

(19) 日本国特許庁(JP)

(12) 公開特許公報(A)

(11) 特許出願公開番号

特開2010-250703

(P2010-250703A)

(43) 公開日 平成22年11月4日(2010.11.4)

(51) Int.Cl.

G06F 3/048 (2006.01)

F I

G06F 3/048 655C

テーマコード (参考)

5E501

審査請求 未請求 請求項の数 12 O L (全 14 頁)

(21) 出願番号 特願2009-101376 (P2009-101376)  
 (22) 出願日 平成21年4月17日 (2009. 4. 17)

(71) 出願人 000001007  
 キヤノン株式会社  
 東京都大田区下丸子3丁目30番2号  
 (74) 代理人 100076428  
 弁理士 大塚 康徳  
 (74) 代理人 100112508  
 弁理士 高柳 司郎  
 (74) 代理人 100115071  
 弁理士 大塚 康弘  
 (74) 代理人 100116894  
 弁理士 木村 秀二  
 (74) 代理人 100130409  
 弁理士 下山 治  
 (74) 代理人 100134175  
 弁理士 永川 行光

最終頁に続く

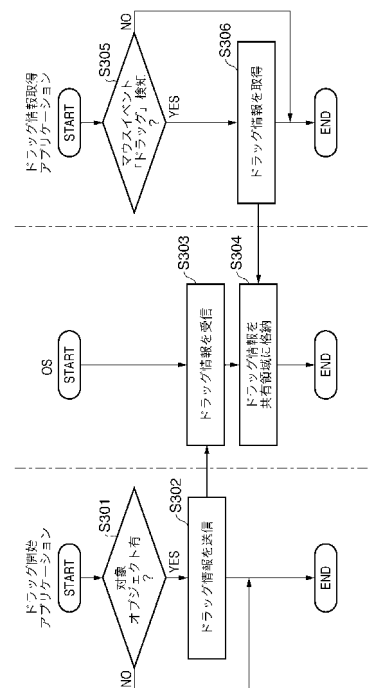
(54) 【発明の名称】 オペレーティングシステム、プログラム、情報処理装置および制御方法

## (57) 【要約】

【課題】マルチウィンドウシステムにおいて、ドラッグされたオブジェクトに関する情報をより効率的に伝達する。

【解決手段】メモリを有するコンピュータ上で動作すると複数のウィンドウを表示可能な表示画面を持つマルチウィンドウシステムを実現するオペレーティングシステムであって、前記コンピュータを、前記表示画面に、前記オペレーティングシステムの制御下で動作する第1プログラムのウィンドウを描画する描画手段と、前記第1プログラムから、前記第1プログラムのウィンドウにおいてドラッグされたオブジェクトに関するドラッグ情報を受信する受信手段と、前記オペレーティングシステムの制御下で動作する第2プログラムからアクセス可能な共有領域を前記メモリに割り当てる割り当て手段と、前記ドラッグ情報を前記共有領域に格納する格納手段と、として機能させるためのオペレーティングシステムを提供する。

【選択図】 図3



**【特許請求の範囲】****【請求項 1】**

メモリを有するコンピュータ上で動作すると複数のウィンドウを表示可能な表示画面を持つマルチウィンドウシステムを実現するオペレーティングシステムにしたがって動作する情報処理装置であって、

前記表示画面に、前記オペレーティングシステムの制御下で動作する第 1 プログラムのウィンドウを描画する描画手段と、

前記第 1 プログラムから、前記第 1 プログラムのウィンドウにおいてドラッグされたオブジェクトに関するドラッグ情報を受信する受信手段と、

前記オペレーティングシステムの制御下で動作する第 2 プログラムからアクセス可能な共有領域を前記メモリに割り当てる割り当て手段と、

前記ドラッグ情報を前記共有領域に格納する格納手段と、

を備えることを特徴とする情報処理装置。

10

**【請求項 2】**

メモリを有するコンピュータ上で動作すると複数のウィンドウを表示可能な表示画面を持つマルチウィンドウシステムを実現するオペレーティングシステムにしたがって動作する情報処理装置であって、

前記表示画面に、前記オペレーティングシステムの制御下で動作する第 1 プログラムのウィンドウを描画する描画手段と、

前記オペレーティングシステムの制御下で動作する第 2 プログラムから、前記第 1 プログラムのウィンドウにおいてオブジェクトがドラッグされた場合に当該オブジェクトに関するドラッグ情報を当該第 2 プログラムに通知するように要求する通知要求を受信する通知要求受信手段と、

前記通知要求を前記メモリに格納する格納手段と、

前記第 1 プログラムから前記ドラッグ情報を受信するドラッグ情報受信手段と、

前記受信手段により前記ドラッグ情報が受信された場合に、前記メモリに格納されている通知要求に対応する前記第 2 プログラムに対して当該ドラッグ情報を通知する通知手段と、

を備えることを特徴とする情報処理装置。

20

**【請求項 3】**

前記第 2 プログラムから、当該第 2 プログラムに対応する前記通知要求を削除する削除要求を受信する削除要求受信手段と、

前記削除要求受信手段により前記削除要求が前記第 2 プログラムから受信された場合に、当該第 2 プログラムに対応する前記通知要求を前記メモリから削除する削除手段と、

を備えることを特徴とする請求項 2 に記載の情報処理装置。

30

**【請求項 4】**

前記ドラッグ情報は、前記オブジェクトに関連づけられたデータと当該データの種別とのうちの少なくとも一方を含むことを特徴とする請求項 1 乃至 3 のいずれか 1 項に記載の情報処理装置。

