



(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)



1-0034798

(51)⁸ G06F 12/02

(13) B

(21) 1-2018-03336

(22) 22/12/2016

(86) PCT/CN2016/111557 22/12/2016

(87) WO 2017/114288 A1 06/07/2017

(30) 201511032537.7 31/12/2015 CN

(45) 25/01/2023 418

(43) 25/10/2018 367A

(73) HUAWEI TECHNOLOGIES CO., LTD. (CN)

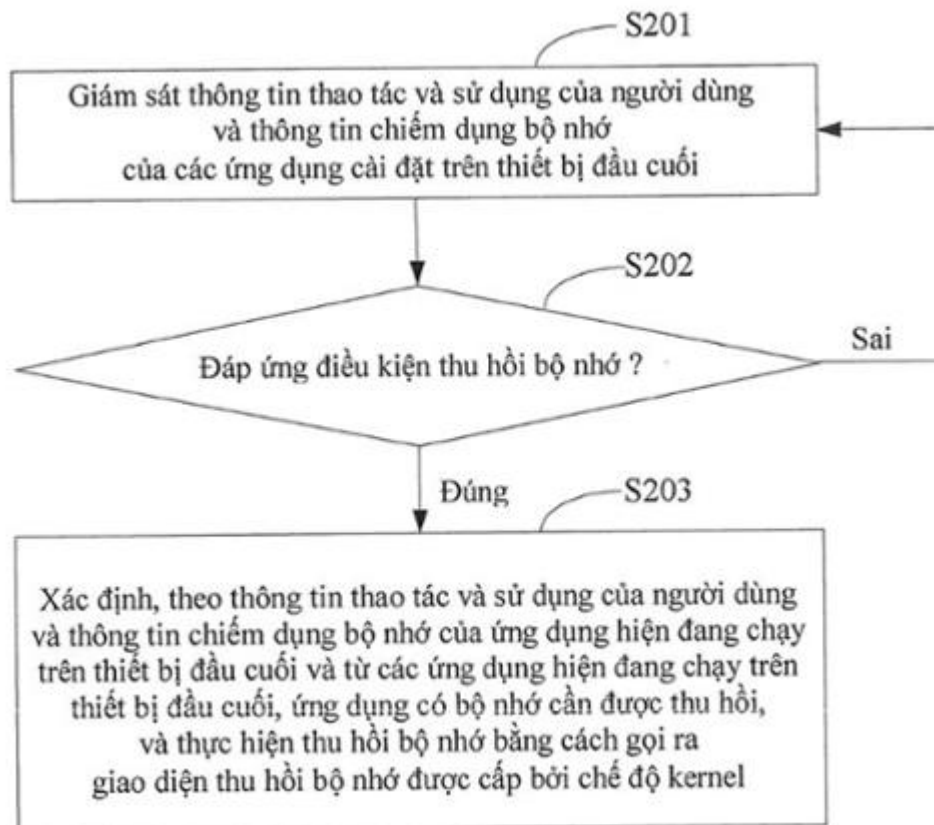
Huawei Administration Building, Bantian, Longgang District, Shenzhen, Guangdong
518129, China

(72) WANG, Bintian (CN); QIU, Xishi (CN).

(74) Công ty TNHH một thành viên Sở hữu trí tuệ VCCI (VCCI-IP CO.,LTD)

(54) PHƯƠNG PHÁP THU HỒI BỘ NHỚ, HỆ THỐNG MÁY TÍNH VÀ PHƯƠNG
TIỆN LƯU TRỮ ĐƯỢC ĐỌC ĐƯỢC BỞI MÁY TÍNH

(57) Sáng chế đề cập đến lĩnh vực kỹ thuật máy tính, và cụ thể, đề cập đến phương pháp thu hồi bộ nhớ, hệ thống máy tính và phương tiện lưu trữ đọc được bởi máy tính, để giải quyết vấn đề ngưng chạy ứng dụng dễ dàng xảy ra do không thực hiện kịp thời việc thu hồi bộ nhớ. Phương pháp thu hồi bộ nhớ theo sáng chế bao gồm các bước: giám sát thông tin thao tác và sử dụng của người dùng và thông tin chiếm dụng bộ nhớ của các ứng dụng cài đặt trong thiết bị đầu cuối; và nếu xác định, theo ít nhất một trong số thông tin thao tác và sử dụng của người dùng hoặc thông tin chiếm dụng bộ nhớ của các ứng dụng cài đặt trong thiết bị đầu cuối, rằng điều kiện thu hồi bộ nhớ hiện được đáp ứng hiện thời, thì xác định, theo thông tin thao tác và sử dụng của người dùng và thông tin chiếm dụng bộ nhớ của các ứng dụng hiện đang chạy trên thiết bị đầu cuối và từ các ứng dụng hiện đang chạy trên thiết bị đầu cuối, ứng dụng có bộ nhớ cần được thu hồi, và tiến hành thu hồi bộ nhớ bằng cách gọi ra giao diện thu hồi bộ nhớ được cấp bởi chế độ lõi (kernel).



Lĩnh vực kỹ thuật được đề cập

Sáng chế đề cập đến lĩnh vực kỹ thuật máy tính, và cụ thể là phương pháp và thiết bị thu hồi bộ nhớ.

Tình trạng kỹ thuật của sáng chế

Xét đến chi phí sản xuất, dung lượng bộ nhớ của thiết bị đầu cuối thường tương đối nhỏ, và trong hầu hết các trường hợp, từ 1 đến 2 G. Hiện tại, kernel đáp ứng, càng nhiều càng tốt, yêu cầu của chương trình ứng dụng để xin bộ nhớ. Kết quả là, trên sản phẩm đầu cuối như máy điện thoại di động, ứng dụng (như Camera) tiêu thụ bộ nhớ tương đối lớn sẽ ngưng chạy, hoặc thậm chí ứng dụng được thoát ra hoặc khởi động lại nhiều lần trong quá trình khởi động hoặc chạy thường là vì không xin được bộ nhớ đủ (ví dụ, ứng dụng có thể xin bộ nhớ từ 300 MB đến 500 MB tức thời).

Quy tắc thu hồi bộ nhớ của hệ thống Linux là kích hoạt thu hồi bộ nhớ khi bộ nhớ không đủ. Ví dụ, khi ứng dụng được khởi động hoặc chuyển sang giao diện nền trước, quy trình ứng dụng ở chế độ người dùng áp dụng vào chế độ lõi (kernel) để xin bộ nhớ, và việc thu hồi bộ nhớ được kích hoạt nếu bộ nhớ mà ứng dụng cần chiếm dụng lớn hơn bộ nhớ hệ thống trống. Cụ thể, bộ nhớ thường ít sử dụng gần đây nhất được chọn, bằng cách dùng thuật toán ít sử dụng gần đây nhất (LRU - *Least Recently Used*), để thu hồi. Ngoài ra, việc thu hồi bộ nhớ cũng có thể được thực hiện bằng cách sử dụng kỹ thuật nén trong bộ nhớ, hợp nhất cùng một trang, chống phân mảnh bộ nhớ, và tương tự. Hệ thống Android (Android) kích hoạt thu hồi bộ nhớ ở chế độ lõi (kernel) bằng cách thiết đặt mức bộ nhớ trống ở chế độ lõi (kernel). Ví dụ, nếu bộ nhớ nhỏ hơn n_1 MB, thì quy trình ứng dụng có quyền ưu tiên thấp nhất được hủy bỏ, và nếu bộ nhớ vẫn nhỏ hơn n_2 ($n_1 < n_2$) MB, thì quy trình ứng dụng có quyền ưu tiên cao hơn một mức được giải phóng.

Theo cả hai quy tắc thu hồi bộ nhớ nêu trên, việc thu hồi bộ nhớ chỉ được

kích hoạt khi nhận thấy, ở chế độ lõi (kernel), rằng bộ nhớ trống có thể được sử dụng là tương đối nhỏ. Do đó, điều sau đây có thể xảy ra: Bộ nhớ có tỷ lệ sử dụng tương đối thấp trong hệ thống không thể được giải phóng kịp thời. Khi việc thu hồi bộ nhớ được kích hoạt sau khi nhận thấy rằng bộ nhớ trống không đủ, điều sau đây có thể xảy ra: Ứng dụng hiện đang chạy không xin được bộ nhớ cần thiết kịp thời sẽ ngưng chạy.

Bản chất kỹ thuật của sáng chế

Sáng chế đề xuất phương pháp và thiết bị thu hồi bộ nhớ, để giải quyết, ở mức độ nhất định, vấn đề ngưng chạy ứng dụng dễ dàng xảy ra do việc thu hồi bộ nhớ không được thực hiện kịp thời.

Sáng chế đề xuất phương pháp thu hồi bộ nhớ, trong đó các bước sau được thực hiện ở chế độ người dùng: giám sát thông tin thao tác và sử dụng của người dùng và thông tin chiếm dụng bộ nhớ của các ứng dụng cài đặt trong thiết bị đầu cuối; và nếu xác định, theo ít nhất một trong số thông tin thao tác và sử dụng của người dùng hoặc thông tin chiếm dụng bộ nhớ của các ứng dụng cài đặt trong thiết bị đầu cuối, rằng điều kiện thu hồi bộ nhớ hiện được đáp ứng hiện thời, thì xác định, theo thông tin thao tác và sử dụng của người dùng và thông tin chiếm dụng bộ nhớ của các ứng dụng hiện đang chạy trên thiết bị đầu cuối và từ các ứng dụng hiện đang chạy trên thiết bị đầu cuối, ứng dụng có bộ nhớ cần được thu hồi, và tiến hành thu hồi bộ nhớ bằng cách gọi ra giao diện thu hồi bộ nhớ được cấp bởi chế độ lõi (kernel).

Theo sáng chế, việc thu hồi bộ nhớ được chủ động thực hiện bằng cách giám sát, ở chế độ người dùng, trạng thái thao tác và sử dụng của người dùng và trạng thái chiếm dụng bộ nhớ của các ứng dụng cài đặt trong thiết bị đầu cuối, để khi một ứng dụng được người dùng thao tác và sử dụng thường xuyên và chiếm dụng bộ nhớ tương đối lớn, tần suất thu hồi bộ nhớ có thể tăng, nhằm đảm bảo sử dụng bộ nhớ bình thường.

Theo một phương án thực hiện, việc xác định, theo thông tin thao tác và sử dụng của người dùng và thông tin chiếm dụng bộ nhớ của các ứng dụng cài đặt

trong thiết bị đầu cuối, rằng điều kiện thu hồi bộ nhớ hiện được đáp ứng hiện thời bao gồm: xác định thời khoảng thu hồi bộ nhớ theo thông tin thao tác và sử dụng của người dùng và thông tin chiếm dụng bộ nhớ của các ứng dụng cài đặt trong thiết bị đầu cuối; và xác định rằng điều kiện thu hồi bộ nhớ hiện được đáp ứng hiện thời khi thời gian hiện thời đã đến thời gian thu hồi bộ nhớ được chỉ ra bởi thời khoảng thu hồi bộ nhớ.

Theo phương án thực hiện này, trước tiên thời khoảng thu hồi bộ nhớ được xác định theo mức độ thường xuyên sử dụng ứng dụng của người dùng, và tiếp đó việc thu hồi bộ nhớ được chủ động thực hiện theo thời gian thu hồi bộ nhớ được chỉ ra bởi thời khoảng thu hồi bộ nhớ. Theo cách này, việc thu hồi theo kế hoạch được thực hiện trên bộ nhớ, để cho bộ nhớ trông khả dụng trong thiết bị đầu cuối được duy trì ở mức tương đối mong muốn.

Theo phương án thực hiện nêu trên, cách tối ưu để xác định thời khoảng thu hồi bộ nhớ là: xác định riêng thông tin thao tác và sử dụng của người dùng và thông tin chiếm dụng bộ nhớ, trong mỗi phân đoạn thời gian trong số ít nhất một phân đoạn thời gian thiết đặt trước, của các ứng dụng cài đặt trong thiết bị đầu cuối; và đối với mỗi phân đoạn thời gian trong số ít nhất một phân đoạn thời gian thiết đặt trước, xác định riêng thời khoảng thu hồi bộ nhớ trong phân đoạn thời gian theo thông tin thao tác và sử dụng của người dùng và thông tin chiếm dụng bộ nhớ, trong phân đoạn thời gian này, của các ứng dụng cài đặt trong thiết bị đầu cuối.

Tần suất thao tác và sử dụng ứng dụng của người dùng là khác nhau trong phân đoạn thời gian khác nhau. Do đó, theo cách này, các thời khoảng thu hồi bộ nhớ khác nhau có thể được sử dụng trong các phân đoạn thời gian khác nhau. Điều này nâng cao tính kịp thời thu hồi bộ nhớ, và cũng tránh lãng phí tài nguyên hệ thống do thường xuyên thực hiện thủ tục thu hồi bộ nhớ không cần thiết.

Theo phương án thực hiện khác, việc xác định, theo thông tin chiếm dụng bộ nhớ của các ứng dụng cài đặt trong thiết bị đầu cuối, rằng điều kiện thu hồi bộ nhớ hiện được đáp ứng hiện thời bao gồm: nếu xác định rằng bộ nhớ hệ

thống trống nhỏ hơn ngưỡng quy định thứ nhất, thì xác định rằng điều kiện thu hồi bộ nhớ hiện được đáp ứng hiện thời.

Theo cách này, việc thiếu bộ nhớ có thể tránh được ở mức độ nhất định.

Theo phương án thực hiện khác nữa, việc xác định, theo thông tin thao tác và sử dụng của người dùng và thông tin chiếm dụng bộ nhớ của các ứng dụng cài đặt trong thiết bị đầu cuối, rằng điều kiện thu hồi bộ nhớ hiện được đáp ứng hiện thời bao gồm: nếu xác định là ứng dụng được khởi động hoặc chuyển sang giao diện nền trước, thì xác định, theo thông tin chiếm dụng bộ nhớ của ứng dụng và kích cỡ của bộ nhớ hệ thống trống, xem việc thu hồi bộ nhớ có cần được thực hiện hay không; và nếu việc thu hồi bộ nhớ cần được thực hiện, thì xác định rằng điều kiện thu hồi bộ nhớ hiện được đáp ứng hiện thời.

Ứng dụng mới khởi động hoặc ứng dụng được chuyển sang giao diện nền trước thường là ứng dụng mà người dùng sắp sử dụng, và theo cách này, có thể đảm bảo chạy bình thường ứng dụng mới khởi động hoặc ứng dụng được chuyển sang giao diện nền trước, nhờ đó nâng cao hiệu suất hệ thống và cải thiện trải nghiệm của người dùng.

Theo phương án thực hiện khác nữa, việc xác định, theo thông tin thao tác và sử dụng của người dùng của các ứng dụng cài đặt trong thiết bị đầu cuối, rằng điều kiện thu hồi bộ nhớ hiện được đáp ứng hiện thời bao gồm: nếu số lượng ứng dụng hiện đang chạy trên nền sau lớn hơn số lượng ứng dụng quy định được phép chạy trên nền sau, thì xác định rằng điều kiện thu hồi bộ nhớ hiện được đáp ứng hiện thời.

Ngoài ra, ở đây, số lượng ứng dụng được phép chạy trên nền sau có thể được thiết đặt tại một thời khoảng theo kích cỡ của bộ nhớ trống hiện thời. Vì các ứng dụng đang chạy trên nền sau thường không phải là các ứng dụng được người dùng sử dụng hiện thời, nên có thể tránh được việc chiếm dụng bộ nhớ không cần thiết quá mức bằng cách giới hạn số lượng các ứng dụng này.

Theo phương án thực hiện nêu trên, cách tối ưu để xác định một ứng dụng có bộ nhớ cần được thu hồi, và tiến hành thu hồi bộ nhớ bằng cách gọi ra giao

diện thu hồi bộ nhớ được cấp bởi chế độ lõi (kernel) là: chọn, theo thứ tự ưu tiên đã lưu trữ trước của các ứng dụng có bộ nhớ cần được thu hồi, ứng dụng hiện đang chạy trên thiết bị đầu cuối và không được người dùng thao tác hoặc sử dụng, và xác định xem thời gian hiện thời có nằm trong phân đoạn thời gian sử dụng xác suất cao của ứng dụng đã chọn hay không, trong đó phân đoạn thời gian sử dụng xác suất cao của ứng dụng được xác định dựa vào thông tin thao tác và sử dụng của người dùng của ứng dụng; nếu thời gian hiện thời không nằm trong phân đoạn thời gian sử dụng xác suất cao của ứng dụng đã chọn, thì xác định rằng ứng dụng đã chọn là ứng dụng có bộ nhớ cần được thu hồi, và tiến hành thu hồi bộ nhớ bằng cách gọi ra giao diện thu hồi bộ nhớ được cấp bởi chế độ lõi (kernel); và lặp lại các bước nêu trên cho đến khi đáp ứng điều kiện kết thúc thu hồi bộ nhớ.

Theo cách này, bộ nhớ của một ứng dụng (ứng dụng này thường là ứng dụng không được sử dụng thường xuyên bởi người dùng) có quyền ưu tiên tương đối thấp được ưu tiên chọn để thu hồi, trong đó thời gian hiện thời không nằm trong phân đoạn thời gian chạy xác suất cao của ứng dụng, nhờ đó nâng cao hiệu suất hệ thống và cải thiện trải nghiệm của người dùng.

Theo phương án thực hiện nêu trên, điều kiện kết thúc thu hồi bộ nhớ có thể bao gồm một hoặc nhiều điều kiện sau:

kích cỡ của bộ nhớ hệ thống trống lớn hơn ngưỡng quy định thứ hai;

bộ nhớ cần có cho ứng dụng được khởi động hiện thời nhỏ hơn bộ nhớ hệ thống trống;

bộ nhớ cần có cho ứng dụng được chuyển sang giao diện nền trước nhỏ hơn bộ nhớ hệ thống trống; và

số lượng ứng dụng đang chạy trên nền sau nhỏ hơn hoặc bằng số lượng ứng dụng quy định được phép chạy trên nền sau.

Ngoài ra, thứ tự ưu tiên của các ứng dụng có bộ nhớ cần được thu hồi có thể được thiết đặt theo các bước sau:

tạo nhóm các ứng dụng được cài đặt trong thiết bị đầu cuối theo tần suất sử

dụng của các ứng dụng, trong đó các ứng dụng trong cùng một nhóm có cùng một dải tần suất sử dụng; và xác định thứ tự ưu tiên thu hồi bộ nhớ của các ứng dụng trong mỗi nhóm theo dải tần suất sử dụng tương ứng với mỗi nhóm, trong đó các ứng dụng trong cùng một nhóm có cùng một thứ tự ưu tiên.

Theo một phương án thực hiện khác nữa, nếu xác định, theo thông tin thao tác và sử dụng của người dùng của các ứng dụng cài đặt trong thiết bị đầu cuối, rằng điều kiện thu hồi bộ nhớ hiện được đáp ứng hiện thời, thì bước tiến hành thu hồi bộ nhớ bằng cách gọi ra giao diện thu hồi bộ nhớ được cấp bởi chế độ lõi (kernel) bao gồm: nếu nhận thấy thời lượng lưu trú, của ứng dụng trong thiết bị đầu cuối, trong nền sau vượt quá thời lượng quy định, thì xác định rằng điều kiện thu hồi bộ nhớ hiện được đáp ứng hiện thời, và thu hồi, bằng cách gọi ra giao diện thu hồi bộ nhớ được cấp bởi chế độ lõi (kernel), bộ nhớ của ứng dụng có thời lượng lưu trú trong nền sau vượt quá thời lượng quy định.

