

(19) 日本国特許庁 (JP)

(12) 特 許 公 報 (B2)

(11) 特許番号

特許第5179171号
(P5179171)

(45) 発行日 平成25年4月10日 (2013. 4. 10)

(24) 登録日 平成25年1月18日 (2013. 1. 18)

(51) Int. Cl.

F I

G 0 6 F 3/048 (2013.01)

G 0 6 F 3/048 6 5 5 A

請求項の数 13 (全 29 頁)

(21) 出願番号	特願2007-341475 (P2007-341475)	(73) 特許権者	301015956
(22) 出願日	平成19年12月28日 (2007. 12. 28)		キャノンソフトウェア株式会社
(65) 公開番号	特開2009-163458 (P2009-163458A)		東京都品川区東品川二丁目4番11号
(43) 公開日	平成21年7月23日 (2009. 7. 23)	(74) 代理人	100145827
審査請求日	平成22年12月1日 (2010. 12. 1)		弁理士 水垣 親房
		(72) 発明者	永松 ともえ
			東京都港区三田3丁目9番7号 キャノン
			ソフトウェア株式会社内
		(72) 発明者	藤崎 朝也
			東京都港区三田3丁目9番7号 キャノン
			ソフトウェア株式会社内
		(72) 発明者	内海 愛美
			東京都港区三田3丁目9番7号 キャノン
			ソフトウェア株式会社内

最終頁に続く

(54) 【発明の名称】 情報処理装置、情報処理方法、プログラム、記録媒体

(57) 【特許請求の範囲】

【請求項 1】

ファイルを表示するウィンドウを複数表示可能な情報処理装置であって、
複数の前記ウィンドウのうち、基準となる第1のウィンドウと第2のウィンドウを関連
付けるウィンドウ関連付け手段と、

前記第1のウィンドウに関連付けられている前記第2のウィンドウの表示状態を記憶す
る表示状態記憶手段と、

前記第1のウィンドウがアクティブな状態になった場合に、前記第1のウィンドウに関
連付けられている前記第2のウィンドウの表示状態を、前記表示状態記憶手段に記憶され
ている前記表示状態に基づいて復元する表示状態復元手段と、を備え、

前記表示状態には、前記第2のウィンドウのサイズ、該第2のウィンドウの表示位置を
含むものであって、

前記表示状態記憶手段は、

前記第2のウィンドウが異なる複数の前記基準となるウィンドウから関連付けられてい
る場合には、該基準となるウィンドウ毎に前記第2のウィンドウの前記表示状態を記憶す
るものであって、

前記表示状態復元手段は、

前記第2のウィンドウが異なる複数の前記基準となるウィンドウから関連付けられてい
る場合には、前記表示状態記憶手段に記憶されている複数の前記表示状態のうち、アクテ
ィブな状態になった前記基準となるウィンドウに対応する1の該表示状態に基づいて、該

10

20

第 2 のウィンドウの表示状態を復元することを特徴とする情報処理装置。

【請求項 2】

前記表示状態記憶手段は、前記第 1 のウィンドウがアクティブな状態でアクティブウィンドウの切り替えがあったタイミングで、当該第 1 のウィンドウに関連付けられている前記第 2 のウィンドウの当該第 1 のウィンドウに対応する表示状態を、更新することを特徴とする請求項 1 に記載の情報処理装置。

【請求項 3】

前記表示状態には、前記第 2 のウィンドウに表示されるファイルのいずれの位置を、該第 2 のウィンドウに表示するかを示す表示位置を含み、

前記表示状態復元手段は、前記第 1 のウィンドウがアクティブな状態になった場合に、前記第 2 のウィンドウに表示される前記ファイルの表示位置を、前記表示状態に記憶された前記表示位置にすることを特徴とする請求項 1 又は請求項 2 のいずれか 1 項に記載の情報処理装置。

【請求項 4】

前記表示状態には、前記第 2 のウィンドウの編集中位置を示すカーソル位置を含み、

前記表示状態復元手段は、前記第 1 のウィンドウがアクティブな状態になった場合に、前記第 2 のウィンドウのカーソル位置を、前記表示状態に記憶された前記カーソル位置にすることを特徴とする請求項 1 乃至請求項 3 のいずれか 1 項に記載の情報処理装置。

【請求項 5】

前記第 2 のウィンドウは、前記第 1 のウィンドウをクローズする際に、前記第 2 のウィンドウが、他の前記第 1 のウィンドウに関連付けられていない場合には、前記第 2 のウィンドウをクローズし、一方、前記第 2 のウィンドウが、他の前記第 1 のウィンドウに関連付けられている場合には、前記第 2 のウィンドウをクローズしないように制御する制御手段を有することを特徴とする請求項 1 乃至請求項 4 のいずれか 1 項に記載の情報処理装置。

【請求項 6】

前記表示状態には、前記第 1 のウィンドウに表示されたファイルのファイルパスと、前記第 1 のウィンドウに関連付けられた前記第 2 のウィンドウに表示されたファイルのファイルパスを、関連ファイルとして含むことを特徴とする請求項 1 乃至請求項 5 のいずれか 1 項に記載の情報処理装置。

