sao chép 12 kèm theo dữ liệu đã đọc được, nút chính 1 thực hiện gửi bản tin pushData đến nút sao chép 11 kèm theo và danh sách $\langle nút\ 2, \langle (bảng1 - key2) \rangle \rangle$.
- Nút sao chép 12 khi nhận được dữ liệu bản ghi *bảng1 - key1* từ nút chính 1, thực hiện cập nhật dữ liệu mới nhất đó vào trong nơi lưu trữ dữ liệu.
- Nút sao chép 11 khi nhận được bản tin từ nút chính 1 thực hiện lấy dữ liệu từ nơi lưu trữ dữ liệu tương ứng với siêu dữ liệu $\langle (bảng1 - key2) \rangle$. Sau đó gửi bản tin sendData kèm theo dữ liệu bản ghi của $\langle (bảng1 - key2) \rangle$ vừa đọc được đến nút chính 1.
- Nút chính 1 nhận được dữ liệu từ bản tin sendData thực hiện cập nhật dữ liệu mới nhất vào trong nơi lưu trữ dữ liệu.
- Khi tất cả nút chính 1, nút sao chép 11, nút sao chép 12 hoàn thành cập nhật dữ liệu mới nhất, nút chính 1 thực hiện cập nhật giá trị phiên bản của các phân vùng đã đồng bộ thành công synPartitionVersion (6, 1) đến tất cả các nút trong phân vùng 1 thông qua quá trình gửi và thực hiện bản tin updatePartitionVersion.

- Sau khi cập nhật giá trị phiên bản của các phân vùng đã được đồng bộ, các nút đánh dấu hoàn thành đồng bộ cho phân vùng 1.

Khi nút thực hiện hoàn thành đồng bộ thành công cho tất cả các phân vùng thuộc nút, nút chuyển sang trạng thái IDLE.

Hiệu quả đạt được của sáng chế

Phương pháp chuyển đổi dữ liệu và đồng bộ dữ liệu định kì dựa vào sự sai khác tối ưu được dung lượng dữ liệu gửi qua mạng, đảm bảo được tính nhất quán của dữ liệu trong cụm không chỉ khi thực hiện thêm nút hoặc nút chết trong cụm mà còn khi quá trình sao chép dữ liệu bị lỗi do lỗi đường truyền mạng. Các hiệu quả đạt được khi sử dụng phương pháp này bao gồm:

- quá trình chuyển dữ liệu trong cụm khi cấu hình cụm thay đổi tối ưu dựa vào phiên bản của các phân vùng đã được đồng bộ. Chỉ những dữ liệu có phiên bản của các phân vùng lớn hơn giá trị phiên bản của các phân vùng đã được đồng bộ của nút cần cập nhật dữ liệu mới nhất được gửi đi. Do đó giảm thiểu số lượng dữ liệu chuyển qua mạng và thời gian chuyển dữ liệu cũng giảm.
- quá trình đồng bộ dữ liệu định kì giúp nhất quán dữ liệu cuối cùng khi quá trình sao chép dữ liệu giữa các bản sao trong một phân vùng bị lỗi. Nút chính đảm nhiệm nhiệm vụ xác định dữ liệu mới nhất và điều phối nút chứa dữ liệu mới nhất chuyển đến nút cần cập nhật dữ liệu mới nhất đó. Quá trình này giúp phân chia dữ liệu tới các nút tránh quá trình một nút phải đảm nhận việc chuyển dữ liệu. Nhiều nút làm đồng thời sẽ giúp giảm thời gian truyền dữ liệu.

Yêu cầu bảo hộ

1. Phương pháp chuyển đổi dữ liệu và đồng bộ dữ liệu định kì dựa vào sự sai khác bao gồm các bước:

thực hiện so sánh giá trị của phiên bản của các phân vùng (partitionVersion, PV) là một số M bit, trong đó N ($N < M$) bit đầu tiên là chỉ số đánh dấu sự thay đổi của cụm và (M-N) bit còn lại là chỉ số đồng bộ đánh dấu cho mỗi chu kì đồng bộ tại chỉ số N bit, giá trị N thay đổi khi cấu hình cụm thay đổi như quá trình thêm nút, mất nút hoặc khi trạng thái cụm thay đổi khi thay đổi nút dẫn đầu (leader), chỉ số đồng bộ được tăng dần sau mỗi chu kì, khi chỉ số đánh dấu cụm N bit thay đổi thì chỉ số đồng bộ được thiết lập về 0, nếu (M-N) bit bị tràn số thì tăng giá trị N bit và chỉ số đồng bộ cũng thực hiện thiết lập lại về 0;