**【請求項 5】**

コンピュータ上で動作すると複数のウィンドウを表示可能な表示画面を持つマルチウィンドウシステムを実現するオペレーティングシステムの制御下で動作するプログラムにしたがって動作する情報処理装置であって、

前記オペレーティングシステムが描画した、当該オペレーティングシステムの制御下で動作する他のプログラムのウィンドウにおいて、オブジェクトがドラッグされたことを検出する検出手段と、

オブジェクトがドラッグされたことを前記検出手段が検出した場合に、当該オブジェクトに関するドラッグ情報を前記他のプログラムから取得する取得手段と、

を備えることを特徴とする情報処理装置。

40

**【請求項 6】**

50

前記ドラッグ情報は、前記オブジェクトに関連づけられたデータと当該データの種別とのうちの少なくとも一方を含むことを特徴とする請求項 5 に記載の情報処理装置。

【請求項 7】

メモリを有するコンピュータ上で動作すると複数のウインドウを表示可能な表示画面を持つマルチウインドウシステムを実現するオペレーティングシステムにしたがって情報処理装置が実行する制御方法であって、

描画手段が、前記表示画面に、前記オペレーティングシステムの制御下で動作する第 1 プログラムのウインドウを描画する描画工程と、

受信手段が、前記第 1 プログラムから、前記第 1 プログラムのウインドウにおいてドラッグされたオブジェクトに関するドラッグ情報を受信する受信工程と、

割り当て手段が、前記オペレーティングシステムの制御下で動作する第 2 プログラムからアクセス可能な共有領域を前記メモリに割り当てる割り当て工程と、

格納手段が、前記ドラッグ情報を前記共有領域に格納する格納工程と、  
を備えることを特徴とする制御方法。

【請求項 8】

メモリを有するコンピュータ上で動作すると複数のウインドウを表示可能な表示画面を持つマルチウインドウシステムを実現するオペレーティングシステムにしたがって情報処理装置が実行する制御方法であって、

描画手段が、前記表示画面に、前記オペレーティングシステムの制御下で動作する第 1 プログラムのウインドウを描画する描画工程と、

通知要求受信手段が、前記オペレーティングシステムの制御下で動作する第 2 プログラムから、前記第 1 プログラムのウインドウにおいてオブジェクトがドラッグされた場合に当該オブジェクトに関するドラッグ情報を当該第 2 プログラムに通知するように要求する通知要求を受信する通知要求受信工程と、

格納手段が、前記通知要求を前記メモリに格納する格納工程と、

ドラッグ情報受信手段が、前記第 1 プログラムから前記ドラッグ情報を受信するドラッグ情報受信工程と、

通知手段が、前記受信工程において前記ドラッグ情報が受信された場合に、前記メモリに格納されている通知要求に対応する前記第 2 プログラムに対して当該ドラッグ情報を通知する通知工程と、

を備えることを特徴とする制御方法。

【請求項 9】

コンピュータ上で動作すると複数のウインドウを表示可能な表示画面を持つマルチウインドウシステムを実現するオペレーティングシステムの制御下で動作するプログラムにしたがって情報処理装置が実行する制御方法であって、

検出手段が、前記オペレーティングシステムが描画した、当該オペレーティングシステムの制御下で動作する他のプログラムのウインドウにおいて、オブジェクトがドラッグされたことを検出する検出工程と、

取得手段が、オブジェクトがドラッグされたことを前記検出工程で検出した場合に、当該オブジェクトに関するドラッグ情報を前記他のプログラムから取得する取得工程と、

を備えることを特徴とする制御方法。

【請求項 10】

メモリを有するコンピュータ上で動作すると複数のウインドウを表示可能な表示画面を持つマルチウインドウシステムを実現するオペレーティングシステムであって、前記コンピュータを、

前記表示画面に、前記オペレーティングシステムの制御下で動作する第 1 プログラムのウインドウを描画する描画手段と、

前記第 1 プログラムから、前記第 1 プログラムのウインドウにおいてドラッグされたオブジェクトに関するドラッグ情報を受信する受信手段と、

前記オペレーティングシステムの制御下で動作する第 2 プログラムからアクセス可能な

10

20

30

40

50

共有領域を前記メモリに割り当てる割り当て手段と、  
前記ドラッグ情報を前記共有領域に格納する格納手段と、  
として機能させるためのオペレーティングシステム。

【請求項 11】

メモリを有するコンピュータ上で動作すると複数のウインドウを表示可能な表示画面を持つマルチウインドウシステムを実現するオペレーティングシステムであって、前記コンピュータを、

前記表示画面に、前記オペレーティングシステムの制御下で動作する第 1 プログラムのウインドウを描画する描画手段と、

前記オペレーティングシステムの制御下で動作する第 2 プログラムから、前記第 1 プログラムのウインドウにおいてオブジェクトがドラッグされた場合に当該オブジェクトに関するドラッグ情報を当該第 2 プログラムに通知するように要求する通知要求を受信する通知要求受信手段と、

前記通知要求を前記メモリに格納する格納手段と、

前記第 1 プログラムから前記ドラッグ情報を受信するドラッグ情報受信手段と、

前記受信手段において前記ドラッグ情報が受信された場合に、前記メモリに格納されている通知要求に対応する前記第 2 プログラムに対して当該ドラッグ情報を通知する通知手段と、

として機能させるための特徴とするオペレーティングシステム。

【請求項 12】

コンピュータ上で動作すると複数のウインドウを表示可能な表示画面を持つマルチウインドウシステムを実現するオペレーティングシステムの制御下で動作するプログラムであって、前記コンピュータを、