Ở đây, ứng dụng có thời lượng lưu trú trong nền sau tương đối dài thường là ứng dụng không được người dùng thao tác trong khoảng thời gian hiện tại, và bộ nhớ của ứng dụng này có thể được giải phóng và do đó được cấp cho ứng dụng cần bộ nhớ lớn hơn, nhờ đó nâng cao tỷ lệ sử dụng hiệu quả của bộ nhớ.

Sáng chế còn đề cập đến thiết bị thu hồi bộ nhớ, bao gồm:

môđun xác định, được tạo cấu hình để giám sát, ở chế độ người dùng, thông tin thao tác và sử dụng của người dùng và thông tin chiếm dụng bộ nhớ của các ứng dụng cài đặt trong thiết bị đầu cuối; và

môđun thu hồi bộ nhớ, được tạo cấu hình để: nếu xác định, theo ít nhất một trong số thông tin thao tác và sử dụng của người dùng hoặc thông tin chiếm dụng bộ nhớ của các ứng dụng cài đặt trong thiết bị đầu cuối, rằng điều kiện thu hồi bộ nhớ hiện được đáp ứng hiện thời, thì xác định, theo thông tin thao tác và sử dụng của người dùng và thông tin chiếm dụng bộ nhớ của các ứng dụng hiện đang chạy trên thiết bị đầu cuối và từ các ứng dụng hiện đang chạy trên thiết bị đầu cuối, ứng dụng có bộ nhớ cần được thu hồi, và tiến hành thu hồi bộ nhớ bằng cách gọi ra giao diện thu hồi bộ nhớ được cấp bởi chế độ lõi (kernel).

Theo sáng chế, việc thu hồi bộ nhớ được chủ động thực hiện bằng cách giám sát, ở chế độ người dùng, trạng thái hoạt động và sử dụng của người dùng và trạng thái chiếm dụng bộ nhớ của các ứng dụng cài đặt trong thiết bị đầu cuối, để khi một ứng dụng được người dùng thao tác và sử dụng thường xuyên và chiếm dụng bộ nhớ tương đối lớn, tần suất thu hồi bộ nhớ có thể tăng, nhằm đảm bảo việc sử dụng bộ nhớ bình thường.

Theo một phương án thực hiện, môđun thu hồi bộ nhớ được tạo cấu hình cụ thể để:

xác định thời khoảng thu hồi bộ nhớ theo thông tin thao tác và sử dụng của người dùng và thông tin chiếm dụng bộ nhớ của các ứng dụng cài đặt trong thiết bị đầu cuối; và xác định rằng điều kiện thu hồi bộ nhớ hiện được đáp ứng hiện thời khi thời gian hiện thời đã đến thời gian thu hồi bộ nhớ được chỉ ra bởi thời khoảng thu hồi bộ nhớ.

Theo phương án thực hiện này, trước tiên thời khoảng thu hồi bộ nhớ được xác định theo mức độ thường xuyên sử dụng ứng dụng của người dùng, và tiếp đó việc thu hồi bộ nhớ được chủ động thực hiện theo thời gian thu hồi bộ nhớ được chỉ ra bởi thời khoảng thu hồi bộ nhớ. Theo cách này, việc thu hồi theo kế hoạch được thực hiện trên bộ nhớ, để cho bộ nhớ trống khả dụng trong thiết bị đầu cuối được duy trì ở mức tương đối mong muốn.

Theo phương án thực hiện nêu trên, cách tối ưu để xác định thời khoảng thu hồi bộ nhớ bởi môđun thu hồi bộ nhớ là: xác định riêng thông tin thao tác và sử dụng của người dùng và thông tin chiếm dụng bộ nhớ, trong mỗi phân đoạn thời gian trong số ít nhất một phân đoạn thời gian thiết đặt trước, của các ứng dụng cài đặt trong thiết bị đầu cuối; và đối với mỗi phân đoạn thời gian trong số ít nhất một phân đoạn thời gian thiết đặt trước, xác định riêng thời khoảng thu hồi bộ nhớ trong phân đoạn thời gian theo thông tin thao tác và sử dụng của người dùng và thông tin chiếm dụng bộ nhớ, trong phân đoạn thời gian này, của các ứng dụng cài đặt trong thiết bị đầu cuối.

Tần suất thao tác và sử dụng ứng dụng của người dùng là khác nhau trong

phân đoạn thời gian khác nhau. Do đó, theo cách này, các thời khoảng thu hồi bộ nhớ khác nhau có thể được sử dụng trong các phân đoạn thời gian khác nhau. Điều này nâng cao tính kịp thời thu hồi bộ nhớ, và cũng tránh lãng phí tài nguyên hệ thống do thường xuyên thực hiện thủ tục thu hồi bộ nhớ không cần thiết.

Theo phương án thực hiện khác, môđun thu hồi bộ nhớ được tạo cấu hình cụ thể để nếu xác định rằng bộ nhớ hệ thống trống nhỏ hơn ngưỡng quy định thứ nhất, thì xác định rằng điều kiện thu hồi bộ nhớ hiện được đáp ứng hiện thời.

Theo cách này, việc thiếu bộ nhớ có thể tránh được.

Theo phương án thực hiện khác nữa, môđun thu hồi bộ nhớ được tạo cấu hình cụ thể để: nếu xác định là ứng dụng được khởi động hoặc chuyển sang giao diện nền trước, thì xác định, theo thông tin chiếm dụng bộ nhớ của ứng dụng và kích cỡ của bộ nhớ hệ thống trống, xem việc thu hồi bộ nhớ có cần được thực hiện hay không; và nếu việc thu hồi bộ nhớ cần được thực hiện, thì xác định rằng điều kiện thu hồi bộ nhớ hiện được đáp ứng hiện thời.

Ứng dụng mới khởi động hoặc ứng dụng được chuyển sang giao diện nền trước thường là ứng dụng mà người dùng sắp sử dụng, và theo cách này, việc chạy bình thường ứng dụng mới khởi động hoặc ứng dụng chuyển sang giao diện nền trước có thể được đảm bảo, nhờ đó nâng cao hiệu suất hệ thống, và còn cải thiện trải nghiệm của người dùng.

Theo phương án thực hiện khác nữa, môđun thu hồi bộ nhớ được tạo cấu hình cụ thể để nếu số lượng ứng dụng hiện đang chạy trên nền sau lớn hơn số lượng ứng dụng quy định được phép chạy trên nền sau, thì xác định rằng điều kiện thu hồi bộ nhớ hiện được đáp ứng hiện thời.

Ngoài ra, ở đây, số lượng ứng dụng được phép chạy trên nền sau có thể được thiết đặt tại một thời khoảng theo kích cỡ của bộ nhớ trống hiện thời. Vì các ứng dụng đang chạy trên nền sau thường không phải là các ứng dụng được người dùng sử dụng hiện thời, nên có thể tránh được việc chiếm dụng bộ nhớ không cần thiết quá mức bằng cách giới hạn số lượng các ứng dụng này.

Theo phương án thực hiện nêu trên, cách tối ưu để xác định, bởi môđun thu hồi bộ nhớ, ứng dụng có bộ nhớ cần được thu hồi, và tiến hành thu hồi bộ nhớ bằng cách gọi ra giao diện thu hồi bộ nhớ được cấp bởi chế độ lõi (kernel) là: chọn, theo thứ tự ưu tiên đã lưu trữ trước của các ứng dụng có bộ nhớ cần được thu hồi, ứng dụng hiện đang chạy trên thiết bị đầu cuối và không được người dùng thao tác hoặc sử dụng, và xác định xem thời gian hiện thời có nằm trong phân đoạn thời gian sử dụng xác suất cao của ứng dụng đã chọn hay không, trong đó phân đoạn thời gian sử dụng xác suất cao của ứng dụng được xác định dựa vào thông tin thao tác và sử dụng của người dùng của ứng dụng; nếu thời gian hiện thời không nằm trong phân đoạn thời gian sử dụng xác suất cao của ứng dụng đã chọn, thì xác định rằng ứng dụng đã chọn là ứng dụng có bộ nhớ cần được thu hồi, và tiến hành thu hồi bộ nhớ bằng cách gọi ra giao diện thu hồi bộ nhớ được cấp bởi chế độ lõi (kernel); và lặp lại các bước nêu trên cho đến khi đáp ứng điều kiện kết thúc thu hồi bộ nhớ.

Theo cách này, bộ nhớ của một ứng dụng (ứng dụng này thường là ứng dụng không được người dùng sử dụng thường xuyên) có quyền ưu tiên tương đối thấp được ưu tiên chọn để thu hồi, trong đó thời gian hiện thời không nằm trong phân đoạn thời gian chạy xác suất cao của ứng dụng, nhờ đó nâng cao hiệu suất hệ thống và cải thiện trải nghiệm của người dùng.

Theo phương án thực hiện nêu trên, điều kiện kết thúc thu hồi bộ nhớ có thể bao gồm một hoặc nhiều điều kiện sau:

kích cỡ của bộ nhớ hệ thống trống lớn hơn ngưỡng quy định thứ hai;

bộ nhớ cần có cho ứng dụng được khởi động hiện thời là nhỏ hơn bộ nhớ hệ thống trống;

bộ nhớ cần có cho ứng dụng được chuyển sang giao diện nền trước là nhỏ hơn bộ nhớ hệ thống trống; và

số lượng ứng dụng đang chạy trên nền sau nhỏ hơn hoặc bằng số lượng ứng dụng quy định được phép chạy trên nền sau.

Ngoài ra, môđun thu hồi bộ nhớ có thể được tạo cấu hình cụ thể để thiết đặt

thứ tự ưu tiên của các ứng dụng có bộ nhớ cần được thu hồi theo các bước sau: tạo nhóm các ứng dụng được cài đặt trong thiết bị đầu cuối theo tần suất sử dụng của các ứng dụng, trong đó các ứng dụng trong cùng một nhóm có cùng một dải tần suất sử dụng; và xác định thứ tự ưu tiên thu hồi bộ nhớ của các ứng dụng trong mỗi nhóm theo dải tần suất sử dụng tương ứng với mỗi nhóm, trong đó các ứng dụng trong cùng một nhóm có cùng một thứ tự ưu tiên.

Theo một phương án thực hiện khác nữa, môđun thu hồi bộ nhớ được tạo cấu hình cụ thể để: nếu nhận thấy thời lượng lưu trú, của ứng dụng trong thiết bị đầu cuối, trong nền sau vượt quá thời lượng quy định, xác định rằng điều kiện thu hồi bộ nhớ hiện được đáp ứng hiện thời, và thu hồi, bằng cách gọi ra giao diện thu hồi bộ nhớ được cấp bởi chế độ lõi (kernel), bộ nhớ của ứng dụng có thời lượng lưu trú trong nền sau vượt quá thời lượng quy định.

Sáng chế đề cập đến thiết bị người dùng, trong đó thiết bị người dùng bao gồm bộ xử lý, bộ nhớ và kênh truyền. Bộ nhớ được tạo cấu hình để lưu trữ lệnh thực thi, và bao gồm thiết bị nhớ trong và bộ nhớ ngoài. Thiết bị nhớ trong ở đây còn được gọi là bộ nhớ trong, và được tạo cấu hình để lưu trữ tạm thời dữ liệu hoạt động của bộ xử lý, và dữ liệu được trao đổi với bộ nhớ ngoài như đĩa cứng. Bộ xử lý trao đổi dữ liệu với bộ nhớ ngoài bằng cách sử dụng thiết bị nhớ trong, và khi thiết bị người dùng chạy, bộ xử lý và bộ nhớ truyền thông với nhau bằng cách sử dụng kênh truyền, để cho bộ xử lý thi hành các lệnh sau ở chế độ người dùng:

giám sát thông tin thao tác và sử dụng của người dùng và thông tin chiếm dụng bộ nhớ của các ứng dụng cài đặt trong thiết bị đầu cuối; và khi xác định, theo ít nhất một trong số thông tin thao tác và sử dụng của người dùng hoặc thông tin chiếm dụng bộ nhớ của các ứng dụng cài đặt trong thiết bị đầu cuối, rằng điều kiện thu hồi bộ nhớ hiện được đáp ứng hiện thời, thì xác định, theo thông tin thao tác và sử dụng của người dùng và thông tin chiếm dụng bộ nhớ của các ứng dụng hiện đang chạy trên thiết bị đầu cuối và từ các ứng dụng hiện đang chạy trên thiết bị đầu cuối, ứng dụng có bộ nhớ cần được thu hồi, và tiến hành thu hồi bộ nhớ bằng cách gọi ra giao diện thu hồi bộ nhớ được cấp bởi chế

độ lõi (kernel).

Theo sáng chế, việc thu hồi bộ nhớ được chủ động thực hiện bằng cách giám sát, ở chế độ người dùng, trạng thái hoạt động và sử dụng của người dùng và trạng thái chiếm dụng bộ nhớ của các ứng dụng cài đặt trong thiết bị đầu cuối, để khi một ứng dụng được người dùng thao tác và sử dụng thường xuyên và chiếm dụng bộ nhớ tương đối lớn, tần suất thu hồi bộ nhớ có thể tăng, nhằm đảm bảo việc sử dụng bộ nhớ bình thường.

Theo một phương án thực hiện, trong các lệnh được thực thi bởi bộ xử lý, việc xác định, theo thông tin thao tác và sử dụng của người dùng và thông tin chiếm dụng bộ nhớ của các ứng dụng cài đặt trong thiết bị đầu cuối, rằng điều kiện thu hồi bộ nhớ hiện được đáp ứng hiện thời bao gồm: xác định thời khoảng thu hồi bộ nhớ theo thông tin thao tác và sử dụng của người dùng và thông tin chiếm dụng bộ nhớ của các ứng dụng cài đặt trong thiết bị đầu cuối; và xác định rằng điều kiện thu hồi bộ nhớ hiện được đáp ứng hiện thời khi thời gian hiện thời đã đến thời gian thu hồi bộ nhớ được chỉ ra bởi thời khoảng thu hồi bộ nhớ.

Theo phương án thực hiện này, trước tiên thời khoảng thu hồi bộ nhớ được xác định theo mức độ thường xuyên sử dụng ứng dụng của người dùng, và tiếp đó việc thu hồi bộ nhớ được chủ động thực hiện theo thời gian thu hồi bộ nhớ được chỉ ra bởi thời khoảng thu hồi bộ nhớ. Theo cách này, việc thu hồi theo kế hoạch được thực hiện trên bộ nhớ, để cho bộ nhớ trống khả dụng trong thiết bị đầu cuối được duy trì ở mức tương đối mong muốn.

Theo phương án thực hiện nêu trên, trong các lệnh được thực thi bởi bộ xử lý, cách tối ưu để xác định thời khoảng thu hồi bộ nhớ bởi bộ xử lý là:

xác định riêng thông tin thao tác và sử dụng của người dùng và thông tin chiếm dụng bộ nhớ, trong mỗi phân đoạn thời gian trong số ít nhất một phân đoạn thời gian thiết đặt trước, của các ứng dụng cài đặt trong thiết bị đầu cuối; và đối với mỗi phân đoạn thời gian trong số ít nhất một phân đoạn thời gian thiết đặt trước, xác định riêng thời khoảng thu hồi bộ nhớ trong phân đoạn thời gian theo thông tin thao tác và sử dụng của người dùng và thông tin chiếm dụng bộ

nhớ, trong phân đoạn thời gian này, của các ứng dụng cài đặt trong thiết bị đầu cuối.

Tần suất thao tác và sử dụng ứng dụng của người dùng là khác nhau trong phân đoạn thời gian khác nhau. Do đó, theo cách này, các thời khoảng thu hồi bộ nhớ khác nhau có thể được sử dụng trong các phân đoạn thời gian khác nhau. Điều này nâng cao tính kịp thời thu hồi bộ nhớ, và cũng tránh lãng phí tài nguyên hệ thống do thường xuyên thực hiện thủ tục thu hồi bộ nhớ không cần thiết.

Theo phương án thực hiện khác, trong các lệnh được thực thi bởi bộ xử lý, việc xác định, theo thông tin chiếm dụng bộ nhớ của các ứng dụng cài đặt trong thiết bị đầu cuối, rằng điều kiện thu hồi bộ nhớ hiện được đáp ứng hiện thời bao gồm: nếu xác định rằng bộ nhớ hệ thống trống nhỏ hơn ngưỡng quy định thứ nhất, thì xác định rằng điều kiện thu hồi bộ nhớ hiện được đáp ứng hiện thời.

Theo cách này, việc thiếu bộ nhớ có thể tránh được.

Theo phương án thực hiện khác nữa, trong các lệnh được thực thi bởi bộ xử lý, việc xác định, theo thông tin thao tác và sử dụng của người dùng và thông tin chiếm dụng bộ nhớ của các ứng dụng cài đặt trong thiết bị đầu cuối, rằng điều kiện thu hồi bộ nhớ hiện được đáp ứng hiện thời bao gồm: nếu xác định là ứng dụng được khởi động hoặc chuyển sang giao diện nền trước, thì xác định, theo thông tin chiếm dụng bộ nhớ của ứng dụng và kích cỡ của bộ nhớ hệ thống trống, xem việc thu hồi bộ nhớ có cần được thực hiện hay không; và nếu việc thu hồi bộ nhớ cần được thực hiện, thì xác định rằng điều kiện thu hồi bộ nhớ hiện được đáp ứng hiện thời.

Ứng dụng mới khởi động hoặc ứng dụng được chuyển sang giao diện nền trước thường là ứng dụng mà người dùng sắp sử dụng, và theo cách này, việc chạy bình thường ứng dụng mới khởi động hoặc ứng dụng chuyển sang giao diện nền trước có thể được đảm bảo, nhờ đó nâng cao hiệu suất hệ thống, và còn cải thiện trải nghiệm của người dùng.

Theo phương án thực hiện khác nữa, trong các lệnh được thực thi bởi bộ xử

lý, việc xác định, theo thông tin thao tác và sử dụng của người dùng của các ứng dụng cài đặt trong thiết bị đầu cuối, rằng điều kiện thu hồi bộ nhớ hiện được đáp ứng hiện thời bao gồm: nếu số lượng ứng dụng hiện đang chạy trên nền sau lớn hơn số lượng ứng dụng quy định được phép chạy trên nền sau, thì xác định rằng điều kiện thu hồi bộ nhớ hiện được đáp ứng hiện thời.

Ngoài ra, ở đây, số lượng ứng dụng được phép chạy trên nền sau có thể được thiết đặt tại một thời khoảng theo kích cỡ của bộ nhớ trống hiện thời. Vì các ứng dụng đang chạy trên nền sau thường không phải là các ứng dụng được người dùng sử dụng hiện thời, nên có thể tránh được việc chiếm dụng bộ nhớ không cần thiết quá mức bằng cách giới hạn số lượng các ứng dụng này.

Theo phương án thực hiện nêu trên, trong các lệnh được thực thi bởi bộ xử lý, cách tối ưu để xác định ứng dụng có bộ nhớ cần được thu hồi, và tiến hành thu hồi bộ nhớ bằng cách gọi ra giao diện thu hồi bộ nhớ được cấp bởi chế độ lõi (kernel) là: chọn, theo thứ tự ưu tiên đã lưu trữ trước của các ứng dụng có bộ nhớ cần được thu hồi, ứng dụng hiện đang chạy trên thiết bị đầu cuối và không được người dùng thao tác hoặc sử dụng, và xác định xem thời gian hiện thời có nằm trong phân đoạn thời gian sử dụng xác suất cao của ứng dụng đã chọn hay không, trong đó phân đoạn thời gian sử dụng xác suất cao của ứng dụng được xác định dựa vào thông tin thao tác và sử dụng của người dùng của ứng dụng; nếu thời gian hiện thời không nằm trong phân đoạn thời gian sử dụng xác suất cao của ứng dụng đã chọn, thì xác định rằng ứng dụng đã chọn là ứng dụng có bộ nhớ cần được thu hồi, và tiến hành thu hồi bộ nhớ bằng cách gọi ra giao diện thu hồi bộ nhớ được cấp bởi chế độ lõi (kernel); và lặp lại các bước nêu trên cho đến khi đáp ứng điều kiện kết thúc thu hồi bộ nhớ.