【請求項 7】

前記表示状態復元手段は、ファイルのオープン時に、該オープンするファイルの前記関連ファイルがオープン済みでない場合には、前記関連ファイルをオープンすることを特徴とする請求項 6 に記載の情報処理装置。

【請求項 8】

操作者からの関連付け指示に応じて、特定のウィンドウを前記基準となる第 1 のウィンドウとし、他のウィンドウを前記第 1 のウィンドウに関連付けられる前記第 2 のウィンドウとする、操作者関連付け手段を有することを特徴とする請求項 1 乃至請求項 7 のいずれか 1 項に記載の情報処理装置。

【請求項 9】

特定のウィンドウと他のウィンドウとの操作履歴に基づいて、前記特定のウィンドウを前記基準となる第 1 のウィンドウとし、前記他のウィンドウを前記関連付けられる第 2 のウィンドウとする、操作履歴関連付け手段を有することを特徴とする請求項 1 乃至請求項 8 のいずれか 1 項に記載の情報処理装置。

【請求項 10】

前記第 1 のウィンドウ又は前記第 1 のウィンドウを示すアイコンの表示形態を、前記第 1 のウィンドウに関連付けられた前記第 2 のウィンドウの表示形態と異ならせるように制御する表示形態制御手段とを有すること特徴とする請求項 1 乃至請求項 9 のいずれか 1 項に記載の情報処理装置。

【請求項 11】

ファイルを表示するウィンドウを複数表示可能な情報処理方法であって、
ウィンドウ関連付け手段が、複数の前記ウィンドウのうち、基準となる第 1 のウィンドウと第 2 のウィンドウを関連付けるウィンドウ関連付けステップと、

表示状態登録手段が、前記第 1 のウィンドウに関連付けられている前記第 2 のウィンドウの表示状態を表示状態記憶手段に記憶させる表示状態記憶ステップと、

表示状態復元手段が、前記第 1 のウィンドウがアクティブな状態になった場合に、前記第 1 のウィンドウに関連付けられている前記第 2 のウィンドウの表示状態を、前記表示状態記憶手段に記憶されている前記表示状態に基づいて復元する表示状態復元ステップと、
を備え、

前記表示状態には、前記第 2 のウィンドウのサイズ、前記第 2 のウィンドウの表示位置を含むものであって、

前記表示状態記憶ステップでは、

前記第 2 のウィンドウが異なる複数の前記基準となるウィンドウから関連付けられている場合には、該基準となるウィンドウ毎に前記第 2 のウィンドウの前記表示状態を前記表示状態記憶手段に記憶させるものであって、

前記表示状態復元ステップは、

前記第 2 のウィンドウが異なる複数の前記基準となるウィンドウから関連付けられている場合には、前記表示状態記憶手段に記憶されている複数の前記表示状態のうち、アクティブな状態になった前記基準となるウィンドウに対応する 1 の該表示状態に基づいて、該第 2 のウィンドウの表示状態を復元することを特徴とする情報処理方法。

【請求項 1 2】

ファイルを表示するウィンドウを複数表示可能なコンピュータを、
複数の前記ウィンドウのうち、基準となる第 1 のウィンドウと第 2 のウィンドウを関連付けるウィンドウ関連付け手段、

前記第 1 のウィンドウに関連付けられている前記第 2 のウィンドウの表示状態を表示状態記憶手段に記憶させる表示状態登録手段、

前記第 1 のウィンドウがアクティブな状態になった場合に、前記第 1 のウィンドウに関連付けられている前記第 2 のウィンドウの表示状態を、前記表示状態記憶手段に記憶されている前記表示状態に基づいて復元する表示状態復元手段として機能させるプログラムであり、

前記表示状態には、前記第 2 のウィンドウのサイズ、前記第 2 のウィンドウの表示位置を含むものであって、

前記表示状態登録手段は、

前記第 2 のウィンドウが異なる複数の前記基準となるウィンドウから関連付けられている場合には、該基準となるウィンドウ毎に前記第 2 のウィンドウの前記表示状態を前記表示状態記憶手段に記憶させるものであって、

前記表示状態復元手段は、

前記第 2 のウィンドウが異なる複数の前記基準となるウィンドウから関連付けられている場合には、前記表示状態記憶手段に記憶されている複数の前記表示状態のうち、アクティブな状態になった前記基準となるウィンドウに対応する 1 の該表示状態に基づいて、該第 2 のウィンドウの表示状態を復元することを特徴とするプログラム。

【請求項 1 3】

請求項 1 2 に記載されたプログラムを記録したコンピュータ読み取り可能な記録媒体。

【発明の詳細な説明】

【技術分野】

【0001】

本発明は、表示装置の画面上に複数のウィンドウを表示するマルチウィンドウ表示制御装置に関し、特に、関連するウィンドウを参照しながら編集、操作を行うような場面で、ウィンドウを切り替える際の関連ウィンドウの参照支援制御に関する。

【背景技術】

【 0 0 0 2 】

近年のパーソナルコンピュータでは、同時に複数のウィンドウを表示可能なマルチウィンドウシステムが採用されている。このマルチウィンドウシステム環境下において、操作者は、複数の関連するウィンドウを参照しながら、編集作業等を行うことが可能である。