thực hiện lưu trữ dữ liệu đối với từng nhóm sao chép theo dạng: `<synPartitionVersion <map<partitionVersion, map<metadata>>>>`, trong đó synPartitionVersion là giá trị của phiên bản của các phân vùng đã được đồng bộ tại thời điểm gần nhất với thời điểm hiện tại, partitionVersion là giá trị phiên bản của các phân vùng tại thời điểm dữ liệu được ghi, metadata (siêu dữ liệu) chứa thông tin bảng, khóa và giá trị phiên bản dữ liệu ;

khi cấu hình của cụm thay đổi, dẫn đến cấu hình của các nhóm sao chép trong cùng một phân vùng có thể bị thay đổi, nút mới trong cấu hình cụm sao chép sẽ yêu cầu chuyển đổi dữ liệu đến nút cũ thuộc trong cả cấu hình cũ và cấu hình mới kèm theo phiên bản của các phân vùng đã được đồng bộ gần nhất (Synchronize Partition Version, SPV) của nút, nút nhận được yêu cầu đồng bộ thực hiện chọn ra những siêu dữ liệu có giá trị phiên bản của các phân vùng lớn hơn phiên bản của các phân vùng được đồng bộ gần nhất, thực hiện lấy dữ liệu từ nơi lưu trữ dữ liệu ứng với siêu dữ liệu tìm được để gửi đến nút cần chuyển đổi dữ liệu, nút cần chuyển đổi dữ liệu sau khi nhận dữ liệu thực hiện cập nhật vào trong nơi lưu trữ dữ liệu;

khi cụm ổn định, dữ liệu sẽ được đồng bộ định kì với chu kì được cài đặt, nút dẫn đầu sẽ là nút phát động quá trình đồng bộ tới tất cả các nút trong cụm, nút khi nhận được yêu cầu thực hiện đồng bộ, phân vùng tương ứng trong mỗi nút sẽ thực hiện xem xét nút là nút chính hay là nút sao chép của phân vùng đó, trong đó:

nút sao chép thực hiện gửi siêu dữ liệu cần xem xét đồng bộ đến nút chính;

nút chính sau khi nhận được toàn bộ siêu dữ liệu từ các nút sao chép và của chính nó, nút chính thực hiện so sánh các siêu dữ liệu để tìm ra dữ liệu mới nhất, nút chứa dữ liệu mới nhất tương ứng và nút cần cập nhật dữ liệu tương ứng đó;

nút chính thực hiện gửi bản tin đẩy dữ liệu (pushData) tới các nút sao chép kèm theo dữ liệu mà nút chính là nút chứa dữ liệu mới nhất mà nút sao chép cần cập nhật và thông tin cập danh sách siêu dữ liệu và nút cần cập nhật các dữ liệu mới nhất tương ứng mà nút sao chép chứa đựng;

nút sao chép sau khi nhận được bản tin pushData, nếu có dữ liệu cập nhật thì nút sao chép thực hiện cập nhật dữ liệu mới nhất vào trong nơi lưu trữ dữ liệu; nếu có thông tin cập danh sách siêu dữ liệu và nút, nút sao chép thực hiện lấy dữ liệu từ nơi lưu trữ dữ liệu với siêu dữ liệu tương ứng và gửi đến nút tương ứng thông qua bản tin gửi dữ liệu (sendData);

nút nhận được bản tin sendData thực hiện cập nhật dữ liệu mới nhất vào trong nơi lưu trữ dữ liệu, sau đó thực hiện phản hồi thông báo quá trình gửi bản tin sendData thành công;

khi tất cả các nút thuộc phân vùng hoàn thành nhận bản tin pushData, nút chính thực hiện cập nhật phiên bản của các phân vùng đồng bộ $synPartitionVersion = PV$;

trong lúc đồng bộ, những dữ liệu được ghi sẽ được đồng bộ ở chu kì tiếp theo;

nếu trong một chu kì dữ liệu chưa được đồng bộ xong sẽ được đồng bộ lại ở chu kì tiếp theo.