Theo cách này, bộ nhớ của một ứng dụng (ứng dụng này thường là ứng dụng không được người dùng sử dụng thường xuyên) có quyền ưu tiên tương đối thấp được ưu tiên chọn để thu hồi, trong đó thời gian hiện thời không nằm trong phân đoạn thời gian chạy xác suất cao của ứng dụng, nhờ đó nâng cao hiệu suất hệ thống và cải thiện trải nghiệm của người dùng.

Theo phương án thực hiện nêu trên, điều kiện kết thúc thu hồi bộ nhớ có

thể bao gồm một hoặc nhiều điều kiện sau:

kích cỡ của bộ nhớ hệ thống trống lớn hơn ngưỡng quy định thứ hai;

bộ nhớ cần có cho ứng dụng được khởi động hiện thời là nhỏ hơn bộ nhớ hệ thống trống;

bộ nhớ cần có cho ứng dụng được chuyển sang giao diện nền trước là nhỏ hơn bộ nhớ hệ thống trống; và

số lượng ứng dụng đang chạy trên nền sau nhỏ hơn hoặc bằng số lượng ứng dụng quy định được phép chạy trên nền sau.

Ngoài ra, trong các lệnh được thực thi bởi bộ xử lý, cách tối ưu để thiết đặt thứ tự ưu tiên của các ứng dụng có bộ nhớ cần được thu hồi là:

tạo nhóm các ứng dụng được cài đặt trong thiết bị đầu cuối theo tần suất sử dụng của các ứng dụng, trong đó các ứng dụng trong cùng một nhóm có cùng một dải tần suất sử dụng; và xác định thứ tự ưu tiên thu hồi bộ nhớ của các ứng dụng trong mỗi nhóm theo dải tần suất sử dụng tương ứng với mỗi nhóm, trong đó các ứng dụng trong cùng một nhóm có cùng một thứ tự ưu tiên.

Theo một phương án thực hiện khác nữa, trong các lệnh được thực thi bởi bộ xử lý, nếu xác định, theo thông tin thao tác và sử dụng của người dùng của các ứng dụng cài đặt trong thiết bị đầu cuối, rằng điều kiện thu hồi bộ nhớ hiện được đáp ứng hiện thời, thì việc thực hiện thu hồi bộ nhớ bằng cách gọi ra giao diện thu hồi bộ nhớ được cấp bởi chế độ lõi (kernel) bao gồm: nếu nhận thấy thời lượng lưu trú, của ứng dụng trong thiết bị đầu cuối, trong nền sau vượt quá thời lượng quy định, thì xác định rằng điều kiện thu hồi bộ nhớ hiện được đáp ứng hiện thời, và thu hồi, bằng cách gọi ra giao diện thu hồi bộ nhớ được cấp bởi chế độ lõi (kernel), bộ nhớ của ứng dụng có thời lượng lưu trú trong nền sau vượt quá thời lượng quy định.

Ở đây, ứng dụng có thời lượng lưu trú trong nền sau tương đối dài thường là ứng dụng không được người dùng thao tác trong khoảng thời gian hiện tại, và bộ nhớ của ứng dụng này có thể được giải phóng và do đó được cấp cho ứng dụng cần bộ nhớ lớn hơn, nhờ đó nâng cao tỷ lệ sử dụng hiệu quả của bộ nhớ.

Mô tả vắn tắt các hình vẽ

Fig.1 là sơ đồ thu hồi bộ nhớ theo sáng chế;

Fig.2 là lưu đồ của phương pháp thu hồi bộ nhớ theo phương án 1 của sáng chế;

Fig.3 là lưu đồ của phương pháp thu hồi bộ nhớ theo phương án 2 của sáng chế;

Fig.4 là lưu đồ của phương pháp thu hồi bộ nhớ theo phương án 3 của sáng chế;

Fig.5 là sơ đồ cấu trúc của thiết bị thu hồi bộ nhớ 50 theo một phương án của sáng chế;

Fig.6 là sơ đồ cấu trúc của thiết bị người dùng 60 theo một phương án của sáng chế;

Fig.7 là sơ đồ cấu trúc của thiết bị đầu cuối 700 theo sáng chế; và

Fig.8 là sơ đồ cấu trúc của thiết bị đầu cuối 800 theo một phương án của sáng chế.

Mô tả chi tiết sáng chế

Như được thể hiện trên Fig.1, Fig.1 là sơ đồ thu hồi bộ nhớ theo sáng chế. Theo sáng chế, việc thu hồi bộ nhớ được thực hiện bằng cách chạy quy tắc thu hồi bộ nhớ ở chế độ người dùng và sử dụng giao diện thu hồi bộ nhớ được cấp bởi chế độ lõi (kernel). Cụ thể, quy trình thu hồi bộ nhớ được kích hoạt bằng cách giám sát thông tin thao tác và sử dụng của người dùng và thông tin chiếm dụng bộ nhớ của các ứng dụng (App - *Application*) cài đặt trên thiết bị đầu cuối. Sau khi điều kiện thu hồi bộ nhớ được đáp ứng, ứng dụng có bộ nhớ cần được thu hồi được chọn theo thông tin thao tác và sử dụng của người dùng và thông tin chiếm dụng bộ nhớ của các ứng dụng hiện đang chạy trên thiết bị đầu cuối và từ các ứng dụng hiện đang chạy trên thiết bị đầu cuối, và việc thu hồi bộ nhớ được thực hiện bằng cách gọi ra giao diện thu hồi bộ nhớ được cấp bởi chế độ lõi (kernel) cho chế độ người dùng. Theo sáng chế, việc thu hồi bộ nhớ được chủ động thực hiện bằng cách giám sát, ở chế độ người dùng, thông tin thao tác và

sử dụng của người dùng và thông tin chiếm dụng bộ nhớ của các ứng dụng, để khi một ứng dụng được người dùng thao tác và sử dụng thường xuyên và chiếm dụng bộ nhớ tương đối lớn, tần suất thu hồi bộ nhớ có thể tăng, nhằm đảm bảo kịp thời thu hồi bộ nhớ.

Phần dưới đây mô tả chi tiết các phương án của sáng chế có dựa vào các hình vẽ kèm theo của bản mô tả.

Phương án 1

Như được thể hiện trên Fig.2, Fig.2 là lưu đồ của phương pháp thu hồi bộ nhớ theo phương án 1 của sáng chế, và phương pháp này thực hiện các bước sau ở chế độ người dùng.

S201. Giám sát thông tin thao tác và sử dụng của người dùng và thông tin chiếm dụng bộ nhớ của các ứng dụng cài đặt trong thiết bị đầu cuối.

Ở đây, thông tin chiếm dụng bộ nhớ có thể bao gồm kích cỡ của bộ nhớ được chiếm dụng khi ứng dụng chạy, và thông tin thao tác và sử dụng của người dùng của các ứng dụng có thể bao gồm tần suất sử dụng, phân đoạn thời gian sử dụng, thời lượng sử dụng và thông tin tương tự về ứng dụng của người dùng.

Theo một phương án thực hiện cụ thể, trạng thái chạy của mỗi ứng dụng trong hệ thống đầu cuối và kích cỡ bộ nhớ được chiếm dụng khi ứng dụng chạy có thể được giám sát theo thời gian thực bằng cách sử dụng quy trình giám sát ở chế độ người dùng. Tần suất sử dụng, phân đoạn thời gian sử dụng, thời lượng sử dụng và thông tin tương tự về ứng dụng của người dùng có thể được xác định nhờ sử dụng trạng thái chạy của ứng dụng có được thông qua giám sát. Ở đây, phân đoạn thời gian sử dụng ứng dụng của người dùng có thể là một phân đoạn thời gian, trong đó người dùng sử dụng ứng dụng, trong nhiều trong nhiều phân đoạn thời gian thiết đặt trước, hoặc có thể là phân đoạn thời gian thực từ thời điểm người dùng bắt đầu thao tác ứng dụng đến thời điểm người dùng hoàn tất thao tác ứng dụng.

Ở đây, việc giám sát trạng thái chạy của mỗi ứng dụng trên hệ thống đầu cuối có thể bao gồm: giám sát trạng thái khởi động ứng dụng, trạng thái đóng

ứng dụng, trạng thái chuyển ứng dụng từ nền sau sang giao diện nền trước, trạng thái chuyển từ giao diện nền trước sang nền sau, và tương tự. Quy trình giám sát có thể thu nhận cụ thể, bằng cách truy vấn, trạng thái chạy trước đó của ứng dụng bằng cách sử dụng giao diện truy vấn được cấp bởi hệ thống tập tin proc.

Theo một phương án thực hiện cụ thể, đối với các ứng dụng khác nhau, các cách thu thập số liệu thống kê về thông tin thao tác và sử dụng của người dùng của các ứng dụng có thể là khác nhau. Ví dụ, đối với ứng dụng cần được thao tác trên giao diện nền trước bởi người dùng, như WeChat hoặc Calendar, thời gian mà tại đó ứng dụng được khởi động hoặc chuyển sang giao diện nền trước có thể được dùng làm thời gian mà tại đó người dùng bắt đầu thao tác ứng dụng, và thời gian mà tại đó ứng dụng được đóng hoặc chuyển sang nền sau để chạy có thể được dùng làm thời gian mà tại đó người dùng hoàn tất thao tác ứng dụng. Đối với ứng dụng không cần người dùng luôn thao tác trên giao diện nền trước, như phần mềm tải xuống, thời gian mà tại đó ứng dụng được khởi động có thể được dùng làm thời gian mà tại đó người dùng bắt đầu thao tác ứng dụng, và thời gian mà tại đó ứng dụng được đóng có thể được dùng làm thời gian mà tại đó người dùng hoàn tất thao tác ứng dụng. Tần suất sử dụng, phân đoạn thời gian sử dụng, thời lượng sử dụng, và thông tin tương tự về việc sử dụng ứng dụng của người dùng có thể được xác định theo thời gian mà tại đó người dùng bắt đầu thao tác ứng dụng và thời gian mà tại đó người dùng hoàn tất thao tác ứng dụng có được bằng cách thu thập số liệu thống kê.

S202. Xác định, theo ít nhất một trong số thông tin thao tác và sử dụng của người dùng hoặc thông tin chiếm dụng bộ nhớ của các ứng dụng cài đặt trong thiết bị đầu cuối, xem điều kiện thu hồi bộ nhớ hiện thời có được đáp ứng hay không; và chuyển sang bước S203 nếu điều kiện thu hồi bộ nhớ hiện được đáp ứng hiện thời, hoặc quay trở lại bước S201 nếu điều kiện thu hồi bộ nhớ hiện thời không được đáp ứng.

Theo một phương án thực hiện cụ thể, điều kiện để kích hoạt thu hồi bộ nhớ ở chế độ người dùng có thể được thiết đặt theo yêu cầu thực tế. Ví dụ, thời khoảng thu hồi bộ nhớ có thể được xác định theo thông tin thao tác và sử dụng

của người dùng và thông tin chiếm dụng bộ nhớ của các ứng dụng cài đặt trong thiết bị đầu cuối, và quy trình thu hồi bộ nhớ được kích hoạt theo thời khoảng thu hồi bộ nhớ thiết đặt trước, để tiến hành thu hồi bộ nhớ theo kế hoạch, để cho bộ nhớ trống khả dụng trong thiết bị đầu cuối được duy trì ở mức tương đối mong muốn. Ở đây, có thể có một hoặc nhiều thời khoảng thu hồi bộ nhớ thiết đặt trước. Ví dụ, nếu cả tần suất sử dụng ứng dụng của người dùng trong cả ngày và việc chiếm dụng bộ nhớ đều tương đối ổn định, thì chỉ một thời khoảng thu hồi bộ nhớ có thể được thiết đặt. Theo cách khác, nếu người dùng chỉ tập trung sử dụng ứng dụng trong một phân đoạn thời gian trong cả ngày, và tần suất sử dụng ứng dụng tương đối ổn định, thì thời khoảng thu hồi bộ nhớ có thể chỉ được thiết đặt trong phân đoạn thời gian này, và việc thu hồi bộ nhớ không được thực hiện trong phân đoạn thời gian khác. Nếu tần suất sử dụng ứng dụng của người dùng thay đổi nhiều trong các phân đoạn thời gian khác nhau trong ngày, thì các thời khoảng thu hồi bộ nhớ khác nhau có thể được sử dụng trong các phân đoạn thời gian khác nhau trong ngày. Ngoài ra, có thể thiết đặt để: việc thu hồi bộ nhớ được kích hoạt khi bộ nhớ hệ thống trống nhỏ hơn ngưỡng (thu hồi bộ nhớ được kích hoạt theo thông tin chiếm dụng bộ nhớ của các ứng dụng cài đặt trong thiết bị đầu cuối); khi nhận thấy ứng dụng được khởi động hoặc chuyển sang giao diện nền trước, việc thu hồi bộ nhớ được kích hoạt theo thông tin chiếm dụng bộ nhớ của ứng dụng và kích cỡ của bộ nhớ hệ thống trống (thu hồi bộ nhớ được kích hoạt theo thông tin thao tác và sử dụng của người dùng và thông tin chiếm dụng bộ nhớ của các ứng dụng cài đặt trong thiết bị đầu cuối); hoặc việc thu hồi bộ nhớ được kích hoạt khi nhận thấy thời lượng lưu trú, của ứng dụng trong thiết bị đầu cuối, trong nền sau vượt quá thời lượng quy định (thu hồi bộ nhớ được kích hoạt theo thông tin thao tác và sử dụng của người dùng của các ứng dụng).

S203. Xác định, theo thông tin thao tác và sử dụng của người dùng và thông tin chiếm dụng bộ nhớ của các ứng dụng hiện đang chạy trên thiết bị đầu cuối và từ các ứng dụng hiện đang chạy trên thiết bị đầu cuối, ứng dụng có bộ nhớ cần được thu hồi, và tiến hành thu hồi bộ nhớ bằng cách gọi ra giao diện thu

hồi bộ nhớ được cấp bởi chế độ lõi (kernel).

Ở đây, các ứng dụng được cài đặt trên thiết bị đầu cuối ở các bước S201 và S202 bao gồm các ứng dụng hiện đang chạy trên thiết bị đầu cuối ở bước S203. Ngoài các ứng dụng hiện đang chạy trên thiết bị đầu cuối ở bước S203, các ứng dụng được cài đặt trên thiết bị đầu cuối còn gồm cả ứng dụng hiện đang ở trạng thái đóng.

Theo một phương án thực hiện cụ thể, chương trình thu hồi bộ nhớ ở chế độ lõi (kernel) được gọi ra ở chế độ người dùng bằng cách sử dụng giao diện thu hồi bộ nhớ được cấp bởi chế độ lõi (kernel), để tiến hành thu hồi bộ nhớ. Giao diện thu hồi bộ nhớ có thể được thi hành dựa vào hệ thống như hệ thống tập tin linux procfs hoặc hệ thống tập tin linux sysfs.

Trong hệ thống linux, việc quản lý thu hồi bộ nhớ hiện có ở chế độ lõi (kernel) được thi hành dựa vào thuật toán LRU. Theo kỹ thuật quản lý thu hồi bộ nhớ hiện có, bộ nhớ được chiếm dụng bởi ứng dụng được chia thành hai loại chính: trang ẩn danh (như bộ nhớ được cấp phát bằng cách sử dụng hàm cấp phát bộ nhớ động (malloc)) và trang tập tin (như bộ nhớ được cấp phát trong quá trình đọc và ghi tập tin). Bộ nhớ gần đây không được sử dụng thường xuyên được đưa vào danh mục liên kết theo thuật toán LRU, và danh mục liên kết được thu nhỏ mỗi khi việc thu hồi được thực hiện, để thu hồi một phần các trang bộ nhớ (chủ yếu là các bộ nhớ đệm trang (page cache)). Hàm cơ bản dùng cho việc thu hồi bộ nhớ ở chế độ lõi (kernel) là `try_to_free_pages()`. Theo sáng chế, giao diện của hàm này được hiển thị cho chế độ người dùng để gọi ra. Ở chế độ người dùng, việc thu hồi bộ nhớ được thực hiện bằng cách gọi ra giao diện của hàm này và dựa vào quy tắc thu hồi bộ nhớ được cấp theo sáng chế.

Phương án 2

Phần dưới đây mô tả một phương án thực hiện ưu tiên của sáng chế: xác định thời khoảng thu hồi bộ nhớ bằng cách thu thập số liệu thống kê về thông tin thao tác và sử dụng của người dùng và thông tin chiếm dụng bộ nhớ của các ứng dụng cài đặt trong thiết bị đầu cuối, và tiến hành thu hồi bộ nhớ dựa vào thời

khoảng thu hồi bộ nhớ đã được xác định.

Như được thể hiện trên Fig.3, Fig.3 là lưu đồ của phương pháp thu hồi bộ nhớ theo phương án 2 của sáng chế, phương pháp này thực hiện các bước sau ở chế độ người dùng:

S301. Giám sát thông tin thao tác và sử dụng của người dùng và thông tin chiếm dụng bộ nhớ của các ứng dụng cài đặt trong thiết bị đầu cuối.

Theo một phương án thực hiện cụ thể, số liệu thống kê về thông tin thao tác và sử dụng của người dùng của ứng dụng, như tần suất sử dụng và phân đoạn thời gian sử dụng đối với việc sử dụng ứng dụng của người dùng, và thông tin chiếm dụng bộ nhớ như kích cỡ của bộ nhớ được chiếm dụng khi ứng dụng chạy được thu thập bằng cách giám sát trạng thái chạy của mỗi ứng dụng trong hệ thống đầu cuối ở chế độ người dùng theo thời gian thực.

S302. Xác định thời khoảng thu hồi bộ nhớ theo thông tin thao tác và sử dụng của người dùng và thông tin chiếm dụng bộ nhớ của các ứng dụng cài đặt trong thiết bị đầu cuối.

Ở đây, thời khoảng thu hồi bộ nhớ có thể được xác định theo thông tin thao tác và sử dụng của người dùng và thông tin chiếm dụng bộ nhớ của các ứng dụng có được bằng cách thu thập số liệu thống kê ở bước S301. Có thể có một hoặc nhiều thời khoảng thu hồi bộ nhớ. Ví dụ, nếu cả tần suất sử dụng ứng dụng của người dùng trong cả ngày và thông tin chiếm dụng bộ nhớ đều tương đối nhất quán, thì chỉ một thời khoảng thu hồi bộ nhớ có thể được thiết đặt. Nếu tần suất sử dụng ứng dụng của người dùng là khác nhau trong các phân đoạn thời gian khác nhau trong ngày, thì các thời khoảng thu hồi bộ nhớ khác nhau có thể được sử dụng trong các phân đoạn thời gian khác nhau trong ngày. Ví dụ, việc thu hồi bộ nhớ có thể được thực hiện thường xuyên hơn trong phân đoạn thời gian mà trong đó ứng dụng được sử dụng thường xuyên và bộ nhớ tương đối lớn được chiếm dụng. Ngược lại, việc thu hồi bộ nhớ có thể được thực hiện ít thường xuyên hơn trong phân đoạn thời gian mà trong đó ứng dụng được sử dụng không thường xuyên và bộ nhớ tương đối nhỏ được chiếm dụng.