【 0 0 0 3 】

このような場面で、表示されている複数のウィンドウの中から、編集作業中のウィンドウの関連ウィンドウを探し出すのは、操作者にとって煩わしい作業である。また、ウィンドウを切り替えるたびにそのような作業を行うのは、操作者にとって大変負荷が大きい。

【 0 0 0 4 】

そのような課題に対し、特許文献 1 では、最上位のウィンドウに対する予め関連付けられたウィンドウを関連しないウィンドウよりも上位に表示することにより、ウィンドウ切り替えに関わる操作者の負担を軽減し、操作性を向上する技術が開示されている。

【特許文献 1】特開 2 0 0 6 - 1 3 9 7 7 9 号公報

【発明の開示】

【発明が解決しようとする課題】

【 0 0 0 5 】

しかしながら、上記特許文献 1 に記載の技術では、関連するウィンドウを他のウィンドウよりも上位に表示し、優先順位順にしているが、関連ウィンドウのレイアウトや参照位置までは考慮していない。

【 0 0 0 6 】

そのため、編集ウィンドウによって、関連ウィンドウの参照位置を変えたいような場合には対応できない。

【 0 0 0 7 】

例えば、同一のウィンドウが複数の編集ウィンドウに関連付けられており、編集ウィンドウによって、その関連ウィンドウの参照したい位置が異なる場合、ユーザは、ウィンドウを切り替えるたびに、その関連ウィンドウのレイアウトを調節し、ページ移動やスクロール移動などを行って参照したい位置を探し出す必要がある。

【 0 0 0 8 】

即ち、従来では、ウィンドウ切り替え時にそれに関連するウィンドウのレイアウトや参照位置を復元する際の操作者の負荷が高く、関連ウィンドウの参照作業の効率が悪いものとなっていたといった課題があった。

【 0 0 0 9 】

本発明は、上記の問題点を解決するためになされたものである。本発明の目的は、ウィンドウ切り替え時の操作者の負荷を軽減し、関連ウィンドウの参照作業の効率を向上させることができる仕組みを提供することである。

【課題を解決するための手段】

【 0 0 1 0 】

本発明は、ファイルを表示するウィンドウを複数表示可能な情報処理装置であって、複数の前記ウィンドウのうち、基準となる第 1 のウィンドウと第 2 のウィンドウを関連付けるウィンドウ関連付け手段と、前記第 1 のウィンドウに関連付けられている前記第 2 のウィンドウの表示状態を記憶する表示状態記憶手段と、前記第 1 のウィンドウがアクティブな状態になった場合に、前記第 1 のウィンドウに関連付けられている前記第 2 のウィンドウの表示状態を、前記表示状態記憶手段に記憶されている前記表示状態に基づいて復元する表示状態復元手段と、を備え、前記表示状態には、前記第 2 のウィンドウのサイズ、該第 2 のウィンドウの表示位置を含むものであって、前記表示状態記憶手段は、前記第 2 のウィンドウが異なる複数の前記基準となるウィンドウから関連付けられている場合には、該基準となるウィンドウ毎に前記第 2 のウィンドウの前記表示状態を記憶するものであって、前記表示状態復元手段は、前記第 2 のウィンドウが異なる複数の前記基準となるウィンドウから関連付けられている場合には、前記表示状態記憶手段に記憶されている複数の前記表示状態のうち、アクティブな状態になった前記基準となるウィンドウに対応する 1

10

20

30

40

50

の該表示状態に基づいて、該第2のウィンドウの表示状態を復元することを特徴とする。

【発明の効果】

【0011】

本発明によれば、ウィンドウが切り替わる等の状況の変化に合わせて、それに関連するウィンドウの表示状態も復元することで、ウィンドウ切り替え時の操作者の負荷を軽減し、関連ウィンドウの参照作業の効率を向上させることができる。

【発明を実施するための最良の形態】

【0012】

以下、図面を参照して、本発明の実施形態を詳細に説明する。

【0013】

10

〔第1実施形態〕

図1は、本発明の一実施形態を示すマルチウィンドウ表示装置の構成の一例を示すブロック図である。

【0014】

図1に示すように、101はOS(オペレーティングシステム)であり、ファイル入出力、メモリ管理、入出力装置管理、アプリケーション実行管理などパーソナルコンピュータの機能として基本的な機能を実現するものである。

【0015】

102はウィンドウ制御部であり、OS101に含まれる機能のうち、直接的なウィンドウ制御全般を行うものである。

20

【0016】

103は監視部であり、ウィンドウの監視を行い、ウィンドウ情報テーブル104を参照しながら、状態復元部106、状態記憶部107、関連付け部108に対し、状況に応じたメッセージを送る。

【0017】

ウィンドウ情報テーブル104は、表示中のウィンドウのハンドル、ファイルパス、関連ウィンドウ情報配列105の参照ポインタ等、表示中のウィンドウ情報を保持する。

【0018】

関連ウィンドウ情報配列105は、アクティブウィンドウに関連するウィンドウの情報を保持する。

30

【0019】

なお、ウィンドウ情報テーブル104、関連ウィンドウ情報配列105は、プログラム実行時に、後述する図2に示すRAM203上に作成されるテーブルである。また、ウィンドウ情報テーブル104、関連ウィンドウ情報配列105の一例を後述する図5に示す。