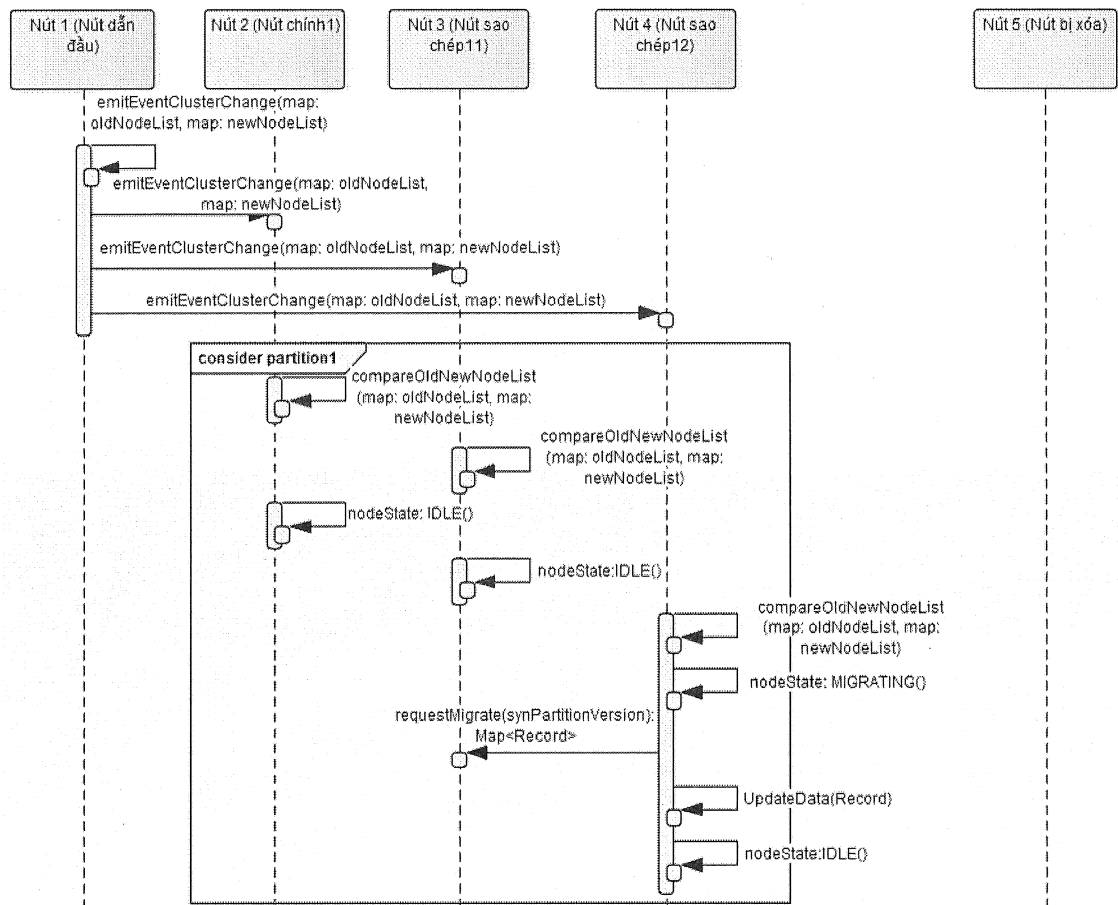
2. Phương pháp chuyển đổi dữ liệu và đồng bộ dữ liệu định kì dựa vào sự sai khác theo điểm 1, trong đó tại một thời điểm, hoặc xảy ra quá trình đồng bộ, hoặc xảy ra quá trình chuyển đổi dữ liệu như sau:

nếu đang trong quá trình đồng bộ, cấu hình nút trong cụm bị thay đổi thì sẽ ưu tiên quá trình chuyển đổi dữ liệu với partitionVersion lớn hơn do N bit đầu tiên tăng;