Theo một phương án thực hiện cụ thể, nhiều phân đoạn thời gian có thể được thiết đặt trước theo kinh nghiệm. Thông tin thao tác và sử dụng của người dùng và thông tin chiếm dụng bộ nhớ, trong mỗi phân đoạn thời gian thiết đặt trước, của các ứng dụng cài đặt trong thiết bị đầu cuối được xác định. Thời khoảng thu hồi bộ nhớ trong mỗi phân đoạn thời gian thiết đặt trước được xác định theo thông tin thao tác và sử dụng của người dùng và thông tin chiếm dụng bộ nhớ, trong phân đoạn thời gian thiết đặt trước, của các ứng dụng cài đặt trong thiết bị đầu cuối.

Ví dụ, ngày được dùng làm độ chi tiết, và một vài phân đoạn thời gian 9:00–17:00, 17:00–23:00, và 23:00–9:00 được thiết đặt trước. Số liệu thống kê về thông tin thao tác và sử dụng của người dùng và thông tin chiếm dụng bộ nhớ, trong một vài phân đoạn thời gian thiết đặt trước, của ứng dụng được thu thập riêng rẽ. Ví dụ, trong phân đoạn thời gian 17:00–23:00, người dùng sử dụng các ứng dụng thường xuyên nhất, kích cỡ bộ nhớ được chiếm dụng bởi các ứng dụng cũng là lớn nhất, và thời khoảng thu hồi bộ nhớ trong phân đoạn thời gian này có thể được thiết đặt là 0,5 giờ. Trong phân đoạn thời gian 9:00–17:00, tần suất sử dụng các ứng dụng bởi người dùng là mức trung gian, kích cỡ bộ nhớ được chiếm dụng cũng là mức trung gian, và thời khoảng thu hồi bộ nhớ trong phân đoạn thời gian này có thể được thiết đặt là 1 giờ. Trong phân đoạn thời gian 23:00–9:00, người dùng sử dụng các ứng dụng ít thường xuyên nhất, kích cỡ bộ nhớ được chiếm dụng cũng là nhỏ nhất, và thời khoảng thu hồi bộ nhớ trong phân đoạn thời gian này có thể được thiết đặt là 3 giờ. Ví dụ khác, sự khác biệt giữa các trạng thái sử dụng trong ngày làm việc và ngày nghỉ trong tuần có thể được xem xét, để thiết đặt tương ứng các thời khoảng thu hồi bộ nhớ khác nhau cho ngày làm việc và ngày nghỉ. Ví dụ, tần suất sử dụng ứng dụng của người dùng vào ngày thứ bảy và chủ nhật rõ ràng là cao hơn tần suất sử dụng ứng dụng trong ngày làm việc. Các thời khoảng thu hồi bộ nhớ nêu trên có thể vẫn được sử dụng từ ngày thứ hai đến thứ sáu, và tần suất thu hồi bộ nhớ vào ngày thứ bảy và chủ nhật được gia tăng. Ví dụ, các thời khoảng thu hồi bộ nhớ trong các phân đoạn thời gian 9:00–17:00, 17:00–23:00, và 23:00–9:00 nêu trên

được giảm đi một nửa vào ngày thứ bảy và chủ nhật. Tức là, thời khoảng thu hồi bộ nhớ trong phân đoạn thời gian 17:00–23:00 được thiết đặt là 0,25 giờ, thời khoảng thu hồi bộ nhớ trong phân đoạn thời gian 9:00–17:00 được thiết đặt là 0,5 giờ, và thời khoảng thu hồi bộ nhớ trong phân đoạn thời gian 23:00–9:00 được thiết đặt là 1,5 giờ.

S303. Nếu xác định, theo thời khoảng thu hồi bộ nhớ, rằng việc thu hồi bộ nhớ có cần được tiến hành hiện thời, thì xác định, theo thông tin thao tác và sử dụng của người dùng và thông tin chiếm dụng bộ nhớ của các ứng dụng hiện đang chạy trên thiết bị đầu cuối và từ các ứng dụng hiện đang chạy trên thiết bị đầu cuối, ứng dụng có bộ nhớ cần được thu hồi. Ở đây, để đảm bảo rằng người dùng sử dụng bình thường một ứng dụng, việc thu hồi bộ nhớ không được xem xét đối với ứng dụng hiện đang được sử dụng bởi người dùng. ứng dụng hiện đang được sử dụng bởi người dùng có thể bao gồm ứng dụng nền trước, và còn có thể bao gồm ứng dụng chạy trên nền sau, như tải xuống tập tin.

S304. Tiến hành thu hồi bộ nhớ bằng cách gọi ra giao diện thu hồi bộ nhớ được cấp bởi chế độ lõi (kernel).

Theo một phương án thực hiện cụ thể, khi xác định rằng đã đến thời điểm thu hồi bộ nhớ theo thời khoảng thu hồi bộ nhớ đã được xác định, bộ nhớ của một ứng dụng hiện đang chạy trên thiết bị đầu cuối và không được thao tác hoặc sử dụng bởi người dùng có thể được thu hồi nếu thời gian hiện thời không nằm trong khoảng thời gian, nhận được bằng cách thu thập số liệu thống kê, trong đó xác suất sử dụng ứng dụng của người dùng là tương đối cao. Theo cách này, các ứng dụng được chọn thành công cho việc thu hồi bộ nhớ cho đến khi đáp ứng điều kiện kết thúc thu hồi bộ nhớ.

Theo một phương án thực hiện, sau khi xác định rằng việc thu hồi bộ nhớ cần được tiến hành hiện thời, ứng dụng hiện đang chạy trên thiết bị đầu cuối và không được thao tác hoặc sử dụng bởi người dùng có thể được chọn theo thứ tự ưu tiên đã lưu trữ trước của các ứng dụng có bộ nhớ cần được thu hồi. Thời gian hiện thời có nằm trong phân đoạn thời gian sử dụng xác suất cao (tức là, phân đoạn thời gian trong đó xác suất thao tác và sử dụng của người dùng đối với ứng

dụng là tương đối cao) của ứng dụng đã chọn hay không được xác định. Phân đoạn thời gian sử dụng xác suất cao của ứng dụng được xác định dựa vào thông tin thao tác và sử dụng của người dùng của ứng dụng (ví dụ, nếu người dùng sử dụng một ứng dụng từ 12:00 đến 13:00 trong năm ngày liên tiếp, thì khoảng từ 12:00 đến 13:00 là phân đoạn thời gian chạy xác suất cao của ứng dụng). Nếu thời gian hiện thời không nằm trong phân đoạn thời gian sử dụng xác suất cao của ứng dụng đã chọn, thì xác định rằng ứng dụng đã chọn là ứng dụng có bộ nhớ cần được thu hồi. Các bước trên được lặp lại cho đến khi đáp ứng điều kiện kết thúc thu hồi bộ nhớ. Ở đây, điều kiện kết thúc thu hồi bộ nhớ có thể bao gồm điều kiện kích cỡ của bộ nhớ hệ thống trống lớn hơn ngưỡng quy định thứ hai.

Thứ tự ưu tiên có thể là thứ tự ưu tiên giữa các nhóm ứng dụng khác nhau. Cụ thể, các ứng dụng được cài đặt trên thiết bị đầu cuối có thể được tạo nhóm theo tần suất sử dụng của các ứng dụng, trong đó các ứng dụng trong cùng một nhóm có cùng một dải tần suất sử dụng. Thứ tự ưu tiên thu hồi bộ nhớ của các ứng dụng trong mỗi nhóm được xác định theo dải tần suất sử dụng tương ứng với mỗi nhóm, trong đó các ứng dụng trong cùng một nhóm có cùng một thứ tự ưu tiên. Ví dụ, một số ứng dụng, như Giao diện Người dùng (UI - *User Interface*), có thể liên quan đến phần tử chủ chốt của hệ điều hành trong thiết bị đầu cuối, loại ứng dụng này có thể được phân loại vào nhóm (*group*) 1, biểu thị các ứng dụng rất quan trọng. Đối với một số ứng dụng được dùng thường xuyên bởi người dùng, như WeChat và Camera, loại ứng dụng này có thể được phân loại vào nhóm 2, biểu thị các ứng dụng quan trọng thông thường. Đối với một số ứng dụng không được dùng thường xuyên, như Calculator và Stock, loại ứng dụng này có thể được phân loại vào nhóm 3, biểu thị các ứng dụng không quan trọng. Thứ tự ưu tiên được thiết đặt như sau: thứ tự ưu tiên thu hồi bộ nhớ của nhóm 3 > thứ tự ưu tiên thu hồi bộ nhớ của nhóm 2 > thứ tự ưu tiên thu hồi bộ nhớ của nhóm 1. Trong quá trình thu hồi bộ nhớ, việc thu hồi bộ nhớ của các ứng dụng trong nhóm 3 được coi là ưu tiên nhất, thu hồi bộ nhớ của các ứng dụng trong nhóm 2 được coi là ưu tiên thứ hai, và bộ nhớ của các ứng dụng trong nhóm 1 không được thu hồi.

Phương án 3

Để tối ưu hơn nữa quy tắc thu hồi bộ nhớ nhằm đảm bảo thu hồi bộ nhớ kịp thời và phù hợp, một phương án của sáng chế còn đề xuất cách thực hiện để tiến hành thu hồi bộ nhớ bằng cách sử dụng nhiều điều kiện thu hồi bộ nhớ.

Như được thể hiện trên Fig.4, Fig.4 là lưu đồ của phương pháp thu hồi bộ nhớ theo phương án 3 của sáng chế, phương pháp này thực hiện các bước sau ở chế độ người dùng.

S401. Giám sát thông tin thao tác và sử dụng của người dùng và thông tin chiếm dụng bộ nhớ của các ứng dụng cài đặt trong thiết bị đầu cuối.

S402. Xác định thời khoảng thu hồi bộ nhớ theo thông tin thao tác và sử dụng của người dùng và thông tin chiếm dụng bộ nhớ của các ứng dụng cài đặt trong thiết bị đầu cuối.

S403. Kích hoạt thủ tục thu hồi bộ nhớ ở bước S404 khi đáp ứng bất kỳ một hoặc nhiều điều kiện trong số các điều kiện từ 1 đến 4 dưới đây, và kích hoạt thủ tục thu hồi bộ nhớ ở bước S405 khi đáp ứng điều kiện 5.

1. Xác định, theo thời khoảng thu hồi bộ nhớ, rằng việc thu hồi bộ nhớ cần được tiến hành hiện thời.

Theo một phương án thực hiện cụ thể, việc thu hồi bộ nhớ có cần được thực hiện hiện thời hay không cũng có thể được xác định liên quan đến bộ nhớ hệ thống trống. Tức là, khi hiện tại đến thời gian thu hồi bộ nhớ được chỉ ra bởi thời khoảng thu hồi bộ nhớ, việc thu hồi bộ nhớ có thể được thực hiện nếu bộ nhớ hệ thống trống nhỏ hơn ngưỡng (ngưỡng này có thể là ngưỡng quy định thứ nhất, ngưỡng quy định thứ hai hoặc ngưỡng khác được mô tả trong phần dưới đây, trong đó ngưỡng khác có thể là ngưỡng nằm giữa ngưỡng quy định thứ nhất và ngưỡng quy định thứ hai, và ngưỡng quy định thứ nhất nhỏ hơn ngưỡng quy định thứ hai).

2. Nhận thấy rằng bộ nhớ hệ thống trống nhỏ hơn ngưỡng quy định thứ nhất.

3. Khi nhận thấy ứng dụng được khởi động hoặc chuyển sang giao diện nền

trước, xác định, theo thông tin chiếm dụng bộ nhớ của ứng dụng và kích cỡ của bộ nhớ hệ thống trống, rằng việc thu hồi bộ nhớ cần được tiến hành hiện thời.

Ở đây, khi kích cỡ bộ nhớ cần có cho ứng dụng mới được khởi động hoặc chuyển sang giao diện nền trước là lớn hơn bộ nhớ hệ thống trống, thì thủ tục thu hồi bộ nhớ được kích hoạt.

4. Số lượng M ứng dụng đang chạy trên nền sau lớn hơn số lượng quy định hiện thời N ứng dụng được phép chạy trên nền sau.

Theo một phương án thực hiện cụ thể, số lượng ứng dụng được phép chạy trên nền sau có thể được thiết đặt theo kích cỡ của bộ nhớ hệ thống trống hiện thời.

5. Nhận thấy rằng thời lượng lưu trú, của ứng dụng trong thiết bị đầu cuối, trong nền sau vượt quá thời lượng quy định.

Cần lưu ý rằng đối với các điều kiện từ 2 đến 5 nêu trên, người có hiểu biết trung bình trong lĩnh vực kỹ thuật này có thể chỉ chọn một phần các điều kiện này theo yêu cầu thực tế.

S404. Xác định, theo thông tin thao tác và sử dụng của người dùng và thông tin chiếm dụng bộ nhớ của các ứng dụng hiện đang chạy trên thiết bị đầu cuối và từ các ứng dụng hiện đang chạy trên thiết bị đầu cuối và không được thao tác hoặc sử dụng bởi người dùng, ứng dụng có bộ nhớ cần được thu hồi, và tiến hành thu hồi bộ nhớ bằng cách gọi ra giao diện thu hồi bộ nhớ được cấp bởi chế độ lõi (kernel).

Theo một phương án thực hiện cụ thể, sau khi khởi động hệ thống đầu cuối, daemon, được dùng cho việc thu hồi bộ nhớ, ở chế độ người dùng theo sáng chế này được khởi động trước tiên, tiếp đó các ứng dụng được khởi động, và ứng dụng đang chạy bảng dữ liệu được duy trì, để ghi trạng thái chiếm dụng bộ nhớ của các ứng dụng trong thiết bị đầu cuối, và thu thập số liệu thống kê về phân đoạn thời gian chạy xác suất cao, như được thể hiện trong bảng 1 dưới đây.

Bảng 1

Ứng dụng	Bộ nhớ được chiếm dụng	Phân đoạn thời gian chạy xác suất cao
App 1	x1KB	T1–T2
App 2	x2KB	T3–T4
App 3	x3KB	T5–T6
App 4	x4KB	T7–T8
...	...	...

Nếu xác định rằng việc thu hồi bộ nhớ cần được thực hiện, thì ứng dụng hiện đang chạy trên thiết bị đầu cuối có thể được chọn theo thứ tự ưu tiên đã lưu trữ trước của các ứng dụng có bộ nhớ cần được thu hồi. Theo bảng 1, thời gian hiện thời có nằm trong phân đoạn thời gian sử dụng xác suất cao của ứng dụng đã chọn hay không được xác định. Nếu thời gian hiện thời không nằm trong phân đoạn thời gian sử dụng xác suất cao của ứng dụng đã chọn, thì ứng dụng đã chọn được dùng làm ứng dụng có bộ nhớ cần được thu hồi, việc thu hồi bộ nhớ được thực hiện bằng cách gọi ra giao diện thu hồi bộ nhớ dùng cho việc thu hồi bộ nhớ và được cấp bởi chế độ lõi (kernel), và các bước trên được lặp lại cho đến khi đáp ứng điều kiện kết thúc thu hồi bộ nhớ. Ở đây, dựa vào sự khác nhau giữa các điều kiện từ 1 đến 4, điều kiện kết thúc thu hồi bộ nhớ nêu trên có thể bao gồm một hoặc nhiều điều kiện trong số: kích cỡ của bộ nhớ hệ thống trống lớn hơn ngưỡng quy định thứ hai; bộ nhớ cần có cho ứng dụng được khởi động hiện thời là nhỏ hơn bộ nhớ hệ thống trống; bộ nhớ cần có cho ứng dụng được chuyển sang giao diện nền trước là nhỏ hơn bộ nhớ hệ thống trống; hoặc số lượng ứng dụng đang chạy trên nền sau nhỏ hơn hoặc bằng số lượng ứng dụng quy định được phép chạy trên nền sau. Ví dụ, đối với các điều kiện 1 và 2, điều kiện kết thúc thu hồi bộ nhớ có thể là kích cỡ của bộ nhớ hệ thống trống lớn hơn ngưỡng quy định thứ hai (ngưỡng quy định thứ hai lớn hơn ngưỡng quy định thứ nhất); đối với điều kiện 3, điều kiện kết thúc thu hồi bộ nhớ có thể là bộ nhớ cần có cho ứng dụng được khởi động hiện thời nhỏ hơn bộ nhớ hệ thống trống; và

đối với điều kiện 4, điều kiện kết thúc thu hồi bộ nhớ có thể là số lượng ứng dụng đang chạy trên nền sau bằng số lượng ứng dụng quy định được phép chạy trên nền sau, tức là, ở bước S303, M–N ứng dụng có bộ nhớ cần được thu hồi được chọn từ các ứng dụng đang chạy trên nền sau.

S405. Thu hồi, bằng cách gọi ra giao diện thu hồi bộ nhớ được cấp bởi chế độ lõi (kernel), bộ nhớ của ứng dụng có thời lượng lưu trú trong nền sau vượt quá thời lượng quy định.

Dựa vào cùng một khái niệm sáng tạo, sáng chế còn đề xuất thiết bị thu hồi bộ nhớ và thiết bị người dùng tương ứng với phương pháp thu hồi bộ nhớ này. Bởi vì các nguyên lý của thiết bị và thiết bị người dùng để giải quyết các vấn đề là tương tự như các nguyên lý của phương pháp thu hồi bộ nhớ theo sáng chế, để thực hiện thiết bị này và thiết bị người dùng, có thể lấy tham chiếu để thực hiện phương pháp. Các phần lặp lại không được mô tả lại.

Phương án 4

Như được thể hiện trên Fig.5, Fig.5 là sơ đồ cấu trúc của thiết bị thu hồi bộ nhớ 50 theo một phương án của sáng chế, và thiết bị thu hồi bộ nhớ 50 này bao gồm:

môđun xác định 51, được tạo cấu hình để giám sát, ở chế độ người dùng, thông tin thao tác và sử dụng của người dùng và thông tin chiếm dụng bộ nhớ của các ứng dụng cài đặt trong thiết bị đầu cuối; và

môđun thu hồi bộ nhớ 52, được tạo cấu hình để: nếu xác định, theo ít nhất một trong số thông tin thao tác và sử dụng của người dùng hoặc thông tin chiếm dụng bộ nhớ của các ứng dụng cài đặt trong thiết bị đầu cuối, rằng điều kiện thu hồi bộ nhớ hiện được đáp ứng hiện thời, thì xác định, theo thông tin thao tác và sử dụng của người dùng và thông tin chiếm dụng bộ nhớ của các ứng dụng hiện đang chạy trên thiết bị đầu cuối và từ các ứng dụng hiện đang chạy trên thiết bị đầu cuối, ứng dụng có bộ nhớ cần được thu hồi, và tiến hành thu hồi bộ nhớ bằng cách gọi ra giao diện thu hồi bộ nhớ được cấp bởi chế độ lõi (kernel).

Theo sáng chế, việc thu hồi bộ nhớ được chủ động thực hiện bằng cách

giám sát, ở chế độ người dùng, trạng thái hoạt động và sử dụng của người dùng và trạng thái chiếm dụng bộ nhớ của các ứng dụng cài đặt trong thiết bị đầu cuối, để khi một ứng dụng được người dùng thao tác và sử dụng thường xuyên và chiếm dụng bộ nhớ tương đối lớn, tần suất thu hồi bộ nhớ có thể tăng, nhằm đảm bảo việc sử dụng bộ nhớ bình thường.

Một cách tùy chọn, môđun thu hồi bộ nhớ 52 được tạo cấu hình cụ thể để:

xác định thời khoảng thu hồi bộ nhớ theo thông tin thao tác và sử dụng của người dùng và thông tin chiếm dụng bộ nhớ của các ứng dụng cài đặt trong thiết bị đầu cuối; và xác định rằng điều kiện thu hồi bộ nhớ hiện được đáp ứng hiện thời khi thời gian hiện thời đã đến thời gian thu hồi bộ nhớ được chỉ ra bởi thời khoảng thu hồi bộ nhớ.