前記オペレーティングシステムが描画した、当該オペレーティングシステムの制御下で動作する他のプログラムのウインドウにおいて、オブジェクトがドラッグされたことを検出する検出手段と、

オブジェクトがドラッグされたことを前記検出手段で検出した場合に、当該オブジェクトに関するドラッグ情報を前記他のプログラムから取得する取得手段と、

として機能させるためのプログラム。

【発明の詳細な説明】

【技術分野】

【0001】

本発明は、オペレーティングシステム、プログラム、情報処理装置および制御方法に関し、特に、マルチウインドウシステムにおいてドラッグされたオブジェクトに関する情報を扱う技術に関する。

【背景技術】

【0002】

現在、コンピュータ上でマルチウインドウシステムを実現するオペレーティングシステム（OS）（例えば、Windows（登録商標））が普及している。マルチウインドウシステムでは、図 8 に示すように、1 つの表示画面 1800 上に複数のウインドウ 1801 及び 1802 を表示することが可能である。

【0003】

ここで、表示画面 1800 において、ウインドウ 1801 に表示されているオブジェクト（例えば、ファイル）をドラッグ&ドロップすることにより、ウインドウ 1802 を司るアプリケーションにこのオブジェクトを移動したりコピーしたりすることを考える。この場合、ウインドウ 1802 がウインドウ 1801 の下に隠れていてオブジェクトのドロップを受け付けることができない。そこで、ユーザはドラッグ&ドロップに先立ち、ウインドウ 1801 内のドラッグ&ドロップ対象のオブジェクトとウインドウ 1802 内のドロップ領域の少なくとも一部とが重ならずに表示されるよう、各ウインドウの位置を調整する必要がある（図 9 参照）。

10

20

30

40

50

## 【 0 0 0 4 】

ドラッグ&ドロップに伴うユーザの操作負担を軽減するいくつかの技術が提案されている。

## 【 0 0 0 5 】

特許文献 1 によれば、画面に表示されたアイコンのドラッグ操作が開始されたことを、OS が全てのウインドウに通知する。各ウインドウはドラッグされたアイコンの属性情報によって、ドラッグ操作に対応可能か否かの判断を行い、ウインドウを非表示の状態にするか、表示の状態にする。

## 【 0 0 0 6 】

特許文献 2 によれば、ドラッグ操作を開始すると、ドラッグされたオブジェクトのドロップ先となるアプリケーションに該当するウインドウをマルチウインドウ環境の最前面に表示してアクティブ化する。

## 【 0 0 0 7 】

特許文献 3 によれば、ファイルブラウザやデスクトップ上のアイコンのドラッグ操作を開始すると、ドラッグ開始位置の近傍にドロップ対象一覧ウインドウを最前面化して表示する。

## 【 先行技術文献 】

## 【 特許文献 】

## 【 0 0 0 8 】

【 特許文献 1 】 特開平 8 - 1 8 5 3 0 1 号公報

【 特許文献 2 】 特開 2 0 0 3 - 1 0 8 2 8 1 号公報

【 特許文献 3 】 特開 2 0 0 0 - 5 6 8 8 5 号公報

## 【 発明の概要 】

## 【 発明が解決しようとする課題 】

## 【 0 0 0 9 】

従来の技術では、ドラッグ対象のオブジェクトに関する情報を OS が全てのウインドウ（少なくとも、このオブジェクトを処理可能な全てのウインドウ）に対して通知していた。換言すれば、ドラッグ対象のオブジェクトの処理を希望しないプログラムに対しても、通知が行われていた。従って、OS が管理しているウインドウの数が多い場合などに特に、コンピュータにとっての処理負荷が大きくなるという問題があった。

## 【 0 0 1 0 】

本発明はこのような状況に鑑みてなされたものであり、マルチウインドウシステムにおいて、ドラッグされたオブジェクトに関する情報をより効率的に伝達する技術を提供することを目的とする。

## 【 課題を解決するための手段 】

## 【 0 0 1 1 】

上記課題を解決するために、第 1 の本発明は、メモリを有するコンピュータ上で動作すると複数のウインドウを表示可能な表示画面を持つマルチウインドウシステムを実現するオペレーティングシステムであって、前記コンピュータを、前記表示画面に、前記オペレーティングシステムの制御下で動作する第 1 プログラムのウインドウを描画する描画手段と、前記第 1 プログラムから、前記第 1 プログラムのウインドウにおいてドラッグされたオブジェクトに関するドラッグ情報を受信する受信手段と、前記オペレーティングシステムの制御下で動作する第 2 プログラムからアクセス可能な共有領域を前記メモリに割り当てる割り当て手段と、前記ドラッグ情報を前記共有領域に格納する格納手段と、として機能させるためのオペレーティングシステムを提供する。

## 【 0 0 1 2 】

また、第 2 の本発明は、メモリを有するコンピュータ上で動作すると複数のウインドウを表示可能な表示画面を持つマルチウインドウシステムを実現するオペレーティングシステムであって、前記コンピュータを、前記表示画面に、前記オペレーティングシステムの制御下で動作する第 1 プログラムのウインドウを描画する描画手段と、前記オペレーティ