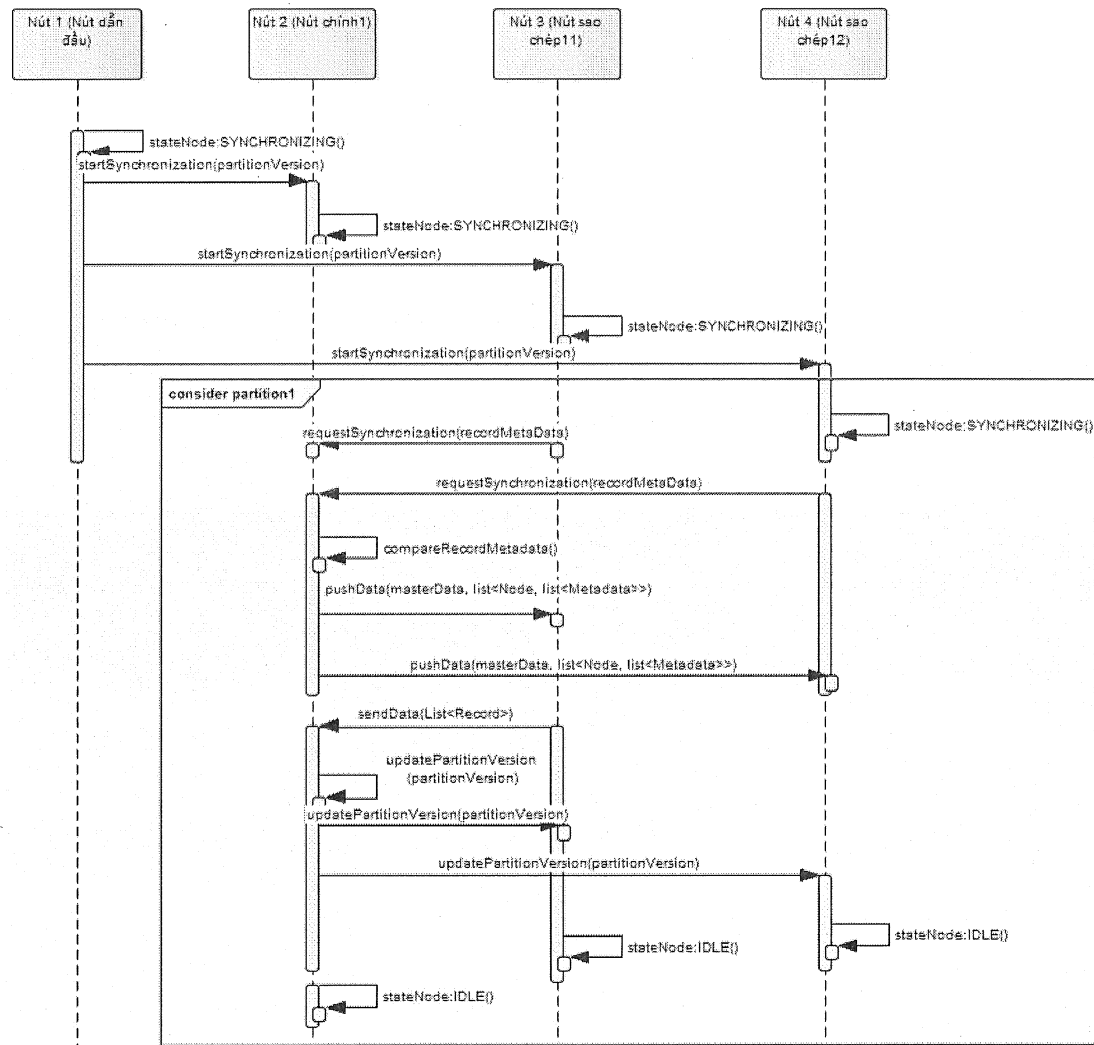
nếu đang trong quá trình đồng bộ, nút nhận được yêu cầu đồng bộ với chu kì lớn hơn thì thực hiện đóng chu kì đồng bộ cũ và bắt đầu thực hiện chu kì đồng bộ mới;

nếu nút đang trong quá trình chuyển đổi dữ liệu thì sẽ không chấp nhận bất kì yêu cầu đồng bộ nào.

3. Phương pháp chuyển đổi dữ liệu và đồng bộ dữ liệu định kì dựa vào sự sai khác theo điểm 1 được thực hiện trong hệ thống quản lý dữ liệu phân tán đa vi xử lý khi cấu hình cụm thay đổi hoặc khi đường truyền mạng lỗi đảm bảo tính nhất quán cuối cùng của dữ liệu.



Hình 1



Hình 2