【0020】

状態復元部106は、監視部103から指令を受けて、アクティブウィンドウに関連するウィンドウの状態復元を行う部分である。

【0021】

状態記憶部107は、監視部103から指令を受けて、アクティブウィンドウに関連するウィンドウの状態の記憶処理を行う部分である。

40

【0022】

関連付け部108は、監視部103から指令を受けて、ウィンドウ同士の関連付けを行う部分である。

【0023】

なお、監視部103、状態復元部106、状態記憶部107、関連付け部108は、CPU201が、後述する図2に示す外部メモリ211からプログラムをRAM203にロードして実行することで実現される機能に対応する。

【0024】

次に、ファイル管理DB109、関連ファイル管理DB110について説明する。なお

50

、ファイル管理DB109, 関連ファイル管理DB110は、第1実施形態の説明では用いず、後述する第2実施形態の説明において用いる。

【0025】

図2は、図1に示したマルチウィンドウ表示装置に適用可能なコンピュータ(情報処理装置)のハードウェア構成の一例を示すブロック図である。

【0026】

図2において、201はCPUであり、システムバス204に接続される各デバイスやコントローラを統括的に制御する。また、ROM202あるいは外部メモリ211には、CPU201の制御プログラムであるBIOS(Basic Input / Output System)やオペレーティングシステムプログラム(以下、OS)や、各サーバ或いは各PCの実行する機能を実現するために必要な後述する各種プログラム等が記憶されている。

10

【0027】

203はRAMであり、CPU201の主メモリ、ワークエリア等として機能する。CPU201は、処理の実行に際して必要なプログラム等を外部メモリ211からRAM203にロードして、プログラムを実行することで各種動作を実現するものである。

【0028】

205は入力コントローラ(入力C)であり、キーボード209や不図示のマウス等のポインティングデバイスからの入力を制御する。

【0029】

206はビデオコントローラ(VC)であり、CRTディスプレイ(CRT)210等の表示器にへの表示を制御する。表示器はCRTだけでなく、液晶ディスプレイでも構わない。これらは必要に応じて管理者が使用するものである。本発明には直接関係があるものではない。

20

【0030】

207はメモリコントローラ(MC)であり、ブートプログラム、ブラウザソフトウェア、各種のアプリケーション、フォントデータ、ユーザファイル、編集ファイル、各種データ等を記憶する外部メモリ211へのアクセスを制御する。なお、外部メモリ211は、一例として、ハードディスク(HD)やフロッピー(登録商標)ディスク(FD)或いはPCMCIAカードスロットにアダプタを介して接続されるコンパクトフラッシュ(登録商標)メモリ等が該当する。

30

【0031】

208は通信I/Fコントローラ(通信I/F C)であり、ネットワークを介して、外部機器と接続・通信するものであり、ネットワークでの通信制御処理を実行する。この通信処理としては、例えば、TCP/IPを用いたインターネット通信等が可能である。

【0032】

なお、CPU201は、例えばRAM203内の表示情報用領域へアウトラインフォントの展開(ラスターライズ)処理を実行することにより、CRT210上での表示を可能としている。また、CPU201は、CRT210上の不図示のマウスカーソル等でのユーザ指示を可能とする。

【0033】

40

本発明を実現するための関連ウィンドウ参照支援プログラムは、外部メモリ211に記録されており、必要に応じてRAM203にロードされることによりCPU201によって実行されるものである。即ち、関連ウィンドウ参照支援プログラムは、CPU201を、図1に示した監視部103、状態復元部106、状態記憶部107、関連付け部108として機能させるためのプログラムである。さらに、関連ウィンドウ参照支援プログラムが用いる定義ファイル213及び各種情報テーブル214は外部メモリ211に格納されており、これらについての詳細な説明は後述する。

【0034】

次に、関連ウィンドウ参照支援装置の基本的な処理について、図3～図8を用いて説明する。

50

【 0 0 3 5 】

図 3 は、本発明における第 1 の制御処理手順の一例を示すフローチャートであり、第 1 実施形態におけるマルチウィンドウ表示装置における関連ウィンドウ参照支援処理に対応する。なお、このフローチャートの処理は、図 1 に示した監視部 1 0 3、状態復元部 1 0 6、状態記憶部 1 0 7、関連付け部 1 0 8 により実行されるものである。即ち、外部メモリ 2 1 1 に記録された関連ウィンドウ参照支援プログラムを、CPU 2 0 1 が、RAM 2 0 3 にロードして実行することにより実現される。

【 0 0 3 6 】

まず、ステップ S 3 0 1 において、監視部 1 0 3 は、ウィンドウ制御部 1 0 2 からアクティブウィンドウ切り替え通知を受け取ると、ステップ S 3 0 2 に処理を進める。即ち、この通知がステップ S 3 0 2 に進むトリガーとなる。

10

【 0 0 3 7 】

ステップ S 3 0 2 では、監視部 1 0 3 は、ウィンドウが切り替わる前のアクティブウィンドウ（以下、旧アクティブウィンドウ）のハンドルをウィンドウ制御部 1 0 2 より取得する。

【 0 0 3 8 】

次に、ステップ S 3 0 3 において、監視部 1 0 3 は、S 3 0 2 で取得した旧アクティブウィンドウのハンドルを元にウィンドウ情報テーブル 1 0 4（図 4）を検索する。