10

20

30

40

50

ングシステムの制御下で動作する第２プログラムから、前記第１プログラムのウインドウにおいてオブジェクトがドラッグされた場合に当該オブジェクトに関するドラッグ情報を当該第２プログラムに通知するように要求する通知要求を受信する通知要求受信手段と、前記通知要求を前記メモリに格納する格納手段と、前記第１プログラムから前記ドラッグ情報を受信するドラッグ情報受信手段と、前記受信手段において前記ドラッグ情報が受信された場合に、前記メモリに格納されている通知要求に対応する前記第２プログラムに対して当該ドラッグ情報を通知する通知手段と、として機能させるための特徴とするオペレーティングシステムを提供する。

【００１３】

また、第３の本発明は、コンピュータ上で動作すると複数のウインドウを表示可能な表示画面を持つマルチウインドウシステムを実現するオペレーティングシステムの制御下で動作するプログラムであって、前記コンピュータを、前記オペレーティングシステムが描画した、当該オペレーティングシステムの制御下で動作する他のプログラムのウインドウにおいて、オブジェクトがドラッグされたことを検出する検出手段と、オブジェクトがドラッグされたことを前記検出手段で検出した場合に、当該オブジェクトに関するドラッグ情報を前記他のプログラムから取得する取得手段と、として機能させるためのプログラムを提供する。

【００１４】

なお、その他の本発明の特徴は、添付図面及び以下の発明を実施するための最良の形態における記載によって更に明らかになるものである。

【発明の効果】

【００１５】

以上の構成により、本発明によれば、マルチウインドウシステムにおいて、ドラッグされたオブジェクトに関する情報をより効率的に伝達することが可能となる。

【図面の簡単な説明】

【００１６】

【図１】実施例１に係るＯＳとアプリケーションとを実行するパーソナルコンピュータ（ＰＣ）１００の構成を示すブロック図である。

【図２】実施例１におけるＯＳ及びアプリケーションの動作の概要を示す図である。

【図３】実施例１においてＯＳがドラッグ情報を受信して共有領域に格納する処理の流れを示すフローチャートである。

【図４】実施例２におけるＯＳ及びアプリケーションの動作の概要を示す図である。

【図５】実施例２においてＯＳがドラッグ情報の通知要求を格納し、通知要求に従ってドラッグ情報をアプリケーションに対して通知する処理の流れを示すフローチャートである。

【図６】実施例３におけるアプリケーションの動作の概要を示す図である。

【図７】実施例３におけるアプリケーションの動作の流れを示すフローチャートである。

【図８】背景技術を説明する図である。

【図９】背景技術を説明する図である。

【発明を実施するための形態】

【００１７】

以下、添付図面を参照して、本発明の実施例を説明する。以下で説明される個別の実施例は、本発明の上位概念から下位概念までの種々の概念を理解するために役立つであろう。

【００１８】

なお、本発明の技術的範囲は、特許請求の範囲によって確定されるのであって、以下の個別の実施例によって限定されるわけではない。また、実施例の中で説明されている特徴の組み合わせすべてが、本発明に必須とは限らない。

【００１９】

また、以下の各実施例で利用するオペレーティングシステム（ＯＳ）は、Window

10

20

30

40

50

s（登録商標）である。しかし、他のOSを利用する場合であっても、利用するプログラム関数などが異なるだけで基本的なコンセプトは変わらない。従って、本実施例は、Windows（登録商標）に限定されず、コンピュータ上で動作すると複数のウインドウを表示可能な表示画面を持つマルチウインドウシステムを実現するものであれば、いかなるOSを利用しても構わない。

【実施例１】

【００２０】

図１は、実施例１に係るOSとアプリケーションとを実行するパーソナルコンピュータ（PC）１００（情報処理装置）の構成を示すブロック図である。

【００２１】

図１に示すように、PC１００は、CRT１０１、VRAM１０２、BMU１０３、キーボード１０４、PD１０５、CPU１０６、ROM１０７、RAM１０８、HDD１０９、FDD１１０、及びネットワークＩ／Ｆ１１１を有する。

【００２２】

CRT（Cathode Ray Tube）１０１には、例えば編集集中の文書、図形、画像、その他の編集情報、アイコン、メッセージ、メニュー、その他のユーザインタフェースなどの情報が表示される。

【００２３】

VRAM１０２には、CRT１０１に表示するための画像が描画される。このVRAM１０２に描画された画像データは、所定の規定に従ってCRT１０１に転送され、これによりCRT１０１に画像が表示される。

【００２４】

BMU（ビットムーブユニット）１０３は、例えば、メモリ間（例えば、VRAM１０２と他のメモリとの間）のデータ転送や、メモリと各Ｉ／Ｏデバイス（例えば、ネットワークＩ／Ｆ１１１）との間のデータ転送を制御する。

【００２５】

キーボード１０４は、文書等を入力するための各種キーを有する。

【００２６】

PD（ポインティングデバイス）１０５は、例えば、CRT１０１に表示されたアイコン、メニュー、その他のコンテンツを指示するために使用される。

【００２７】

CPU１０６は、ROM１０７、HDD１０９、又はフレキシブルディスク（FD）に格納された制御プログラムに基づいて、各デバイスを制御する。

【００２８】

ROM１０７は、各種制御プログラムやデータを保存する。

【００２９】

RAM１０８は、CPU１０６のワーク領域、エラー処理時のデータの退避領域、制御プログラムのロード領域等を有する。

【００３０】

HDD１０９は、PC１００内で実行される各制御プログラムやコンテンツを格納する。

【００３１】

例えば、HDD１０９には、ブラウザ、OS、アプリケーションプログラム、画像ファイル、ドキュメントファイル、音楽ファイル、電子アルバムデータ、電子アルバム編集プログラムなどが格納される。