Theo phương án này, môđun thu hồi bộ nhớ 52 trước tiên xác định thời khoảng thu hồi bộ nhớ theo mức độ thường xuyên sử dụng ứng dụng của người dùng, và tiếp đó chủ động tiến hành việc thu hồi bộ nhớ theo thời gian thu hồi bộ nhớ được chỉ ra bởi thời khoảng thu hồi bộ nhớ. Theo cách này, việc thu hồi theo kế hoạch được thực hiện trên bộ nhớ, để cho bộ nhớ trống khả dụng trong thiết bị đầu cuối được duy trì ở mức tương đối mong muốn.

Một cách tùy chọn, môđun thu hồi bộ nhớ 52 được sử dụng cụ thể để xác định thời khoảng thu hồi bộ nhớ theo các bước sau:

xác định riêng thông tin thao tác và sử dụng của người dùng và thông tin chiếm dụng bộ nhớ, trong mỗi phân đoạn thời gian trong số ít nhất một phân đoạn thời gian thiết đặt trước, của các ứng dụng cài đặt trong thiết bị đầu cuối; và đối với mỗi phân đoạn thời gian trong số ít nhất một phân đoạn thời gian thiết đặt trước, xác định riêng thời khoảng thu hồi bộ nhớ trong phân đoạn thời gian theo thông tin thao tác và sử dụng của người dùng và thông tin chiếm dụng bộ nhớ, trong phân đoạn thời gian này, của các ứng dụng cài đặt trong thiết bị đầu cuối.

Tần suất thao tác và sử dụng ứng dụng của người dùng là khác nhau trong phân đoạn thời gian khác nhau. Do đó, theo cách này, các thời khoảng thu hồi bộ

nhớ khác nhau có thể được sử dụng trong các phân đoạn thời gian khác nhau. Điều này nâng cao tính kịp thời thu hồi bộ nhớ, và cũng tránh lãng phí tài nguyên hệ thống do thường xuyên thực hiện thủ tục thu hồi bộ nhớ không cần thiết.

Một cách tùy chọn, môđun thu hồi bộ nhớ 52 được tạo cấu hình cụ thể để:

nếu xác định rằng bộ nhớ hệ thống trống nhỏ hơn ngưỡng quy định thứ nhất, thì xác định rằng điều kiện thu hồi bộ nhớ hiện được đáp ứng hiện thời.

Theo cách này, việc thiếu bộ nhớ có thể tránh được ở mức độ nhất định.

Một cách tùy chọn, môđun thu hồi bộ nhớ 52 được tạo cấu hình cụ thể để:

nếu xác định là ứng dụng được khởi động hoặc chuyển sang giao diện nền trước, thì xác định, theo thông tin chiếm dụng bộ nhớ của ứng dụng và kích cỡ của bộ nhớ hệ thống trống, xem việc thu hồi bộ nhớ có cần được thực hiện hay không; và nếu việc thu hồi bộ nhớ cần được thực hiện, thì xác định rằng điều kiện thu hồi bộ nhớ hiện được đáp ứng hiện thời.

Ứng dụng mới khởi động hoặc ứng dụng được chuyển sang giao diện nền trước thường là ứng dụng mà người dùng sắp sử dụng, và theo cách này, việc chạy bình thường ứng dụng mới khởi động hoặc ứng dụng chuyển sang giao diện nền trước có thể được đảm bảo, nhờ đó nâng cao hiệu suất hệ thống và cải thiện trải nghiệm của người dùng.

Một cách tùy chọn, môđun thu hồi bộ nhớ 52 được tạo cấu hình cụ thể để:

nếu số lượng ứng dụng hiện đang chạy trên nền sau lớn hơn số lượng ứng dụng quy định được phép chạy trên nền sau, thì xác định rằng điều kiện thu hồi bộ nhớ hiện được đáp ứng hiện thời.

Ngoài ra, ở đây, số lượng ứng dụng được phép chạy trên nền sau có thể được thiết đặt tại một thời khoảng theo kích cỡ của bộ nhớ trống hiện thời. Vì các ứng dụng đang chạy trên nền sau thường không phải là các ứng dụng được người dùng sử dụng hiện thời, nên có thể tránh được việc chiếm dụng bộ nhớ không cần thiết quá mức bằng cách giới hạn số lượng các ứng dụng này.

Một cách tùy chọn, môđun thu hồi bộ nhớ 52 được tạo cấu hình cụ thể để:

chọn, theo thứ tự ưu tiên đã lưu trữ trước của các ứng dụng có bộ nhớ cần được thu hồi, ứng dụng hiện đang chạy trên thiết bị đầu cuối và không được người dùng thao tác hoặc sử dụng, và xác định xem thời gian hiện thời có nằm trong phân đoạn thời gian sử dụng xác suất cao của ứng dụng đã chọn hay không, trong đó phân đoạn thời gian sử dụng xác suất cao của ứng dụng được xác định dựa vào thông tin thao tác và sử dụng của người dùng của ứng dụng; nếu thời gian hiện thời không nằm trong phân đoạn thời gian sử dụng xác suất cao của ứng dụng đã chọn, thì xác định rằng ứng dụng đã chọn là ứng dụng có bộ nhớ cần được thu hồi, và tiến hành thu hồi bộ nhớ bằng cách gọi ra giao diện thu hồi bộ nhớ được cấp bởi chế độ lõi (kernel); và lặp lại các bước trên cho đến khi đáp ứng điều kiện kết thúc thu hồi bộ nhớ.

Theo cách này, bộ nhớ của một ứng dụng (ứng dụng này thường là ứng dụng không được người dùng sử dụng thường xuyên) có quyền ưu tiên tương đối thấp được ưu tiên chọn để thu hồi, trong đó thời gian hiện thời không nằm trong phân đoạn thời gian chạy xác suất cao của ứng dụng, bằng cách này nâng cao hiệu suất hệ thống và cải thiện trải nghiệm của người dùng.

Một cách tùy chọn, điều kiện kết thúc thu hồi bộ nhớ bao gồm một hoặc nhiều điều kiện sau:

kích cỡ của bộ nhớ hệ thống trống lớn hơn ngưỡng quy định thứ hai;

bộ nhớ cần có cho ứng dụng được khởi động hiện thời là nhỏ hơn bộ nhớ hệ thống trống;

bộ nhớ cần có cho ứng dụng được chuyển sang giao diện nền trước là nhỏ hơn bộ nhớ hệ thống trống; và

số lượng ứng dụng đang chạy trên nền sau nhỏ hơn hoặc bằng số lượng ứng dụng quy định được phép chạy trên nền sau.

Một cách tùy chọn, mô đun thu hồi bộ nhớ 52 được tạo cấu hình cụ thể để thiết đặt thứ tự ưu tiên của các ứng dụng có bộ nhớ cần được thu hồi theo các bước sau:

tạo nhóm các ứng dụng được cài đặt trong thiết bị đầu cuối theo tần suất sử

dụng của các ứng dụng, trong đó các ứng dụng trong cùng một nhóm có cùng một dải tần suất sử dụng; và

xác định thứ tự ưu tiên thu hồi bộ nhớ của các ứng dụng trong mỗi nhóm theo dải tần suất sử dụng tương ứng với mỗi nhóm, trong đó các ứng dụng trong cùng một nhóm có cùng một thứ tự ưu tiên.

Một cách tùy chọn, môđun thu hồi bộ nhớ 52 được tạo cấu hình cụ thể để:

nếu nhận thấy thời lượng lưu trữ, của ứng dụng trong thiết bị đầu cuối, trong nền sau vượt quá thời lượng quy định, thì thu hồi bộ nhớ của ứng dụng có thời lượng lưu trữ trong nền sau vượt quá thời lượng quy định.

Ở đây, ứng dụng có thời lượng lưu trữ trong nền sau tương đối dài thường là ứng dụng không được người dùng thao tác trong khoảng thời gian hiện thời, và bộ nhớ của ứng dụng có thể được giải phóng và do đó được cấp cho ứng dụng cần bộ nhớ lớn hơn, nhờ đó nâng cao tỷ lệ sử dụng hiệu quả của bộ nhớ.

Phương án 5

Như được thể hiện trên Fig.6, Fig.6 là sơ đồ cấu trúc của thiết bị người dùng 60 theo một phương án của sáng chế, thiết bị người dùng 60 này bao gồm bộ xử lý 61, bộ nhớ 62, và kênh truyền 63. Bộ nhớ 62 được tạo cấu hình để lưu trữ lệnh thực thi, và bao gồm thiết bị nhớ trong 621 và bộ nhớ ngoài 622. Thiết bị nhớ trong 621 ở đây còn được gọi là bộ nhớ trong, và được tạo cấu hình để lưu trữ tạm thời dữ liệu hoạt động của bộ xử lý 61, và dữ liệu được trao đổi với bộ nhớ ngoài 622 như đĩa cứng. Bộ xử lý 61 trao đổi dữ liệu với bộ nhớ ngoài 622 bằng cách sử dụng thiết bị nhớ trong 621, và khi thiết bị người dùng 60 chạy, bộ xử lý 61 và bộ nhớ 62 truyền thông với nhau bằng cách sử dụng kênh truyền 63, để cho bộ xử lý 61 thực thi các lệnh sau ở chế độ người dùng:

giám sát thông tin thao tác và sử dụng của người dùng và thông tin chiếm dụng bộ nhớ của các ứng dụng cài đặt trong thiết bị đầu cuối; và khi xác định, theo ít nhất một trong số thông tin thao tác và sử dụng của người dùng hoặc thông tin chiếm dụng bộ nhớ của các ứng dụng cài đặt trong thiết bị đầu cuối, rằng điều kiện thu hồi bộ nhớ hiện được đáp ứng hiện thời, xác định, theo thông

tin thao tác và sử dụng của người dùng và thông tin chiếm dụng bộ nhớ của các ứng dụng hiện đang chạy trên thiết bị đầu cuối và từ các ứng dụng hiện đang chạy trên thiết bị đầu cuối, ứng dụng có bộ nhớ cần được thu hồi, và tiến hành thu hồi bộ nhớ bằng cách gọi ra giao diện thu hồi bộ nhớ được cấp bởi chế độ lõi (kernel).

Một cách tùy chọn, trong các lệnh được thực thi bởi bộ xử lý 61, việc xác định, theo thông tin thao tác và sử dụng của người dùng và thông tin chiếm dụng bộ nhớ của các ứng dụng cài đặt trong thiết bị đầu cuối, rằng điều kiện thu hồi bộ nhớ hiện được đáp ứng hiện thời bao gồm: xác định thời khoảng thu hồi bộ nhớ theo thông tin thao tác và sử dụng của người dùng và thông tin chiếm dụng bộ nhớ của các ứng dụng cài đặt trong thiết bị đầu cuối; và xác định rằng điều kiện thu hồi bộ nhớ hiện được đáp ứng hiện thời khi thời gian hiện thời đã đến thời gian thu hồi bộ nhớ được chỉ ra bởi thời khoảng thu hồi bộ nhớ.

Một cách tùy chọn, trong các lệnh được thực thi bởi bộ xử lý 61, việc xác định thời khoảng thu hồi bộ nhớ bao gồm:

xác định riêng thông tin thao tác và sử dụng của người dùng và thông tin chiếm dụng bộ nhớ, trong mỗi phân đoạn thời gian trong số ít nhất một phân đoạn thời gian thiết đặt trước, của các ứng dụng cài đặt trong thiết bị đầu cuối; và đối với mỗi phân đoạn thời gian trong số ít nhất một phân đoạn thời gian thiết đặt trước, xác định riêng thời khoảng thu hồi bộ nhớ trong phân đoạn thời gian theo thông tin thao tác và sử dụng của người dùng và thông tin chiếm dụng bộ nhớ, trong phân đoạn thời gian này, của các ứng dụng cài đặt trong thiết bị đầu cuối.

Một cách tùy chọn, trong các lệnh được thực thi bởi bộ xử lý 61, việc xác định, theo thông tin chiếm dụng bộ nhớ của các ứng dụng cài đặt trong thiết bị đầu cuối, rằng điều kiện thu hồi bộ nhớ hiện được đáp ứng hiện thời bao gồm: nếu xác định rằng bộ nhớ hệ thống trống nhỏ hơn ngưỡng quy định thứ nhất, thì xác định rằng điều kiện thu hồi bộ nhớ hiện được đáp ứng hiện thời.

Một cách tùy chọn, trong các lệnh được thực thi bởi bộ xử lý 61, việc xác

định, theo thông tin thao tác và sử dụng của người dùng và thông tin chiếm dụng bộ nhớ của các ứng dụng cài đặt trong thiết bị đầu cuối, rằng điều kiện thu hồi bộ nhớ hiện được đáp ứng hiện thời bao gồm: nếu xác định là ứng dụng được khởi động hoặc chuyển sang giao diện nền trước, thì xác định, theo thông tin chiếm dụng bộ nhớ của ứng dụng và kích cỡ của bộ nhớ hệ thống trống, xem việc thu hồi bộ nhớ có cần được thực hiện hay không; và nếu việc thu hồi bộ nhớ cần được thực hiện, thì xác định rằng điều kiện thu hồi bộ nhớ hiện được đáp ứng hiện thời.

Một cách tùy chọn, trong các lệnh được thực thi bởi bộ xử lý 61, việc xác định, theo thông tin thao tác và sử dụng của người dùng của các ứng dụng cài đặt trong thiết bị đầu cuối, rằng điều kiện thu hồi bộ nhớ hiện được đáp ứng hiện thời bao gồm: nếu số lượng ứng dụng hiện đang chạy trên nền sau lớn hơn số lượng ứng dụng quy định được phép chạy trên nền sau, thì xác định rằng điều kiện thu hồi bộ nhớ hiện được đáp ứng hiện thời.

Ngoài ra, ở đây, số lượng ứng dụng được phép chạy trên nền sau có thể được thiết đặt tại một thời khoảng theo kích cỡ của bộ nhớ trống hiện thời. Vì các ứng dụng đang chạy trên nền sau thường không phải là các ứng dụng được người dùng sử dụng hiện thời, nên có thể tránh được việc chiếm dụng bộ nhớ không cần thiết quá mức bằng cách giới hạn số lượng các ứng dụng này.

Một cách tùy chọn, trong các lệnh được thực thi bởi bộ xử lý 61, việc xác định ứng dụng có bộ nhớ cần được thu hồi, và tiến hành thu hồi bộ nhớ bằng cách gọi ra giao diện thu hồi bộ nhớ được cấp bởi chế độ lõi (kernel) bao gồm: chọn, theo thứ tự ưu tiên đã lưu trữ trước của các ứng dụng có bộ nhớ cần được thu hồi, ứng dụng hiện đang chạy trên thiết bị đầu cuối và không được người dùng thao tác hoặc sử dụng, và xác định xem thời gian hiện thời có nằm trong phân đoạn thời gian sử dụng xác suất cao của ứng dụng đã chọn hay không, trong đó phân đoạn thời gian sử dụng xác suất cao của ứng dụng được xác định dựa vào thông tin thao tác và sử dụng của người dùng của ứng dụng; nếu thời gian hiện thời không nằm trong phân đoạn thời gian sử dụng xác suất cao của ứng dụng đã chọn, thì xác định rằng ứng dụng đã chọn là ứng dụng có bộ nhớ

cần được thu hồi, và tiến hành thu hồi bộ nhớ bằng cách gọi ra giao diện thu hồi bộ nhớ được cấp bởi chế độ lõi (kernel); và lặp lại các bước nêu trên cho đến khi đáp ứng điều kiện kết thúc thu hồi bộ nhớ.

Một cách tùy chọn, điều kiện kết thúc thu hồi bộ nhớ bao gồm một hoặc nhiều điều kiện sau:

kích cỡ của bộ nhớ hệ thống trống lớn hơn ngưỡng quy định thứ hai;

bộ nhớ cần có cho ứng dụng được khởi động hiện thời là nhỏ hơn bộ nhớ hệ thống trống;

bộ nhớ cần có cho ứng dụng được chuyển sang giao diện nền trước là nhỏ hơn bộ nhớ hệ thống trống; và

số lượng ứng dụng đang chạy trên nền sau nhỏ hơn hoặc bằng số lượng ứng dụng quy định được phép chạy trên nền sau.

Một cách tùy chọn, trong các lệnh được thực thi bởi bộ xử lý 61, việc thiết đặt thứ tự ưu tiên của các ứng dụng có bộ nhớ cần được thu hồi bao gồm:

tạo nhóm các ứng dụng được cài đặt trong thiết bị đầu cuối theo tần suất sử dụng của các ứng dụng, trong đó các ứng dụng trong cùng một nhóm có cùng một dải tần suất sử dụng; và xác định thứ tự ưu tiên thu hồi bộ nhớ của các ứng dụng trong mỗi nhóm theo dải tần suất sử dụng tương ứng với mỗi nhóm, trong đó các ứng dụng trong cùng một nhóm có cùng một thứ tự ưu tiên.

Một cách tùy chọn, trong các lệnh được thực thi bởi bộ xử lý 61, nếu xác định, theo thông tin thao tác và sử dụng của người dùng của các ứng dụng cài đặt trong thiết bị đầu cuối, rằng điều kiện thu hồi bộ nhớ hiện được đáp ứng hiện thời, thì việc tiến hành thu hồi bộ nhớ bằng cách gọi ra giao diện thu hồi bộ nhớ được cấp bởi chế độ lõi (kernel) bao gồm: nếu nhận thấy thời lượng lưu trú, của ứng dụng trong thiết bị đầu cuối, trong nền sau vượt quá thời lượng quy định, thì xác định rằng điều kiện thu hồi bộ nhớ hiện được đáp ứng hiện thời, và thu hồi, bằng cách gọi ra giao diện thu hồi bộ nhớ được cấp bởi chế độ lõi (kernel), bộ nhớ của ứng dụng có thời lượng lưu trú trong nền sau vượt quá thời lượng quy định.

Phương án 6

Fig.7 mô tả cấu trúc của thiết bị đầu cuối 700 theo sáng chế, và thiết bị đầu cuối 700 bao gồm: ít nhất một bộ xử lý 701, ít nhất một giao diện mạng 704 hoặc giao diện người dùng 703 khác, bộ nhớ 705, và ít nhất một kênh truyền thông 702. Kênh truyền thông 702 được tạo cấu hình để thực hiện kết nối và truyền thông giữa các thành phần. Ở đây, bộ nhớ 705 bao gồm thiết bị nhớ trong và bộ nhớ ngoài. Thiết bị nhớ trong còn được gọi là bộ nhớ trong, và được tạo cấu hình để lưu trữ tạm thời dữ liệu hoạt động của bộ xử lý 701, và dữ liệu được trao đổi với bộ nhớ ngoài như đĩa cứng. Bộ xử lý 701 trao đổi dữ liệu với bộ nhớ ngoài bằng cách sử dụng thiết bị nhớ trong.

Bộ nhớ 705 lưu trữ các phần tử, môđun thực thi hoặc kết quả dữ liệu dưới đây, hoặc các tập hợp con của chúng, hoặc các tập hợp mở rộng của chúng.

Hệ điều hành 7051 gồm các chương trình hệ thống khác nhau, và có thể có hai mức chạy: chế độ lõi (kernel) 7051a và chế độ người dùng 7051b. Khi một quy trình tiến hành gọi ra hệ thống và nhập mã kernel để thi hành, quy trình này ở chế độ chạy kernel, còn được gọi ngắn gọn là chế độ lõi (kernel). Khi quy trình ở chế độ lõi (kernel), mã kernel khi thực thi có thể sử dụng ngăn xếp kernel của quy trình hiện thời, và mỗi quy trình có ngăn xếp kernel riêng của nó. Khi quy trình chạy mã của người dùng, quy trình ở chế độ chạy mã người dùng, được gọi ngắn gọn là chế độ người dùng. Trong trường hợp này, bộ xử lý 701 chạy mã người dùng có mức đặc quyền thấp nhất.