【 0 0 3 9 】

図 4 は、ウィンドウ情報テーブル 1 0 4、関連ウィンドウ情報配列 1 0 5 の一例を示す図である。なお、ウィンドウ情報テーブル 1 0 4、関連ウィンドウ情報配列 1 0 5 は、プログラム実行時に RAM 2 0 3 で保持される情報である。

20

【 0 0 4 0 】

まず、ウィンドウ情報テーブル 1 0 4 のデータ構造について説明する。

【 0 0 4 1 】

ウィンドウ情報テーブル 1 0 4 は、最小化されているものも含めて、その時点で表示されている全てのウィンドウの情報を持っている。以下、各項目について説明する。

【 0 0 4 2 】

ウィンドウハンドル 1 0 0 1 は、表示しているウィンドウのハンドルをあらわしている。このハンドルにより、ウィンドウ制御部 1 0 2 からの基本的なウィンドウ情報の取得、ウィンドウの制御が可能となる。

30

【 0 0 4 3 】

ファイルパス・URL 1 0 0 2 は、そのウィンドウで表示しているファイルのパスを表すものである。この例では、ファイルパスもしくは URL としているが、そのほかにもエイリアス等のファイルの識別に用いられるものは何でも適用可である。以下、各テーブルの「ファイルパス・URL」についても同様である。

【 0 0 4 4 】

関連ウィンドウ表示機能 1 0 0 3 は、そのウィンドウの関連ウィンドウを表示するか否かを表すものである。そのステータスには、「ON」、「OFF」がある。「ON」の場合は、ウィンドウの関連付け、関連ウィンドウの記憶、復元処理を行う。「OFF」の場合は、以上の処理は行わない。

40

【 0 0 4 5 】

ユーザは、後述する関連ウィンドウ表示機能設定画面 1 4 0 0（図 1 4）により関連ウィンドウ表示機能 1 0 0 3 の「ON」、「OFF」を切り替えることができる。

【 0 0 4 6 】

関連ウィンドウ情報配列ポインタ 1 0 0 4 は、関連ウィンドウ情報配列 1 0 5 のポインタを表すものである。

【 0 0 4 7 】

関連ウィンドウ情報配列 1 0 5 は、関連ウィンドウの情報を格納する配列である。以下、各項目について説明する。

50

【 0 0 4 8 】

ウィンドウハンドル 1 0 0 5 は、ファイルを表示しているウィンドウのハンドルをあらわしている。このハンドルにより、ウィンドウ情報の取得や制御が可能となる。ファイルパス・URL 1 0 0 6 は、ウィンドウで表示しているファイルのパスを表すものである。

【 0 0 4 9 】

表示順 1 0 0 7 は、関連元のウィンドウを除いて、スクリーン上で上から何番目に表示するかを表すものである。関連元のウィンドウが最上位に表示され、それに続いて、表示順 1 0 0 7 が「 1 」のウィンドウから順に上位に表示される。

【 0 0 5 0 】

ウィンドウ領域 1 0 0 8 は、表示画面上のウィンドウ位置を示しており、左から順に、X 座標、Y 座標、幅、高さ、となっており、X 座標、Y 座標は、ウィンドウの左上の座標を表している。幅、高さはウィンドウの幅と高さを表している。

【 0 0 5 1 】

参照位置(X、Y) 1 0 0 9 は、ウィンドウにおいて、ファイル内のどの場所が表示されているかを表すものである。左から、X 座標、Y 座標となっており、この例では、スクロールの位置を表している。

【 0 0 5 2 】

以下、図 3 のフローチャートの説明に戻る。

【 0 0 5 3 】

次に、ステップ S 3 0 4 において、監視部 1 0 3 は、S 3 0 3 の検索結果に基づいて、旧アクティブウィンドウの関連ウィンドウ表示機能 1 0 0 3 を調べ、それが「ON」である場合は、S 3 0 5 に処理を進め、一方、「OFF」の場合は、S 3 0 7 に処理を進める。

【 0 0 5 4 】

ステップ S 3 0 5 では、監視部 1 0 3 は、関連付け部 1 0 8 に対し、関連付け処理を行うようメッセージを送る。これにより、関連付け部 1 0 8 は、監視部 1 0 3 からのメッセージを受け、旧アクティブウィンドウの関連付け処理を行う。なお、関連付け処理の詳細は後述する図 5 で示す。

【 0 0 5 5 】

次に、ステップ S 3 0 6 において、監視部 1 0 3 は、状態記憶部 1 0 7 に対し、処理対象のアクティブウィンドウの関連ウィンドウ記憶処理を行うようメッセージを送る。これにより、状態記憶部 1 0 7 は、メッセージを受けて、旧アクティブウィンドウの関連ウィンドウの状態を記憶する処理を行う。なお、関連ウィンドウ記憶処理の詳細は後述する図 6 で示す。

【 0 0 5 6 】

次に、ステップ S 3 0 7 において、監視部 1 0 3 は、ウィンドウ切り替えにより、旧アクティブウィンドウに代わって新しくアクティブになるウィンドウ（以下、新アクティブウィンドウと呼ぶ）のハンドルをウィンドウ制御部 1 0 2 より取得する。