【００３２】

FDD１１０は、フロッピー（登録商標）等に代表されるFDに対するアクセスを制御する。

【００３３】

ネットワークＩ／Ｆ１１１は、他のPCやプリンタ等とネットワークを介して通信を行

10

20

30

40

50

う。

【0034】

CPUバス112は、アドレスバス、データバス、及びコントロールバスを含む。

【0035】

CPU106に対する制御プログラムの提供は、ROM107、HDD109、FDD110から行うこともできるし、ネットワークI/F111を介してネットワーク経由で他のPC等から行うこともできる。

【0036】

以下の説明で、OS、ブラウザ、各アプリケーションプログラムおよびそれらのウィンドウが実行する動作は、CPU106がこれらをHDD109から読み出して起動し、上述の各モジュールを制御することにより実現される。

10

【0037】

図2は、実施例1におけるOS及びアプリケーションの動作の概要を示す図である。図2において、画面200は、OSによってCRT101に表示される表示画面である。ウィンドウ201は、アプリケーションのウィンドウであり、ドラッグ対象のオブジェクト207（例えば、ファイルアイコン）を含む。ウィンドウ202及びウィンドウ203はそれぞれ、別々のアプリケーションのウィンドウである。共有領域205は、OSがRAM108などのメモリに割り当てた共有領域であり、OSの制御下で動作する各アプリケーションからアクセス可能である。

【0038】

20

ウィンドウ201においてオブジェクト207がドラッグされると、矢印204に示すように、ウィンドウ201のアプリケーション（第1プログラム）はオブジェクト207に関する情報（ドラッグ情報）を共有領域205に格納する。

【0039】

これにより、他のアプリケーション、例えばウィンドウ203のアプリケーション（第2プログラム）は、矢印206に示すように、共有領域205を参照することによってドラッグ情報を取得することができる。

【0040】

なお、ドラッグ情報には例えば、ドラッグ対象のオブジェクトに関連づけられたデータと、このデータの種別（例えば、JPEG、TEXT等のファイルタイプ）とのうちの少なくとも一方が含まれる。

30

【0041】

図3は、実施例1においてOSがドラッグ情報を受信して共有領域に格納する処理の流れを示すフローチャートである。本フローチャートの各ステップの処理は、PC100のCPU106が、特に断らない限り、OSやアプリケーション（例えば、ウィンドウ201-203のアプリケーション）を実行することによって実現される。これは、以下の各フローチャートにおいても同様である。

【0042】

S301で、ウィンドウ201においてドラッグが開始されると、ウィンドウ201のアプリケーション（ドラッグ開始アプリケーション）は、ドラッグ対象のオブジェクトが存在するか否かを判定する。例えば範囲指定などのためにドラッグが行われた場合はドラッグ対象のオブジェクトが存在しないので、処理は終了する。そうでなければ、処理はS302に進む。S302で、ドラッグ開始アプリケーションは、OSに対してドラッグ情報を送信する。

40

【0043】

OSは、S303で、ドラッグ開始アプリケーションからドラッグ情報を受信し、RAM108に共有領域205を割り当てる。共有領域205は、他のウィンドウから参照できるセキュリティ属性を持つ。但し、共有領域205の割り当ては、事前に行われても構わない。次いでS304で、OSは、ドラッグ情報を共有領域205に格納する。

【0044】

50



S 3 0 4 の処理が実行されると、ウインドウ 2 0 3 のアプリケーション（ドラッグ情報取得アプリケーション）は、S 3 0 5 で、OS のマウスイベントを検知し、ドラッグが行われたか否か判定する。ドラッグが行われていれば、S 3 0 6 で、ウインドウ 2 0 3 のアプリケーションは共有領域 2 0 5 からドラッグ情報を取得する。

【 0 0 4 5 】

図 3 においてはウインドウ 2 0 3 のアプリケーションをドラッグ情報取得アプリケーションとして説明したが、代わりに、或いはこれに加えて、ウインドウ 2 0 2 のアプリケーションがドラッグ情報取得アプリケーションとして動作してもよい。

【 0 0 4 6 】

従って、ドラッグ情報を必要とするアプリケーションのみがドラッグ情報を取得するので、ドラッグ情報がより効率的に伝達され、P C 1 0 0 の処理負荷が軽減される。また、マウスカーソルがドラッグ情報を必要とするアプリケーションのウインドウ上に無い状態でも、ドラッグ情報の取得が可能である。

【 実施例 2 】

【 0 0 4 7 】

実施例 2 でも、実施例 1 の P C 1 0 0 （図 1 参照）を利用する。

【 0 0 4 8 】

図 4 は、実施例 2 における OS 及びアプリケーションの動作の概要を示す図である。図 4 において、画面 4 0 0 は、OS によって C R T 1 0 1 に表示される表示画面である。ウインドウ 4 0 1 は、アプリケーションのウインドウであり、ドラッグ対象のオブジェクト 4 0 9 （例えば、ファイルアイコン）を含む。ウインドウ 4 0 2 - 4 0 5 はそれぞれ、別々のアプリケーションのウインドウである。

【 0 0 4 9 】

他のアプリケーションにおけるドラッグ情報の取得を欲するアプリケーション（例えば、ウインドウ 4 0 3 のアプリケーション）は、矢印 4 0 6 に示すように、OS に対してイベント取得登録（通知要求）を行う。これにより、イベント取得リスト 4 0 7 に、ウインドウ 4 0 3 が登録される。