Môđun chương trình ứng dụng 7052 gồm nhiều chương trình ứng dụng khác nhau (gọi là Ứng dụng cho ngắn gọn).

Theo phương án này của sáng chế, bộ xử lý 701 được tạo cấu hình để tiến hành các bước sau ở chế độ người dùng bằng cách gọi ra chương trình hoặc lệnh lưu trữ trong bộ nhớ 705:

giám sát thông tin thao tác và sử dụng của người dùng và thông tin chiếm dụng bộ nhớ của các ứng dụng cài đặt trong thiết bị đầu cuối; và khi xác định, theo ít nhất một trong số thông tin thao tác và sử dụng của người dùng hoặc

thông tin chiếm dụng bộ nhớ của các ứng dụng cài đặt trong thiết bị đầu cuối, rằng điều kiện thu hồi bộ nhớ hiện được đáp ứng hiện thời, xác định, theo thông tin thao tác và sử dụng của người dùng và thông tin chiếm dụng bộ nhớ của các ứng dụng hiện đang chạy trên thiết bị đầu cuối và từ các ứng dụng hiện đang chạy trên thiết bị đầu cuối, ứng dụng có bộ nhớ cần được thu hồi, và tiến hành thu hồi bộ nhớ bằng cách gọi ra giao diện thu hồi bộ nhớ được cấp bởi chế độ lõi (kernel).

Một cách tùy chọn, theo một phương án, bộ xử lý 701 còn được tạo cấu hình để: xác định thời khoảng thu hồi bộ nhớ theo thông tin thao tác và sử dụng của người dùng và thông tin chiếm dụng bộ nhớ của các ứng dụng cài đặt trong thiết bị đầu cuối; và xác định rằng điều kiện thu hồi bộ nhớ hiện được đáp ứng hiện thời khi thời gian hiện thời đã đến thời gian thu hồi bộ nhớ được chỉ ra bởi thời khoảng thu hồi bộ nhớ.

Ngoài ra, bộ xử lý 701 còn được tạo cấu hình để: xác định riêng thông tin thao tác và sử dụng của người dùng và thông tin chiếm dụng bộ nhớ, trong mỗi phân đoạn thời gian trong số ít nhất một phân đoạn thời gian thiết đặt trước, của các ứng dụng cài đặt trong thiết bị đầu cuối; và đối với mỗi phân đoạn thời gian trong số ít nhất một phân đoạn thời gian thiết đặt trước, xác định riêng thời khoảng thu hồi bộ nhớ trong phân đoạn thời gian theo thông tin thao tác và sử dụng của người dùng và thông tin chiếm dụng bộ nhớ, trong phân đoạn thời gian này, của các ứng dụng cài đặt trong thiết bị đầu cuối.

Một cách tùy chọn, theo phương án khác, bộ xử lý 701 còn được tạo cấu hình để nếu xác định rằng bộ nhớ hệ thống trống nhỏ hơn ngưỡng quy định thứ nhất, thì xác định rằng điều kiện thu hồi bộ nhớ hiện được đáp ứng hiện thời.

Một cách tùy chọn, theo phương án khác nữa, bộ xử lý 701 còn được tạo cấu hình để: nếu xác định là ứng dụng được khởi động hoặc chuyển sang giao diện nền trước, thì xác định, theo thông tin chiếm dụng bộ nhớ của ứng dụng và kích cỡ của bộ nhớ hệ thống trống, việc thu hồi bộ nhớ có cần được thực hiện hay không; và nếu việc thu hồi bộ nhớ cần được thực hiện, thì xác định rằng điều kiện thu hồi bộ nhớ hiện được đáp ứng hiện thời.

Một cách tùy chọn, theo phương án khác nữa, bộ xử lý 701 còn được tạo cấu hình để nếu số lượng ứng dụng hiện đang chạy trên nền sau lớn hơn số lượng ứng dụng quy định được phép chạy trên nền sau, thì xác định rằng điều kiện thu hồi bộ nhớ hiện được đáp ứng hiện thời.

Ngoài ra, ở đây, số lượng ứng dụng được phép chạy trên nền sau có thể được thiết đặt tại một thời khoảng theo kích cỡ của bộ nhớ trống hiện thời. Vì các ứng dụng đang chạy trên nền sau thường không phải là các ứng dụng được người dùng sử dụng hiện thời, nên có thể tránh được việc chiếm dụng bộ nhớ không cần thiết quá mức bằng cách giới hạn số lượng các ứng dụng này.

Ngoài ra, bộ xử lý 701 còn được tạo cấu hình để xác định ứng dụng có bộ nhớ cần được thu hồi, và tiến hành thu hồi bộ nhớ bằng cách gọi ra giao diện thu hồi bộ nhớ được cấp bởi chế độ lõi (kernel) bao gồm: chọn, theo thứ tự ưu tiên đã lưu trữ trước của các ứng dụng có bộ nhớ cần được thu hồi, ứng dụng hiện đang chạy trên thiết bị đầu cuối và không được người dùng thao tác hoặc sử dụng, và xác định xem thời gian hiện thời có nằm trong phân đoạn thời gian sử dụng xác suất cao của ứng dụng đã chọn hay không, trong đó phân đoạn thời gian sử dụng xác suất cao của ứng dụng được xác định dựa vào thông tin thao tác và sử dụng của người dùng của ứng dụng; nếu thời gian hiện thời không nằm trong phân đoạn thời gian sử dụng xác suất cao của ứng dụng đã chọn, thì xác định rằng ứng dụng đã chọn là ứng dụng có bộ nhớ cần được thu hồi, và tiến hành thu hồi bộ nhớ bằng cách gọi ra giao diện thu hồi bộ nhớ được cấp bởi chế độ lõi (kernel); và lặp lại các bước nêu trên cho đến khi đáp ứng điều kiện kết thúc thu hồi bộ nhớ.

Điều kiện kết thúc thu hồi bộ nhớ bao gồm một hoặc nhiều điều kiện sau:

kích cỡ của bộ nhớ hệ thống trống lớn hơn ngưỡng quy định thứ hai;

bộ nhớ cần có cho ứng dụng được khởi động hiện thời là nhỏ hơn bộ nhớ hệ thống trống;

bộ nhớ cần có cho ứng dụng được chuyển sang giao diện nền trước là nhỏ hơn bộ nhớ hệ thống trống; và

số lượng ứng dụng đang chạy trên nền sau nhỏ hơn hoặc bằng số lượng ứng dụng quy định được phép chạy trên nền sau.

Ngoài ra, bộ xử lý 701 còn được tạo cấu hình để: tạo nhóm các ứng dụng được cài đặt trên thiết bị đầu cuối theo tần suất sử dụng của các ứng dụng, trong đó các ứng dụng trong cùng một nhóm có cùng một dải tần suất sử dụng; và xác định thứ tự ưu tiên thu hồi bộ nhớ của các ứng dụng trong mỗi nhóm theo dải tần suất sử dụng tương ứng với mỗi nhóm, trong đó các ứng dụng trong cùng một nhóm có cùng một thứ tự ưu tiên.

Một cách tùy chọn, theo một phương án khác nữa, bộ xử lý 701 còn được tạo cấu hình để: nếu nhận thấy thời lượng lưu trú, của ứng dụng trong thiết bị đầu cuối, trong nền sau vượt quá thời lượng quy định, thì thu hồi bộ nhớ của ứng dụng có thời lượng lưu trú trong nền sau vượt quá thời lượng quy định.

Ngoài ra, thiết bị đầu cuối 700 còn có thể thực hiện phương pháp và các phương án trên các hình vẽ từ Fig.2 đến Fig.4, và chi tiết không được mô tả lại ở đây theo phương án này của sáng chế.

Có thể thấy rằng sau khi các giải pháp nêu trên được sử dụng, khi một ứng dụng được thao tác và sử dụng thường xuyên bởi người dùng và chiếm dụng bộ nhớ tương đối lớn, tần suất thu hồi bộ nhớ có thể tăng, để đảm bảo việc sử dụng bộ nhớ bình thường.

Phương án 7

Một phương án của sáng chế liên quan đến thiết bị đầu cuối 800 và phương pháp thu hồi bộ nhớ, trong đó thiết bị đầu cuối 800 có thể là máy điện thoại di động, máy tính bảng, thiết bị trợ giúp số cá nhân (PDA - *Personal Digital Assistant*), điểm bán hàng (POS - *Point of Sales*), hoặc máy tính trong xe.

Fig.8 là sơ đồ cấu trúc của thiết bị đầu cuối 800 theo một phương án của sáng chế.

Theo Fig.8, thiết bị đầu cuối 800 theo phương án này của sáng chế bao gồm: bộ xử lý 810, bộ nhớ 820, thiết bị đầu vào 830, bộ nguồn 850, mạch tần số vô tuyến (RF - *Radio Frequency*) 860, mạch âm tần 870, và môđun mạng không

dây (WiFi - *Wireless Fidelity*) 880.

Bộ nhớ 820 bao gồm thiết bị nhớ trong 821 và bộ nhớ ngoài 822. Thiết bị nhớ trong 821 được tạo cấu hình để lưu trữ tạm thời dữ liệu hoạt động của bộ xử lý 810, và dữ liệu được trao đổi với bộ nhớ ngoài 822 như đĩa cứng. Bộ xử lý 810 trao đổi dữ liệu với bộ nhớ ngoài 822 bằng cách sử dụng thiết bị nhớ trong 821. Thiết bị nhớ trong 821 có thể là một trong số bộ nhớ truy nhập ngẫu nhiên bất khả biến (NVRAM - *Non-Volatile Random Access Memory*), bộ nhớ truy nhập ngẫu nhiên động (DRAM - *Dynamic Random Access Memory*), bộ nhớ truy nhập ngẫu nhiên tĩnh (SRAM - *Static RAM*), hoặc bộ nhớ tốc độ nhanh; và bộ nhớ ngoài 822 có thể là đĩa cứng, đĩa quang, thẻ nhớ tốc độ nhanh USB, đĩa mềm, hoặc ổ đĩa băng.

Bộ xử lý 810 thực thi các lệnh trong bộ nhớ 820 ở chế độ người dùng, để: giám sát thông tin thao tác và sử dụng của người dùng và thông tin chiếm dụng bộ nhớ của các ứng dụng cài đặt trong thiết bị đầu cuối; và khi xác định, theo ít nhất một trong số thông tin thao tác và sử dụng của người dùng hoặc thông tin chiếm dụng bộ nhớ của các ứng dụng cài đặt trong thiết bị đầu cuối, rằng điều kiện thu hồi bộ nhớ hiện được đáp ứng hiện thời, thì xác định, theo thông tin thao tác và sử dụng của người dùng và thông tin chiếm dụng bộ nhớ của các ứng dụng hiện đang chạy trên thiết bị đầu cuối và từ các ứng dụng hiện đang chạy trên thiết bị đầu cuối, ứng dụng có bộ nhớ cần được thu hồi, và tiến hành thu hồi bộ nhớ bằng cách gọi ra giao diện thu hồi bộ nhớ được cấp bởi chế độ lõi (kernel). Việc thu hồi bộ nhớ có thể được kích hoạt theo thời khoảng thu hồi bộ nhớ đã quy định, và cụ thể, bộ xử lý 810 có thể xác định riêng thông tin thao tác và sử dụng của người dùng và thông tin chiếm dụng bộ nhớ, trong mỗi phân đoạn thời gian trong số ít nhất một phân đoạn thời gian thiết đặt trước, của các ứng dụng cài đặt trong thiết bị đầu cuối; và đối với mỗi phân đoạn thời gian trong số ít nhất một phân đoạn thời gian thiết đặt trước, xác định riêng thời khoảng thu hồi bộ nhớ trong phân đoạn thời gian theo thông tin thao tác và sử dụng của người dùng và thông tin chiếm dụng bộ nhớ, trong phân đoạn thời gian này, của các ứng dụng cài đặt trong thiết bị đầu cuối. Một cách tùy chọn, việc

thu hồi bộ nhớ còn có thể được kích hoạt khi xác định rằng bộ nhớ hệ thống trống nhỏ hơn ngưỡng quy định thứ nhất. Việc thu hồi bộ nhớ cũng có thể được kích hoạt nếu xác định là ứng dụng được khởi động hoặc chuyển sang giao diện nền trước và xác định, theo thông tin chiếm dụng bộ nhớ của ứng dụng và kích cỡ của bộ nhớ hệ thống trống, rằng việc thu hồi bộ nhớ cần được tiến hành. Điều kiện thu hồi bộ nhớ cũng có thể được kích hoạt khi số lượng ứng dụng hiện đang chạy trên nền sau lớn hơn số lượng ứng dụng quy định được phép chạy trên nền sau. Ngoài ra, ở đây, số lượng ứng dụng được phép chạy trên nền sau có thể được thiết đặt tại một thời khoảng theo kích cỡ của bộ nhớ trống hiện thời. Vì các ứng dụng đang chạy trên nền sau thường không phải là các ứng dụng được người dùng sử dụng hiện thời, nên có thể tránh được việc chiếm dụng bộ nhớ không cần thiết quá mức bằng cách giới hạn số lượng các ứng dụng này.

Một cách tùy chọn, dựa vào các điều kiện kích hoạt nêu trên, quy trình trong đó bộ xử lý 810 tiến hành thu hồi bộ nhớ có thể là: chọn, theo thứ tự ưu tiên đã lưu trữ trước của các ứng dụng có bộ nhớ cần được thu hồi, ứng dụng hiện đang chạy trên thiết bị đầu cuối và không được người dùng thao tác hoặc sử dụng, và xác định xem thời gian hiện thời có nằm trong phân đoạn thời gian sử dụng xác suất cao của ứng dụng đã chọn hay không, trong đó phân đoạn thời gian sử dụng xác suất cao của ứng dụng được xác định dựa vào thông tin thao tác và sử dụng của người dùng của ứng dụng; nếu thời gian hiện thời không nằm trong phân đoạn thời gian sử dụng xác suất cao của ứng dụng đã chọn, thì xác định rằng ứng dụng đã chọn là ứng dụng có bộ nhớ cần được thu hồi, và tiến hành thu hồi bộ nhớ bằng cách gọi ra giao diện thu hồi bộ nhớ được cấp bởi chế độ lõi (kernel); và lặp lại các bước nêu trên cho đến khi đáp ứng điều kiện kết thúc thu hồi bộ nhớ. Ngoài ra, điều kiện kết thúc thu hồi bộ nhớ có thể bao gồm một hoặc nhiều điều kiện sau: kích cỡ của bộ nhớ hệ thống trống lớn hơn ngưỡng quy định thứ hai; bộ nhớ cần có cho ứng dụng được khởi động hiện thời là nhỏ hơn bộ nhớ hệ thống trống; bộ nhớ cần có cho ứng dụng được chuyển sang giao diện nền trước là nhỏ hơn bộ nhớ hệ thống trống; và số lượng ứng dụng đang chạy trên nền sau nhỏ hơn hoặc bằng số lượng ứng dụng quy định

được phép chạy trên nền sau. Ngoài ra, các ứng dụng cài đặt trong thiết bị đầu cuối có thể được tạo nhóm theo tần suất sử dụng của các ứng dụng, trong đó các ứng dụng trong cùng một nhóm có cùng một dải tần suất sử dụng. Thứ tự ưu tiên thu hồi bộ nhớ của các ứng dụng trong mỗi nhóm được xác định theo dải tần suất sử dụng tương ứng với mỗi nhóm, trong đó các ứng dụng trong cùng một nhóm có cùng một thứ tự ưu tiên.

Ngoài ra, bộ xử lý 810 còn có thể tiến hành, khi xác định rằng điều kiện thu hồi bộ nhớ hiện được đáp ứng hiện thời, thu hồi bộ nhớ bằng cách gọi ra giao diện thu hồi bộ nhớ được cấp bởi chế độ lõi (kernel).

Thiết bị đầu vào 830 có thể được tạo cấu hình để thu thông tin chữ số hoặc ký tự đầu vào, và tạo đầu vào tín hiệu liên quan đến cài đặt của người dùng và điều khiển chức năng của thiết bị đầu cuối 800. Cụ thể, theo phương án này của sáng chế, thiết bị đầu vào 830 có thể bao gồm bảng cảm ứng 831. Bảng cảm ứng 831, còn được gọi là màn hình cảm ứng, có thể thu thập thao tác chạm (như thao tác được thực hiện bởi người dùng trên bảng cảm ứng 831 hoặc gần bảng cảm ứng 831 bằng cách sử dụng ngón tay, hoặc đối tượng hoặc phụ kiện phù hợp bất kỳ như bút trỏ stylus) được thực hiện bởi người dùng trên hoặc gần bảng cảm ứng, và điều khiển thiết bị kết nối tương ứng theo chương trình thiết đặt trước. Một cách tùy chọn, bảng cảm ứng 831 có thể gồm hai phần: thiết bị phát hiện chạm và bộ điều khiển chạm. Thiết bị phát hiện chạm phát hiện hướng chạm của người dùng, phát hiện tín hiệu được mang theo thao tác chạm, và gửi tín hiệu đến bộ điều khiển chạm. Bộ điều khiển chạm thu thông tin chạm từ thiết bị phát hiện chạm, biến đổi thông tin chạm thành các tọa độ chạm, và gửi tọa độ chạm đến bộ xử lý 810, và có thể thu chỉ lệnh truyền từ bộ xử lý 810 và thi hành chỉ lệnh này. Ngoài ra, bảng cảm ứng 831 có thể được thực hiện ở nhiều loại, như loại điện trở, loại tụ điện, loại hồng ngoại, và loại sóng âm bề mặt. Ngoài bảng cảm ứng 831, thiết bị đầu vào 830 còn có thể bao gồm một hoặc nhiều thiết bị đầu vào 832 khác, trong đó một hoặc nhiều thiết bị đầu vào 832 khác có thể bao gồm nhưng không giới hạn ở một hoặc nhiều loại sau: bàn phím vật lý, phím chức năng (như phím điều khiển âm lượng và phím bật/tắt), bi xoay, chuột, cần

điều khiển, hoặc tương tự.

Thiết bị đầu cuối 800 còn có thể bao gồm thiết bị màn hình 840, trong đó thiết bị màn hình 840 có thể được tạo cấu hình để hiển thị thông tin được nhập bởi người dùng hoặc thông tin được cấp cho người dùng, cũng như các giao diện lệnh đơn khác nhau của thiết bị đầu cuối 800. Thiết bị màn hình 840 có thể bao gồm bảng hiển thị 841. Một cách tùy chọn, dạng màn hình tinh thể lỏng (LCD - *Liquid Crystal Display*), điốt phát quang hữu cơ (OLED - *Organic Light-Emitting Diode*), hoặc tương tự có thể được dùng để tạo cấu hình bảng hiển thị 841.

Theo phương án này của sáng chế, bảng cảm ứng 831 bao phủ bảng hiển thị 841 để tạo thành màn hình cảm ứng. Khi phát hiện thấy thao tác chạm trên hoặc gần màn hình cảm ứng, màn hình cảm ứng truyền thao tác chạm này đến bộ xử lý 810 để xác định loại sự kiện chạm, và sau đó bộ xử lý 810 cung cấp đầu ra trực quan tương ứng trên màn hình cảm ứng theo kiểu sự kiện chạm.