【 0 0 5 7 】

次に、ステップ S 3 0 8 では、監視部 1 0 3 は、S 3 0 7 で取得した新アクティブウィンドウのハンドルをキーにウィンドウ情報テーブル 1 0 4（図 4）を検索する。

【 0 0 5 8 】

次に、ステップ S 3 0 9 において、監視部 1 0 3 は、新アクティブウィンドウの関連ウィンドウ表示機能 1 0 0 3 を調べ、「ON」である場合は、ステップ S 3 1 0 に処理を進める。

【 0 0 5 9 】

ステップ S 3 1 0 では、監視部 1 0 3 は、状態復元部 1 0 6 に対し、新アクティブウィンドウの関連ウィンドウ復元処理を行うようメッセージを送る。これにより、状態復元部 1 0 6 は、メッセージを受けて新アクティブウィンドウに関連するウィンドウの復元処理を行う。なお、関連ウィンドウ復元処理の詳細は、後述する図 8 に示す。そして、ステッ

10

20

30

40

50

プ S 3 0 9 の処理が終了すると、監視部 1 0 3 は、本フローチャートの処理を終了する。

【 0 0 6 0 】

一方、ステップ S 3 0 9 で、新アクティブウィンドウの関連ウィンドウ表示機能 1 0 0 3 が「ON」でないと判定した場合には、監視部 1 0 3 は、本フローチャートの処理を終了する。

【 0 0 6 1 】

次に、図 3 のステップ S 3 0 5 に示した旧アクティブウィンドウの関連付け処理について、図 5 を参照して説明する。

【 0 0 6 2 】

図 5 は、本発明における第 2 の制御処理手順の一例を示すフローチャートであり、図 3 のステップ S 3 0 5 に示した旧アクティブウィンドウの関連付け処理に対応する。なお、このフローチャートの処理は、関連付け部 1 0 8 により実行されるものである。即ち、外部メモリ 2 1 1 に記録された関連ウィンドウ参照支援プログラムを、CPU 2 0 1 が、RAM 2 0 3 にロードして実行することにより実現される。

【 0 0 6 3 】

まず、ステップ S 4 0 1 において、関連付け部 1 0 8 は、ウィンドウ制御部 1 0 2 から旧アクティブウィンドウのイベントを取得する。

【 0 0 6 4 】

次に、ステップ S 4 0 2 において、関連付け部 1 0 8 は、旧アクティブウィンドウのイベントがクローズもしくは、最小化であると判断した場合、ステップ S 4 0 3 に処理を進める。クローズもしくは、最小化以外の場合は、関連付けは行わない。

【 0 0 6 5 】

一方、ステップ S 4 0 2 で、関連付け部 1 0 8 が、旧アクティブウィンドウのイベントがクローズでなく、かつ、最小化でもないと判断した場合、関連付けは行わず、そのまま本フローチャートの処理を終了する。

【 0 0 6 6 】

ステップ S 4 0 3 では、関連付け部 1 0 8 は、旧アクティブウィンドウ以外で、画面上に表示されているウィンドウのウィンドウハンドルをウィンドウ制御部 1 0 2 から取得する。

【 0 0 6 7 】

次に、ステップ S 4 0 4 において、関連付け部 1 0 8 は、ステップ S 4 0 5 ~ S 4 0 9 の処理を、ステップ S 4 0 3 で取得した各ウィンドウに対して繰り返すように制御する。

【 0 0 6 8 】

ステップ S 4 0 5 では、関連付け部 1 0 8 は、S 4 0 3 で取得したウィンドウハンドルを元に旧アクティブウィンドウの関連ウィンドウ情報配列 1 0 5 (図 4) を検索する。

【 0 0 6 9 】

そして、ステップ S 4 0 6 において、関連付け部 1 0 8 は、S 4 0 3 で取得したウィンドウハンドル(以下、処理対象のウィンドウハンドル)が関連ウィンドウ情報配列に存在した場合、処理対象のウィンドウハンドルに対応するウィンドウ(処理対象のウィンドウ)が、旧アクティブウィンドウに関連付けされているか否かを判定する。

【 0 0 7 0 】

ステップ S 4 0 6 で、処理対象のウィンドウハンドルが関連ウィンドウ情報配列に存在していると判定した場合には、関連付け部 1 0 8 は、処理対象のウィンドウが既に旧アクティブウィンドウに関連付けされていると判断し、関連付けは行わず、ステップ S 4 0 5 に処理を戻し、次のウィンドウハンドルに処理対象を移す。

【 0 0 7 1 】

一方、ステップ S 4 0 6 で、処理対象のウィンドウハンドルが関連ウィンドウ情報配列に存在しないと判定した場合には、関連付け部 1 0 8 は、処理対象のウィンドウが旧アクティブウィンドウに関連付けされていないと判断し、ステップ S 4 0 7 に処理を進める。

【 0 0 7 2 】

10

20

30

40

50

ステップS 4 0 7では、関連付け部 1 0 8は、旧アクティブウィンドウとの関連付けが必要か否か判断するために、関連付け確認画面 1 3 0 0（図 6）を表示し、ユーザの回答を求める。