【 0 0 5 0 】

ウインドウ 4 0 1 においてオブジェクト 4 0 9 がドラッグされると、矢印 4 0 8 に示すように、ウインドウ 4 0 1 のアプリケーションは OS に対してドラッグ情報を送信する。ドラッグ情報を受信した OS は、イベント取得リスト 4 0 7 に従って、ドラッグ情報を各アプリケーションに対して順次送信する。これにより、ウインドウ 4 0 3 のアプリケーションは、ドラッグ情報を取得することができる。

【 0 0 5 1 】

図 5 は、実施例 2 において OS がドラッグ情報の通知要求を格納し、通知要求に従ってドラッグ情報をアプリケーションに対して通知する処理の流れを示すフローチャートである。

【 0 0 5 2 】

S 5 0 1 で、ドラッグ情報取得アプリケーション（例えば、ウインドウ 4 0 3 のアプリケーション）は、実行に必要な初期化処理を行う。S 5 0 2 で、ドラッグ情報取得アプリケーションは、OS に対してドラッグ情報の通知要求を送信する（即ち、イベント取得登録を行う）。S 5 0 3 で、OS は、通知要求を受信し（通知要求受信手段）、これを R A M 1 0 8 に格納する。

【 0 0 5 3 】

なお、不図示ではあるが、イベントにはドラッグイベントの他にもキーボードやマウスなどの入力イベントやウインドウの生成や表示などイベントが存在する。従って、必要のないイベントを取得できなうようにするためにフィルタリング設定ができることは言うまでもない。また、複数のアプリケーションがドラッグ情報取得アプリケーションとして動作してもよい。

【 0 0 5 4 】

10

20

30

40

50

次に、ドラッグ開始アプリケーションが S 3 0 1 及び S 3 0 2 の処理を行うが、これは実施例 1 と同様である（図 3 参照）。

【 0 0 5 5 】

その後、S 5 0 4 で、OS は、ドラッグ情報を受信し（ドラッグ情報受信手段）、且つ、フィルタ設定の条件に基づき、ドラッグ情報を取得するように設定されている通知要求を R A M 1 0 8 から取得する。

【 0 0 5 6 】

S 5 0 5 で、OS は、通知要求を参照する変数 I N D E X を 0 に設定し、S 5 0 4 で取得した通知要求の数を変数 C O U N T に設定する。S 5 0 6 で、OS は、変数 I N D E X < 変数 C O U N T であるか否かを判定し、Y e s であれば S 5 0 7 へ、N o であれば S 5 1 1 へ進む。

10

【 0 0 5 7 】

S 5 0 7 で、OS は、変数 I N D E X の示す通知要求に対応するドラッグ情報取得アプリケーションに対して、ドラッグ情報を通知する。ドラッグ情報取得アプリケーションは S 5 0 8 でこのドラッグ情報を受信する。S 5 0 9 で、OS は変数 I N D E X に 1 を加算して、S 5 0 6 に戻る。

【 0 0 5 8 】

また、ドラッグ情報取得アプリケーションは、例えばアプリケーション終了時などに、通知要求を削除することができる。

【 0 0 5 9 】

20

まず、S 5 1 0 で、ドラッグ情報取得アプリケーションは通知要求の削除を要求（削除要求）を OS に対して送信する。OS は削除要求を受信する（削除要求受信手段）と、S 5 1 1 で、この削除要求を送信したドラッグ情報取得アプリケーションに対応する通知要求を R A M 1 0 8 から削除する。最後に、S 5 1 2 で、ドラッグ情報取得アプリケーションは終了に必要な終了処理を行う。

【 0 0 6 0 】

なお、通知要求は OS 管理しており、ドラッグ情報取得アプリケーションの終了を OS は知ることができるので、ドラッグ情報取得アプリケーションの終了に合わせて OS が自動的に通知要求を削除してもよい。また、ドラッグ情報取得アプリケーションは、図 5 に示すタイミング以外の任意のタイミングで削除要求を送信することができる。

30

【 0 0 6 1 】

以上の処理により、ウインドウ 4 0 3 のアプリケーションなどのように、ドラッグ情報を必要とするアプリケーションのみがドラッグ情報を取得することができる。

【 0 0 6 2 】

従って、ドラッグ情報がより効率的に伝達され、P C 1 0 0 の処理負荷が軽減される。また、マウスカーソルがドラッグ情報を必要とするアプリケーションのウインドウ上に無い状態でも、ドラッグ情報の取得が可能である。

【 0 0 6 3 】

なお、本実施例では S 5 0 6 - S 5 0 9 においてループ処理を行っているが、イベント取得の情報（通知要求）をチェーンにしてドラッグ情報を通知したり、通知要求ごとにスレッド化してドラッグ情報を通知したりしてもよい。

40

【実施例 3】

【 0 0 6 4 】

実施例 3 でも、実施例 1 の P C 1 0 0 （図 1 参照）を利用する。

【 0 0 6 5 】

図 6 は、実施例 3 におけるアプリケーションの動作の概要を示す図である。図 6 において、画面 6 0 0 は、OS によって C R T 1 0 1 に表示される表示画面である。ウインドウ 6 0 1 は、アプリケーションのウインドウであり、ドラッグ対象のオブジェクト 6 0 4 （例えば、ファイルアイコン）を含む。ウインドウ 6 0 2 は、別のアプリケーションのウインドウである。

50

## 【 0 0 6 6 】

画面 6 0 0 において、ウインドウ 6 0 2 のアプリケーションは矢印 6 0 3 に示すように、ドラッグを開始したウインドウ 6 0 1 にドラッグ情報を問い合わせることにより、ドラッグ情報を取得できる。