Theo phương án này của sáng chế, màn hình cảm ứng bao gồm vùng hiển thị giao diện ứng dụng và vùng hiển thị điều khiển chung. Cách bố trí vùng hiển thị giao diện ứng dụng và vùng hiển thị điều khiển chung không có giới hạn, và có thể là cách bố trí có thể phân biệt hai vùng hiển thị, như cách bố trí trên-dưới hoặc cách bố trí trái-phải. Vùng hiển thị giao diện ứng dụng có thể được sử dụng để hiển thị giao diện chương trình ứng dụng. Mỗi giao diện có thể bao gồm các phần tử giao diện, như biểu tượng hoặc tiện ích màn hình chính hoặc cả hai, của ít nhất một chương trình ứng dụng. Vùng hiển thị giao diện ứng dụng cũng có thể là giao diện rỗng không chứa nội dung. Vùng hiển thị điều khiển chung được dùng để hiển thị sự điều khiển sử dụng tương đối cao, ví dụ, biểu tượng chương trình ứng dụng như nút thiết đặt, số giao diện, thanh cuộn, biểu tượng danh bạ điện thoại, hoặc tương tự.

Bộ xử lý 810 là trung tâm điều khiển của thiết bị đầu cuối 800, và sử dụng nhiều giao diện và dòng khác nhau để kết nối tất cả các phần của toàn bộ máy điện thoại di động. Bằng cách chạy hoặc thực thi chương trình phần mềm hoặc mô đun hoặc cả hai đều được lưu trữ trong thiết bị nhớ trong 821 và gọi ra dữ

liệu lưu trữ trong bộ nhớ ngoài 822, bộ xử lý 810 thực thi các chức năng khác nhau của thiết bị đầu cuối 800 và xử lý dữ liệu, để thực hiện giám sát tổng thể trên thiết bị đầu cuối 800.

Người có hiểu biết trung bình trong lĩnh vực kỹ thuật này cần phải hiểu rằng các phương án của sáng chế có thể được đề xuất dưới dạng phương pháp, hệ thống, hoặc sản phẩm chương trình máy tính. Do đó, sáng chế có thể sử dụng dạng phương án chỉ dùng phần cứng, phương án chỉ dùng phần mềm, hoặc phương án kết hợp phần mềm và phần cứng. Ngoài ra, sáng chế có thể sử dụng dạng sản phẩm chương trình máy tính được cài đặt trên một hoặc nhiều phương tiện nhớ dùng được bằng máy tính (bao gồm nhưng không giới hạn ở bộ nhớ đĩa, CD-ROM, bộ nhớ quang, và tương tự) chứa mã chương trình dùng được bằng máy tính.

Sáng chế được mô tả có dựa vào các lưu đồ và/hoặc các sơ đồ khối của phương pháp, thiết bị (hệ thống), và sản phẩm chương trình máy tính theo các phương án của sáng chế. Cần phải hiểu rằng các lệnh chương trình máy tính có thể được sử dụng để thực thi mỗi quy trình và/hoặc mỗi khối trong các lưu đồ và/hoặc các sơ đồ khối và tổ hợp của quy trình và/hoặc khối trong các lưu đồ và/hoặc các sơ đồ khối. Các lệnh chương trình máy tính này có thể được cấp cho máy tính đa năng, máy tính chuyên dụng, bộ xử lý nhúng trong, hoặc bộ xử lý của thiết bị xử lý điện thoại có thể lập trình bất kỳ khác để tạo ra máy, để cho các lệnh được thi hành bởi máy tính hoặc bộ xử lý của thiết bị xử lý dữ liệu có thể lập trình bất kỳ khác này tạo ra thiết bị thực hiện một chức năng cụ thể trong một hoặc nhiều quy trình trong các lưu đồ và/hoặc trong một hoặc nhiều khối trong các sơ đồ khối.

Các lệnh chương trình máy tính này có thể được lưu trữ trong bộ nhớ đọc được bằng máy tính để có thể lệnh cho máy tính hoặc thiết bị xử lý dữ liệu có thể lập trình bất kỳ khác làm việc theo một cách cụ thể, để cho các lệnh lưu trữ trong bộ nhớ đọc được bằng máy tính tạo ra vật phẩm chứa bộ lệnh. Bộ lệnh này thực hiện một chức năng cụ thể trong một hoặc nhiều quy trình trong các lưu đồ và/hoặc trong một hoặc nhiều khối trong các sơ đồ khối.

Các lệnh chương trình máy tính này có thể được nạp vào máy tính hoặc thiết bị xử lý dữ liệu có thể lập trình khác, để cho chuỗi các thao tác và các bước được thực hiện trên máy tính hoặc thiết bị có thể lập trình khác, nhờ đó tạo ra quy trình xử lý cài đặt trên máy tính. Do đó, các lệnh chạy trên máy tính hoặc thiết bị có thể lập trình khác cung cấp các bước để thực hiện một chức năng cụ thể trong một hoặc nhiều quy trình trong các lưu đồ và/hoặc trong một hoặc nhiều khối trong các sơ đồ khối.

Mặc dù một số phương án của sáng chế đã được mô tả, nhưng người có hiểu biết trung bình trong lĩnh vực kỹ thuật này có thể thay đổi và sửa đổi các phương án này khi họ tìm hiểu khái niệm sáng tạo cơ bản. Do đó, các điểm yêu cầu bảo hộ sau dự định được hiểu là bao hàm các phương án này và tất cả các thay đổi và các sửa đổi trong phạm vi bảo hộ của sáng chế.

Hiển nhiên, người có hiểu biết trung bình trong lĩnh vực kỹ thuật này có thể thực hiện các sửa đổi và các cải biến khác nhau đối với các phương án của sáng chế mà không vượt quá phạm vi của các phương án của sáng chế. Sáng chế dự định bao hàm các sửa đổi và các cải biến này với điều kiện chúng nằm trong phạm vi bảo hộ được định rõ bởi các yêu cầu bảo hộ dưới đây và các kỹ thuật tương đương của chúng.

YÊU CẦU BẢO HỘ

1. Phương pháp thu hồi bộ nhớ, trong đó các bước sau đây được thực hiện trong chế độ người dùng:

giám sát thông tin thao tác và sử dụng của người dùng và thông tin chiếm giữ bộ nhớ của các ứng dụng được cài đặt trên thiết bị đầu cuối; và

để phản hồi lại việc xác định, theo ít nhất một trong số thông tin thao tác và sử dụng của người dùng hoặc thông tin chiếm giữ bộ nhớ của các ứng dụng được cài đặt trên thiết bị đầu cuối, rằng điều kiện thu hồi bộ nhớ được đáp ứng hiện thời:

lựa chọn, theo thứ tự ưu tiên được lưu trữ trước của các ứng dụng, ứng dụng mà đang chạy trên thiết bị đầu cuối và không được thao tác hoặc được sử dụng bởi người dùng;

xác định rằng thời gian hiện thời có nằm trong phân đoạn thời gian sử dụng xác suất cao của ứng dụng được lựa chọn hay không, trong đó phân đoạn thời gian sử dụng xác suất cao của ứng dụng được xác định dựa trên thông tin thao tác và sử dụng của người dùng của ứng dụng;

thực hiện việc thu hồi bộ nhớ đối với ứng dụng được lựa chọn khi thời gian hiện thời không nằm trong phân đoạn thời gian sử dụng xác suất cao của ứng dụng được lựa chọn; và

lặp lại các bước nêu trên cho đến khi điều kiện kết thúc thu hồi bộ nhớ được đáp ứng.

2. Phương pháp theo điểm 1, trong đó bước xác định, theo ít nhất một trong số thông tin thao tác và sử dụng của người dùng hoặc thông tin chiếm giữ bộ nhớ của các ứng dụng được cài đặt trên thiết bị đầu cuối, rằng điều kiện thu hồi bộ nhớ được đáp ứng hiện thời bao gồm:

xác định khoảng thời gian thu hồi bộ nhớ theo thông tin thao tác và sử dụng của người dùng và thông tin chiếm giữ bộ nhớ của các ứng dụng được cài đặt trên thiết bị đầu cuối; và

trong đó điều kiện thu hồi bộ nhớ được đáp ứng hiện thời khi xác định rằng thời gian hiện thời đã đến thời gian thu hồi bộ nhớ được chỉ báo bởi khoảng thời gian thu hồi bộ nhớ.

3. Phương pháp theo điểm 2, trong đó bước xác định khoảng thời gian thu hồi bộ nhớ theo thông tin thao tác và sử dụng của người dùng và thông tin chiếm giữ bộ nhớ của các ứng dụng được cài đặt trên thiết bị đầu cuối bao gồm:

xác định riêng thông tin thao tác và sử dụng của người dùng và thông tin chiếm giữ bộ nhớ, trong mỗi phân đoạn thời gian của ít nhất một phân đoạn thời gian được thiết đặt trước, của các ứng dụng được cài đặt trên thiết bị đầu cuối; và

đối với mỗi phân đoạn thời gian của ít nhất một phân đoạn thời gian được thiết đặt trước, xác định riêng khoảng thời gian thu hồi bộ nhớ trong phân đoạn thời gian theo thông tin thao tác và sử dụng của người dùng và thông tin chiếm giữ bộ nhớ, trong phân đoạn thời gian, của các ứng dụng được cài đặt trên thiết bị đầu cuối.

4. Phương pháp theo điểm 1, trong đó điều kiện thu hồi bộ nhớ được đáp ứng hiện thời khi xác định rằng bộ nhớ trống hệ thống nhỏ hơn ngưỡng quy định thứ nhất.

5. Phương pháp theo điểm 1, trong đó bước xác định, theo ít nhất một trong số thông tin thao tác và sử dụng của người dùng hoặc thông tin chiếm giữ bộ nhớ của các ứng dụng được cài đặt trên thiết bị đầu cuối, rằng điều kiện thu hồi bộ nhớ được đáp ứng hiện thời bao gồm:

xác định, theo thông tin chiếm giữ bộ nhớ của ứng dụng và kích cỡ của bộ nhớ trống hệ thống, rằng việc thu hồi bộ nhớ cần được thực hiện hay không khi ứng dụng được khởi động hoặc được chuyển sang giao diện nền trước; và

trong đó điều kiện thu hồi bộ nhớ được đáp ứng hiện thời khi xác định rằng việc thu hồi bộ nhớ cần được thực hiện.

6. Phương pháp theo điểm 1, trong đó điều kiện thu hồi bộ nhớ được đáp ứng hiện thời khi xác định rằng số lượng ứng dụng đang chạy trong nền sau lớn hơn

số lượng ứng dụng quy định được cho phép để chạy trong nền sau.

7. Phương pháp theo điểm 1, trong đó điều kiện kết thúc thu hồi bộ nhớ bao gồm một hoặc nhiều điều kiện sau đây:

kích cỡ của bộ nhớ trống hệ thống lớn hơn ngưỡng quy định thứ hai;

bộ nhớ được yêu cầu bởi ứng dụng đang được khởi động nhỏ hơn bộ nhớ trống hệ thống;

bộ nhớ được yêu cầu bởi ứng dụng được chuyển đổi sang giao diện nền trước nhỏ hơn bộ nhớ trống hệ thống; và

số lượng ứng dụng chạy trong nền sau nhỏ hơn hoặc bằng số lượng ứng dụng quy định được cho phép để chạy trong nền sau.

8. Phương pháp theo điểm bất kỳ trong số các điểm từ 1 đến 7, trong đó thứ tự ưu tiên của các ứng dụng được thiết đặt theo các bước sau đây:

tạo nhóm các ứng dụng được cài đặt trên thiết bị đầu cuối theo tần suất sử dụng của các ứng dụng, trong đó các ứng dụng trong cùng nhóm có cùng dải tần suất sử dụng; và

xác định thứ tự ưu tiên thu hồi bộ nhớ của các ứng dụng trong mỗi nhóm theo dải tần suất sử dụng mà tương ứng với mỗi nhóm, trong đó các ứng dụng trong cùng nhóm có cùng thứ tự ưu tiên.

9. Hệ thống máy tính, bao gồm:

bộ nhớ có cấu trúc để lưu trữ các lệnh có thể đọc được bởi máy tính; và

ít nhất một bộ xử lý có cấu trúc để đọc các lệnh có thể đọc được bởi máy tính từ bộ nhớ để khiến ít nhất một bộ xử lý:

giám sát thông tin thao tác và sử dụng của người dùng và thông tin chiếm giữ bộ nhớ của các ứng dụng được cài đặt trên thiết bị đầu cuối; và

để phản hồi lại việc xác định, theo ít nhất một trong số thông tin thao tác và sử dụng của người dùng hoặc thông tin chiếm giữ bộ nhớ của các ứng dụng được cài đặt trên thiết bị đầu cuối, rằng điều kiện thu hồi bộ nhớ được đáp ứng hiện thời,

lựa chọn, theo thứ tự ưu tiên được lưu trữ trước của các ứng dụng, ứng dụng mà đang chạy trên thiết bị đầu cuối và không được thao tác hoặc được sử dụng bởi người dùng;

xác định rằng thời gian hiện thời có nằm trong phân đoạn thời gian sử dụng xác suất cao của ứng dụng được lựa chọn hay không, trong đó phân đoạn thời gian sử dụng xác suất cao của ứng dụng được xác định dựa trên thông tin thao tác và sử dụng của người dùng của ứng dụng;

thực hiện việc thu hồi bộ nhớ đối với ứng dụng được lựa chọn khi thời gian hiện thời không nằm trong phân đoạn thời gian sử dụng xác suất cao của ứng dụng được lựa chọn; và

lặp lại các bước nêu trên cho đến khi điều kiện kết thúc thu hồi bộ nhớ được đáp ứng.

10. Hệ thống máy tính theo điểm 9, trong đó ít nhất một bộ xử lý có cấu trúc để:

xác định khoảng thời gian thu hồi bộ nhớ theo thông tin thao tác và sử dụng của người dùng và thông tin chiếm giữ bộ nhớ của các ứng dụng được cài đặt trên thiết bị đầu cuối; và

trong đó điều kiện thu hồi bộ nhớ được đáp ứng hiện thời khi xác định rằng thời gian hiện thời đã đến thời gian thu hồi bộ nhớ được chỉ báo bởi khoảng thời gian thu hồi bộ nhớ.

11. Hệ thống máy tính theo điểm 10, trong đó ít nhất một bộ xử lý có cấu trúc để:

xác định riêng thông tin thao tác và sử dụng của người dùng và thông tin chiếm giữ bộ nhớ, trong mỗi phân đoạn thời gian của ít nhất một phân đoạn thời gian được thiết đặt trước, của các ứng dụng được cài đặt trên thiết bị đầu cuối; và

đối với mỗi phân đoạn thời gian của ít nhất một phân đoạn thời gian được thiết đặt trước, xác định riêng khoảng thời gian thu hồi bộ nhớ trong phân đoạn thời gian theo thông tin thao tác và sử dụng của người dùng và thông tin chiếm giữ bộ nhớ, trong phân đoạn thời gian, của các ứng dụng được cài đặt trên thiết

bị đầu cuối.

12. Hệ thống máy tính theo điểm 9, trong đó điều kiện thu hồi bộ nhớ được đáp ứng hiện thời khi xác định rằng bộ nhớ trống hệ thống nhỏ hơn ngưỡng quy định thứ nhất.

13. Hệ thống máy tính theo điểm 9, trong đó ít nhất một bộ xử lý có cấu trúc để:

xác định, theo thông tin chiếm giữ bộ nhớ của ứng dụng và kích cỡ của bộ nhớ trống hệ thống, rằng việc thu hồi bộ nhớ cần được thực hiện hay không khi ứng dụng được khởi động hoặc được chuyển sang giao diện nền trước; và

trong đó điều kiện thu hồi bộ nhớ được đáp ứng hiện thời khi xác định rằng việc thu hồi bộ nhớ cần được thực hiện.

14. Hệ thống máy tính theo điểm 9, trong đó điều kiện thu hồi bộ nhớ được đáp ứng hiện thời khi xác định rằng số lượng ứng dụng đang chạy trong nền sau lớn hơn số lượng ứng dụng quy định được cho phép để chạy trong nền sau.

15. Hệ thống máy tính theo điểm 9, trong đó điều kiện kết thúc thu hồi bộ nhớ bao gồm một hoặc nhiều điều kiện sau đây:

kích cỡ của bộ nhớ trống hệ thống lớn hơn ngưỡng quy định thứ hai;

bộ nhớ được yêu cầu bởi ứng dụng đang được khởi động nhỏ hơn bộ nhớ trống hệ thống;

bộ nhớ được yêu cầu bởi ứng dụng được chuyển đổi sang giao diện nền trước nhỏ hơn bộ nhớ trống hệ thống; và

số lượng ứng dụng chạy trong nền sau nhỏ hơn hoặc bằng số lượng ứng dụng quy định được cho phép để chạy trong nền sau.

16. Hệ thống máy tính theo điểm 9 hoặc 15, trong đó ít nhất một bộ xử lý có cấu trúc để:

tạo nhóm các ứng dụng được cài đặt trên thiết bị đầu cuối theo tần suất sử dụng của các ứng dụng, trong đó các ứng dụng trong cùng nhóm có cùng dải tần suất sử dụng; và

xác định thứ tự ưu tiên thu hồi bộ nhớ của các ứng dụng trong mỗi nhóm

theo dải tần suất sử dụng mà tương ứng với mỗi nhóm, trong đó các ứng dụng trong cùng nhóm có cùng thứ tự ưu tiên.

17. Phương tiện lưu trữ đọc được bởi máy tính, mà lưu trữ chương trình máy tính, trong đó chương trình máy tính khiến cho máy tính:

giám sát thông tin thao tác và sử dụng của người dùng và thông tin chiếm giữ bộ nhớ của các ứng dụng được cài đặt trên thiết bị đầu cuối; và

để phản hồi lại việc xác định, theo ít nhất một trong số thông tin thao tác và sử dụng của người dùng hoặc thông tin chiếm giữ bộ nhớ của các ứng dụng được cài đặt trên thiết bị đầu cuối, rằng điều kiện thu hồi bộ nhớ được đáp ứng hiện thời:

lựa chọn, theo thứ tự ưu tiên được lưu trữ trước của các ứng dụng, ứng dụng mà đang chạy trên thiết bị đầu cuối và không được thao tác hoặc được sử dụng bởi người dùng;

xác định rằng thời gian hiện thời có nằm trong phân đoạn thời gian sử dụng xác suất cao của ứng dụng được lựa chọn hay không, trong đó phân đoạn thời gian sử dụng xác suất cao của ứng dụng được xác định dựa trên thông tin thao tác và sử dụng của người dùng của ứng dụng;

thực hiện việc thu hồi bộ nhớ đối với ứng dụng được lựa chọn khi thời gian hiện thời không nằm trong phân đoạn thời gian sử dụng xác suất cao của ứng dụng được lựa chọn; và

lặp lại các bước nêu trên cho đến khi điều kiện kết thúc thu hồi bộ nhớ được thỏa mãn.

18. Phương tiện lưu trữ đọc được bởi máy tính theo điểm 17, trong đó chương trình máy tính còn khiến máy tính:

xác định khoảng thời gian thu hồi bộ nhớ theo thông tin thao tác và sử dụng của người dùng và thông tin chiếm giữ bộ nhớ của các ứng dụng được cài đặt trên thiết bị đầu cuối; và

trong đó điều kiện thu hồi bộ nhớ được đáp ứng hiện thời khi xác định rằng thời gian hiện thời đã đến thời gian thu hồi bộ nhớ được chỉ báo bởi khoảng thời

gian thu hồi bộ nhớ.

19. Phương tiện lưu trữ đọc được bởi máy tính theo điểm 17, trong đó điều kiện thu hồi bộ nhớ được đáp ứng hiện thời khi xác định rằng bộ nhớ trống hệ thống nhỏ hơn ngưỡng quy định thứ nhất.