【 0 0 7 3 】

図 6 は、関連付け確認画面 1 3 0 0 の一例を示す図である。

【 0 0 7 4 】

図 6 において、メッセージ 1 3 0 1 に含まれる名前は、関連ウィンドウ表示機能が「ON」となっているアクティブウィンドウのタイトル名を表す。

【 0 0 7 5 】

1 3 0 2 は、メッセージ 1 3 0 1 に含まれるウィンドウの関連ウィンドウ候補のウィンドウタイトル名である。

【 0 0 7 6 】

1 3 0 3 は、OK ボタンであり、ユーザがこのボタンを押すことで、メッセージ 1 3 0 1 に含まれるウィンドウと、ウィンドウタイトル名の示すのウィンドウが関連付けられる。

【 0 0 7 7 】

1 3 0 4 は、キャンセルボタンであり、ユーザがこのボタンを押すことで、メッセージ 1 3 0 1 に含まれるウィンドウと、ウィンドウタイトル名の示すのウィンドウとは関連付けされない。

【 0 0 7 8 】

以下、図 5 のフローチャートの説明に戻る。

【 0 0 7 9 】

ステップS 4 0 7において、関連付け部 1 0 8が、ユーザが関連付け確認画面でキャンセルボタン 1 3 0 4 を押したと判定した場合には、関連付けが必要ない判断する。そして、関連付け部 1 0 8 は、ステップS 4 0 5 に処理を戻し、次のウィンドウハンドルに処理対象を移す。

【 0 0 8 0 】

一方、ステップS 4 0 7において、関連付け部 1 0 8が、ユーザが関連付け確認画面でOK ボタン 1 3 0 3 を押したと判定した場合には、関連付けが必要と判断し、ステップS 4 0 8 に処理を進める。

【 0 0 8 1 】

ステップS 4 0 8 では、関連付け部 1 0 8 は、処理対象のウィンドウハンドル、処理対象のウィンドウハンドルに対応するファイルパス等の関連ウィンドウ情報を作成する。

【 0 0 8 2 】

次に、ステップS 4 0 9 において、関連付け部 1 0 8 は、S 4 0 8 で作成した関連ウィンドウ情報を旧アクティブウィンドウの関連ウィンドウ情報配列 1 0 5 に追加する。そして、関連付け部 1 0 8 は、ステップS 4 0 5 に処理を戻し、次のウィンドウハンドルに処理対象を移す。

【 0 0 8 3 】

以上のステップS 4 0 3 ～S 4 0 9 の処理を、S 4 0 3 で取得した表示ウィンドウに対して繰り返すことにより、旧アクティブウィンドウの関連ウィンドウの表示状態が関連ウィンドウ情報配列 1 0 5 に追加される。

【 0 0 8 4 】

なお、図 5 に示した例では、S 4 0 2 の判定ステップにより、ウィンドウのクローズ及び最小化時に、該ウィンドウ以外に表示されているウィンドウを該ウィンドウの関連付け対象とする関連付け方法をとっている。しかし、関連付け方法はこれに限らず、あらゆる関連付け方法を適用可である。

【 0 0 8 5 】

例えば、関連付けのタイミングは、ウィンドウのクローズ及び最小化時に限定することなく、単にアクティブウィンドウが切り替えられた時に行うように構成してもよい。即ち

10

20

30

40

50

、旧アクティブウィンドウのイベントが「クローズ」もしくは「最小化」であるか否かに関係なく、ステップS 4 0 1の実行後、関連付けは行う（ステップS 4 0 3～S 4 0 9の処理を行う）ように構成してもよい。

【0086】

また、各ウィンドウの関連付けの有無の判定（S 4 0 7）は、図16に示した関連付け確認画面によるユーザ指示に限定するものではない。例えば、ウィンドウに対するユーザの操作履歴（コピー量、参照切り替え回数、参照時間等）を監視し、関連付けデータとして蓄え、その関連付けデータをもとに関連付けを自動的に行う方法をとっても良い。

【0087】

さらに、関連元のウィンドウに対する操作に基づいて、アプリケーションが自動表示したウィンドウは、自動的に、上記関連元のウィンドウに関連付けするように構成してもよい。

10

【0088】

また、関連付け編集用のユーザインタフェースをユーザに提示し、ユーザがウィンドウ同士の関連付けを自由に編集できるようにしても良い。

【0089】

さらに、関連付け部108等が、関連付け元となるウィンドウ、又は、該ウィンドウを示すアイコン（タスクバー上のウィンドウアイコン等）の表示形態（表示色、形状、ウィンドウ名の表示フォント等）を、関連ウィンドウの表示形態と異ならせるように制御する構成であってもよい。

20

【0090】

図7は、本発明における第3の制御処理手順の一例を示すフローチャートであり、図3のステップS 3 0 6に示した旧アクティブウィンドウの関連ウィンドウ記憶処理に対応する。なお、このフローチャートの処理は、状態記憶部107により実行されるものである。即ち、外部メモリ211に記録された関連ウィンドウ参照支援プログラムを、CPU 201が、RAM 203にロードして実行することにより実現される。

【0091】

まず、ステップS 5 0 1において、状態記憶部107は、旧アクティブウィンドウのウィンドウハンドルを元に、ウィンドウ情報テーブル104を検索し、関連ウィンドウ情報配列105を取得する。

30

【0092】

次に、ステップS 5 0 2において、状態記憶部107は、S 5 0 1で取得した関連ウィンドウ情報配列に含まれる関連ウィンドウ全てに対し、ステップS 5 0 3～S 5 0 6の処理を繰り返す。