## 【 0 0 6 7 】

図 7 は、実施例 3 におけるアプリケーションの動作の流れを示すフローチャートである。

## 【 0 0 6 8 】

S 7 0 1 で、ドラッグ情報取得アプリケーション（ウインドウ 6 0 2 のアプリケーション）は、他のプログラムであるドラッグ開始アプリケーション（ウインドウ 6 0 1 のアプリケーション）においてオブジェクト 6 0 4 がドラッグされたことを検出する。

10

## 【 0 0 6 9 】

S 7 0 2 で、ドラッグ情報取得アプリケーションは、S 7 0 1 で検出したアプリケーションに対してドラッグ情報を要求し、S 7 0 3 で、ドラッグ開始アプリケーションは要求に応じてドラッグ情報を返信する。

## 【 0 0 7 0 】

本実施例では、OS で規定されているウィンドウメッセージを利用してドラッグ情報取得アプリケーションとドラッグ開始アプリケーションとが通信することができる。或いは、セマフォなどの仕組みを利用して構築したプロセス間通信の技術を利用して通信が実現されてもよい。

20

## 【 0 0 7 1 】

以上の処理により、ウインドウ 6 0 2 のアプリケーションなどのように、ドラッグ情報を必要とするアプリケーションのみがドラッグ情報を取得することができる。

## 【 0 0 7 2 】

従って、ドラッグ情報がより効率的に伝達され、P C 1 0 0 の処理負荷が軽減される。また、マウスカーソルがドラッグ情報を必要とするアプリケーションのウインドウ上に無い状態でも、ドラッグ情報の取得が可能である。

## 【 0 0 7 3 】

[ その他の実施例 ]

上述した各実施例の機能を実現するためには、各機能を実現化したソフトウェアのプログラムコードを記録した記録媒体をシステム或は装置に提供してもよい。そして、そのシステム或は装置のコンピュータ（又は C P U や M P U ）が記録媒体に格納されたプログラムコードを読み出し実行することによって、上述した各実施例の機能が実現される。この場合、記録媒体から読み出されたプログラムコード自体が上述した各実施例の機能を実現することになり、そのプログラムコードを記録した記録媒体は本発明を構成することになる。このようなプログラムコードを供給するための記録媒体としては、例えば、フロッピー（登録商標）ディスク、ハードディスク、光ディスク、光磁気ディスクなどを用いることができる。或いは、C D - R O M 、 C D - R 、 磁気テープ、不揮発性のメモリカード、R O M などを用いることもできる。

30

## 【 0 0 7 4 】

また、上述した各実施例の機能を実現するための構成は、コンピュータが読み出したプログラムコードを実行することだけには限られない。そのプログラムコードの指示に基づき、コンピュータ上で稼動している O S （オペレーティングシステム）などが実際の処理の一部又は全部を行い、その処理によって上述した各実施例の機能が実現される場合も含まれている。

40

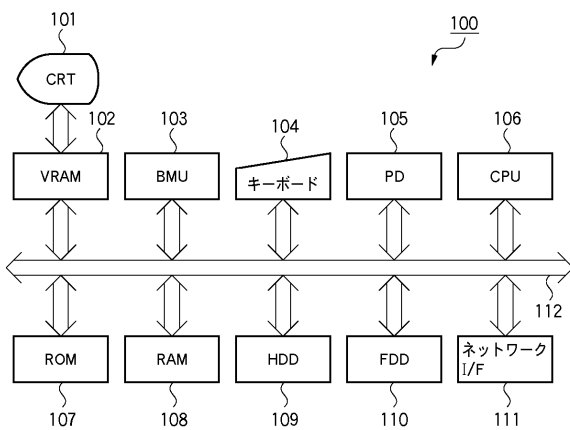
## 【 0 0 7 5 】

更に、記録媒体から読み出されたプログラムコードが、コンピュータに挿入された機能拡張ボードやコンピュータに接続された機能拡張ユニットに備わるメモリに書きこまれてもよい。その後、そのプログラムコードの指示に基づき、その機能拡張ボードや機能拡張ユニットに備わる C P U などが実際の処理の一部又は全部を行い、その処理によって上述