20. Phương tiện lưu trữ đọc được bởi máy tính theo điểm 17, trong đó chương trình máy tính còn khiến máy tính:

xác định, theo thông tin chiếm giữ bộ nhớ của ứng dụng và kích cỡ của bộ nhớ trống hệ thống, rằng việc thu hồi bộ nhớ cần được thực hiện hay không khi ứng dụng được khởi động hoặc được chuyển sang giao diện nền trước; và

trong đó điều kiện thu hồi bộ nhớ được đáp ứng hiện thời khi xác định rằng việc thu hồi bộ nhớ cần được thực hiện.

1/6

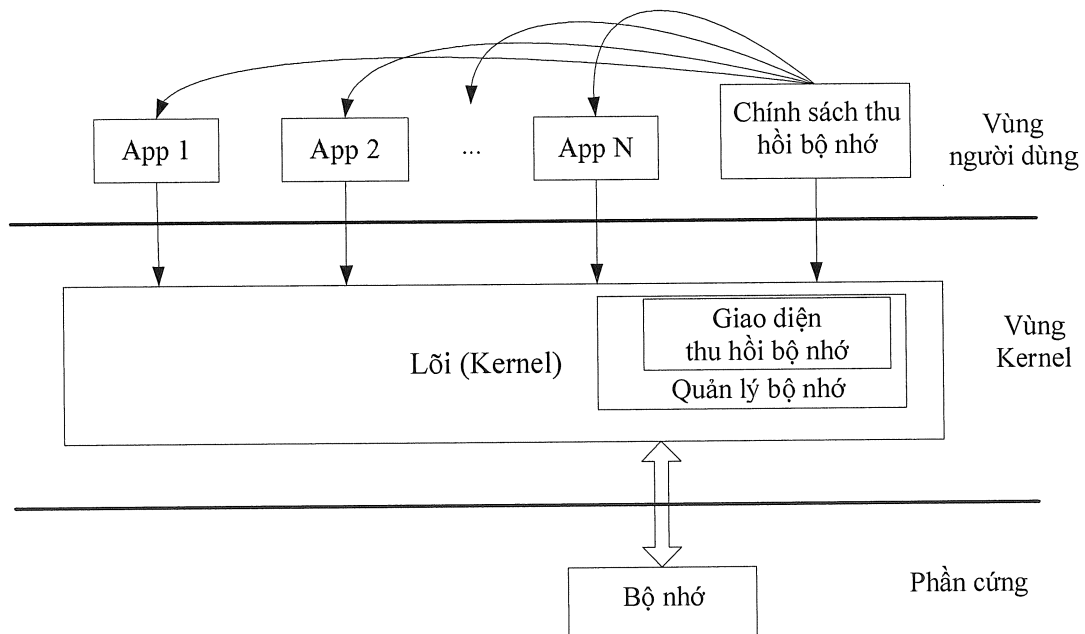


Fig.1

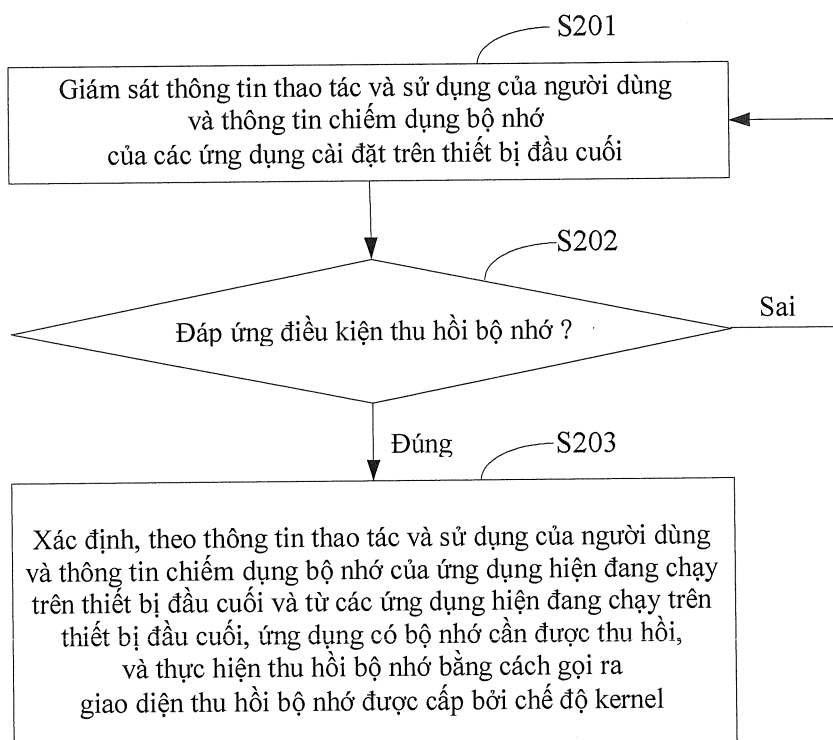


Fig.2

2/6

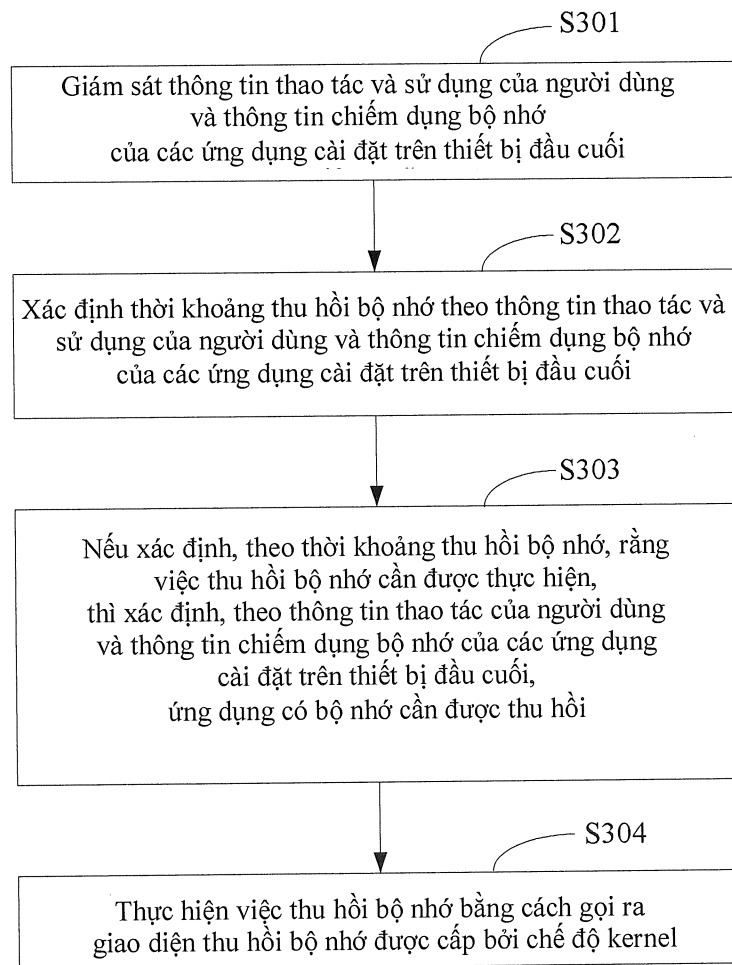


Fig.3

3/6

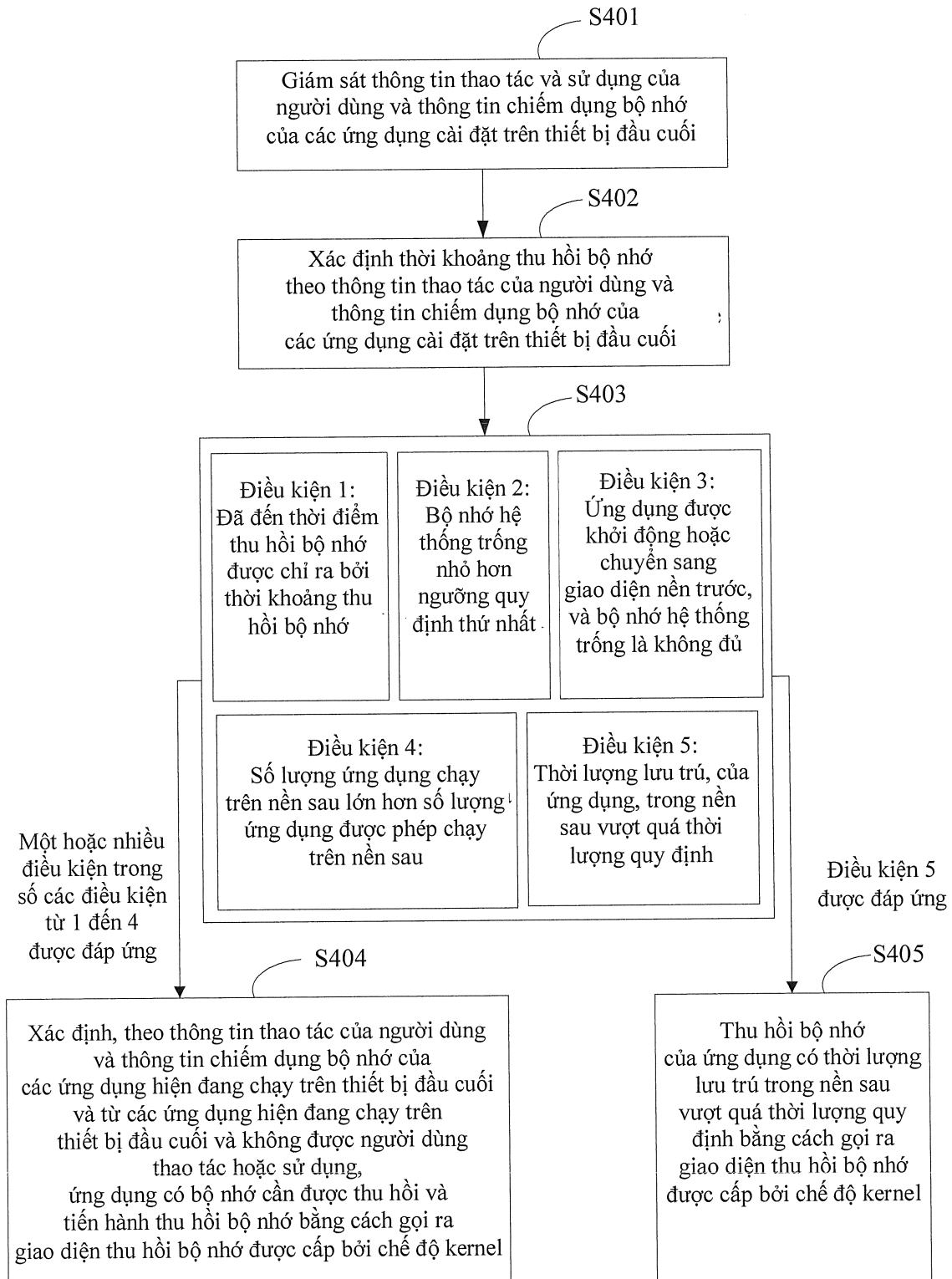


Fig.4

4/6

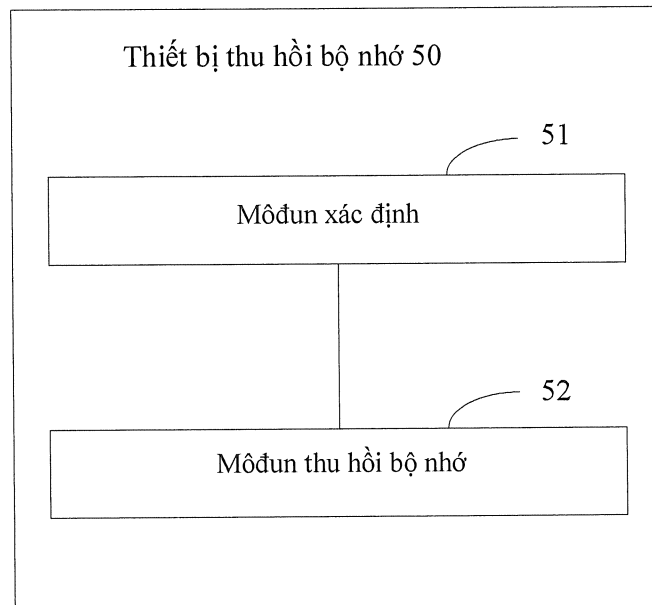


Fig.5

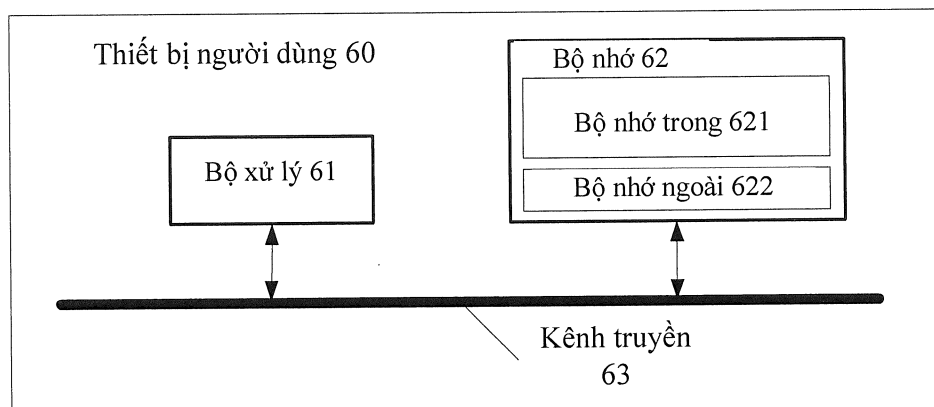


Fig.6

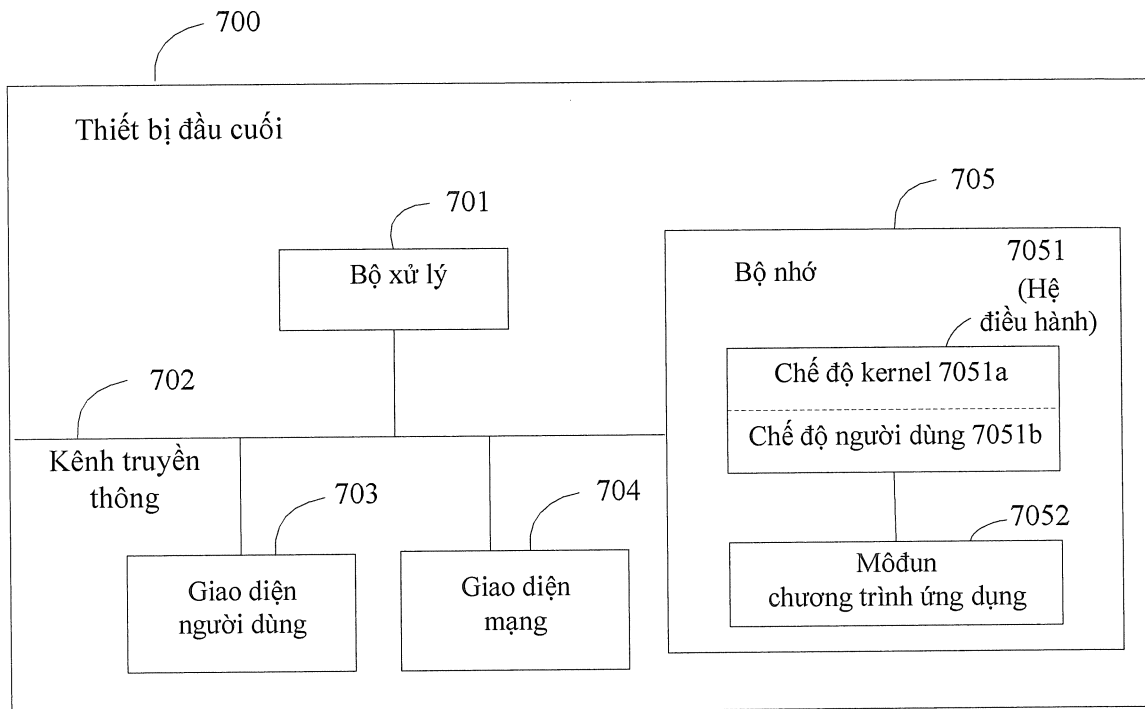


Fig.7

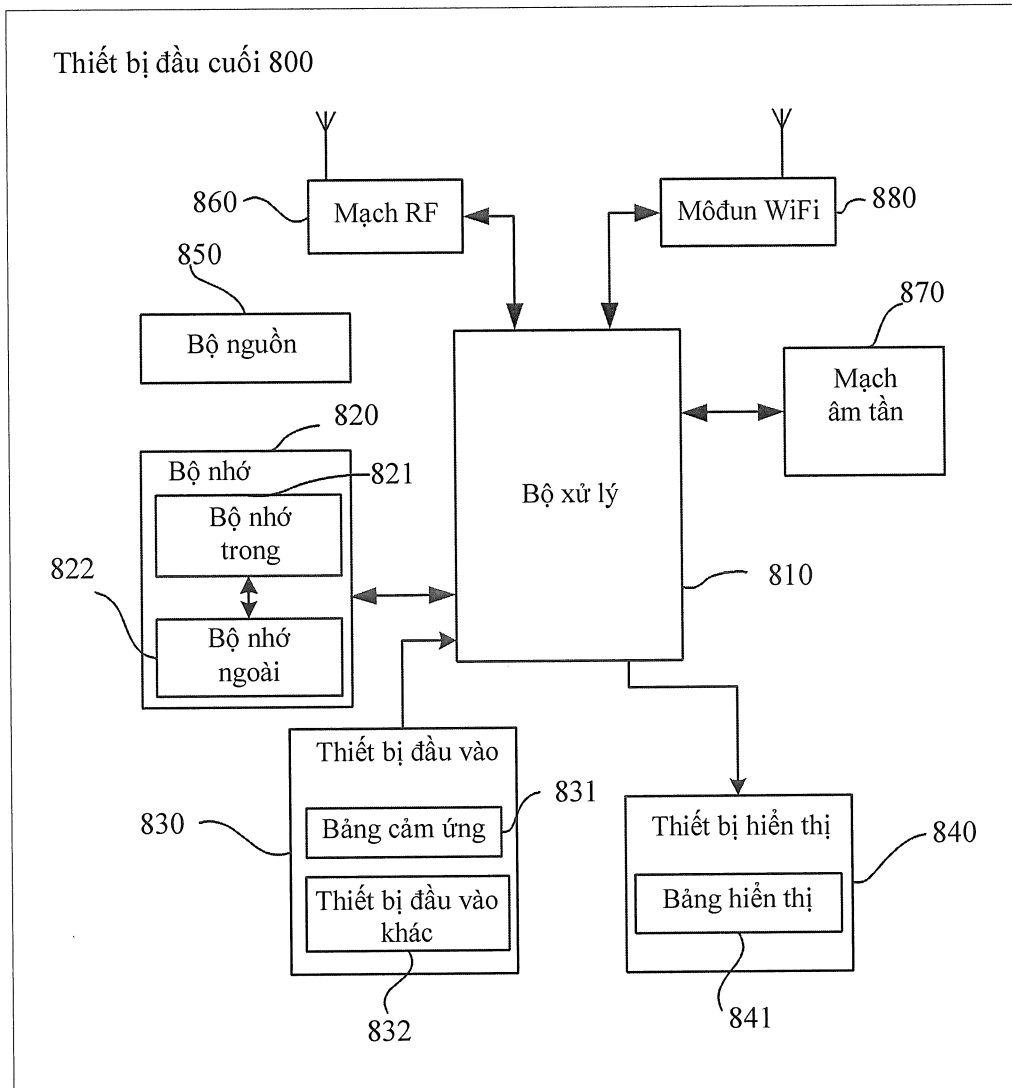


Fig.8