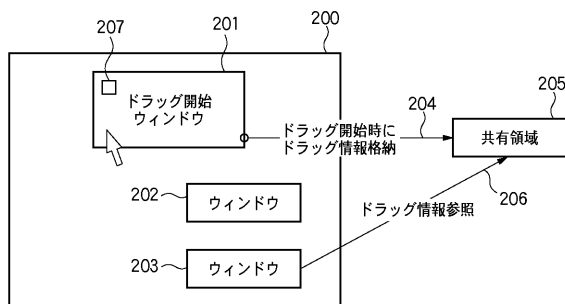
50

した各実施例の機能が実現される場合も含むものである。

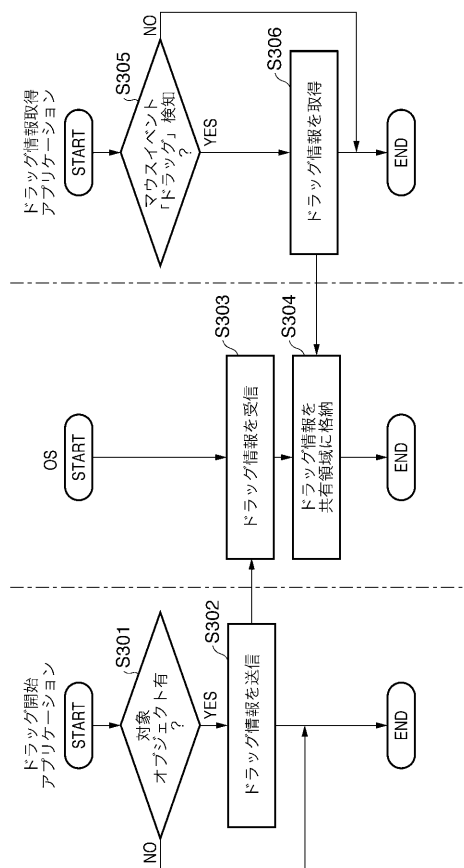
【図 1】



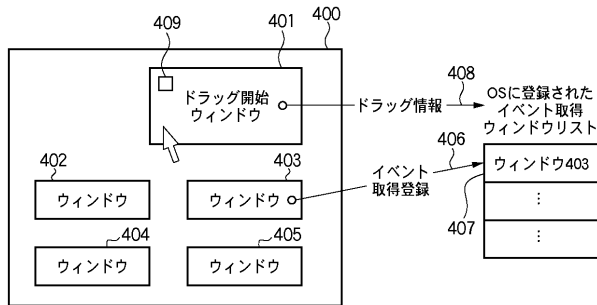
【図 2】



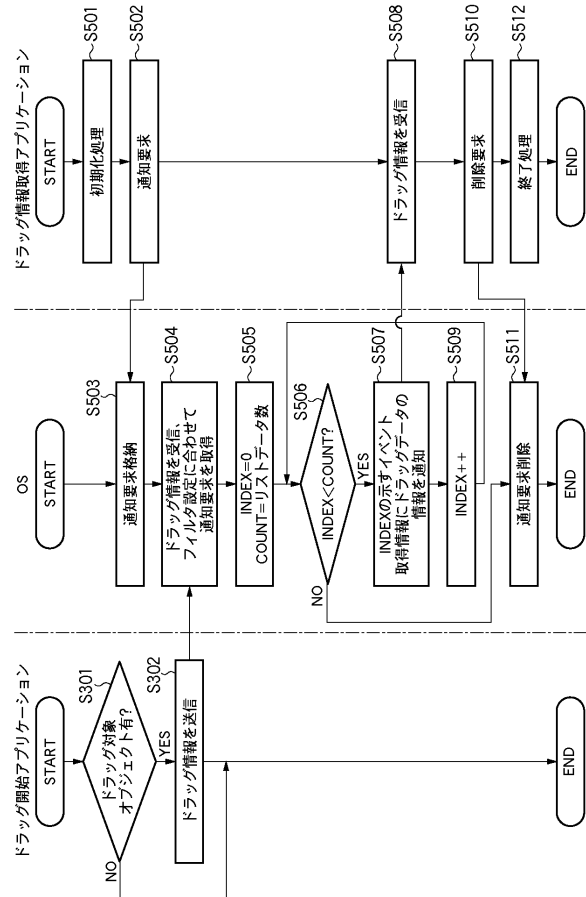
【図 3】



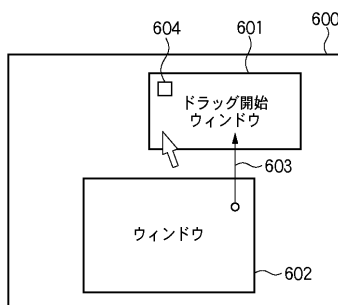
【図 4】



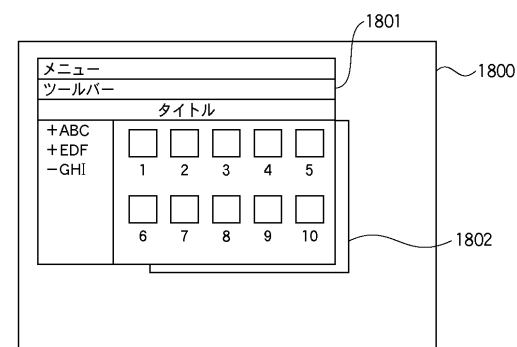
【図 5】



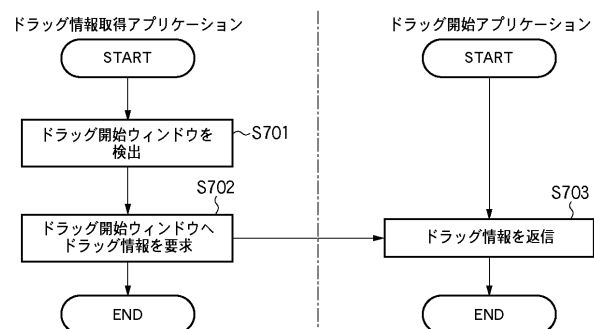
【図 6】



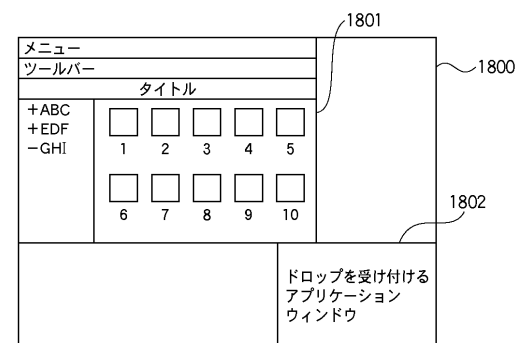
【図 8】



【図 7】



【図 9】



背景技術

背景技術

---

フロントページの続き

(72)発明者 井桁 敏

東京都大田区下丸子3丁目30番2号 キヤノン株式会社内

Fターム(参考) 5E501 AA02 AC37 BA20 DA05 EA08 EB01 EB05 FA02 FA23 FA45