【0093】

まず、ステップS 5 0 3では、状態記憶部107は、処理対象の関連ウィンドウの参照位置、ウィンドウ領域、表示順等の情報をウィンドウ制御部102から取得する。

【0094】

次に、ステップS 5 0 4において、状態記憶部107は、S 5 0 3で取得した関連ウィンドウの情報で、処理対象の関連ウィンドウの関連ウィンドウ情報配列を上書きする。

40

【0095】

次に、ステップS 5 0 5において、状態記憶部107は、旧アクティブウィンドウのイベントが最小化であったか否かを判定する。

【0096】

ステップS 5 0 5で、旧アクティブウィンドウのイベントが最小化であったと判定した場合には、状態記憶部107は、ステップS 5 0 6に処理を進め、ウィンドウ制御部102にメッセージを送り、処理対象の関連ウィンドウを最小化する。そして、状態記憶部107は、ステップS 5 0 3に処理を戻し、次の関連ウィンドウに処理対象を移す。

【0097】

一方、ステップS 5 0 5で、旧アクティブウィンドウのイベントが最小化でなかったと

50

判定した場合には、状態記憶部 107 は、そのままステップ S503 に処理を戻し、次の関連ウィンドウに処理対象を移す。

【0098】

以上のステップ S503 ~ S506 の処理を、S501 で取得した旧アクティブウィンドウの関連ウィンドウ情報配列に含まれる関連ウィンドウ全てに対して繰り返すことにより、関連ウィンドウ情報配列 105 に記憶される旧アクティブウィンドウの全ての関連ウィンドウの表示状態が更新される。

【0099】

図 8 は、本発明における第 4 の制御処理手順の一例を示すフローチャートであり、図 3 のステップ S310 に示した新アクティブウィンドウの関連ウィンドウ復元処理に対応する。なお、このフローチャートの処理は、状態復元部 106 により実行されるものである。即ち、外部メモリ 211 に記録された関連ウィンドウ参照支援プログラムを、CPU 201 が、RAM 203 にロードして実行することにより実現される。

10

【0100】

まず、ステップ S601 において、状態復元部 106 は、旧アクティブウィンドウに代わって、新たにアクティブになる新アクティブウィンドウのウィンドウハンドルを元に、ウィンドウ情報テーブル 104 を検索し、関連ウィンドウ情報配列を取得する。

【0101】

次に、ステップ S602 において、状態復元部 106 は、S601 で取得した関連ウィンドウ情報配列に含まれる関連ウィンドウ全てに対し、ステップ S603 ~ S604 の処理を繰り返す。

20

【0102】

まず、ステップ S603 において、状態復元部 106 は、関連ウィンドウ情報配列から、処理対象の関連ウィンドウの参照位置 1009、ウィンドウ領域 1008、表示順 1007 の情報（関連ウィンドウ情報）を取得する。

【0103】

次に、ステップ S604 において、状態復元部 106 は、S603 で取得した関連ウィンドウ情報を元に、処理対象の関連ウィンドウの状態を復元する。そして、状態復元部 106 は、ステップ S603 に処理を戻し、次の関連ウィンドウに処理対象を移す。

【0104】

30

以上のステップ S603 ~ S604 の処理を、S601 で取得した新アクティブウィンドウの関連ウィンドウ情報配列に含まれる関連ウィンドウ全てに対して繰り返すことにより、新アクティブウィンドウの全ての関連ウィンドウの状態を復元することができる。

【0105】

〔第 2 実施形態〕

以下、本発明の第 2 実施形態について説明する。

【0106】

本実施形態では、図 1 に示したファイル管理 DB 109、関連ファイル管理 DB 12 を用いる。

【0107】

40

図 1 に示すファイル管理 DB 109 は、ウィンドウに対応する各ファイルのファイルパスや URL、関連ウィンドウ参照支援機能の ON もしくは OFF、関連ファイル ID 等の情報を記憶するデータベースである。このファイル管理 DB 109 は、アクティブウィンドウクローズ時にそのウィンドウで開いていたファイルの情報が記憶され、次回そのファイルがオープンされる時に参照される。

【0108】

関連ファイル管理 DB 110 は、アクティブウィンドウのクローズ時に該アクティブウィンドウの関連ウィンドウの表示状態を記憶しておくデータベースである。この関連ファイル管理 DB 110 は、次回関連ウィンドウオープン時に参照される。

【0109】

50

なお、ファイル管理DB109、関連ファイル管理DB110は、図2に示した外部メモリ211上に実現されるものである。

【0110】

以下、本実施形態の関連ウィンドウ参照支援処理を説明する。

【0111】

図9は、本発明における第5の制御処理手順の一例を示すフローチャートであり、第2実施形態におけるマルチウィンドウ表示装置における関連ウィンドウ参照支援処理に対応する。なお、このフローチャートの処理は、図1に示した監視部103、状態復元部106、状態記憶部107、関連付け部108により実行されるものである。即ち、外部メモリ211に記録された関連ウィンドウ参照支援プログラムを、CPU201が、RAM203にロードして実行することにより実現される。

10

【0112】

まず、ステップS701において、監視部103は、ウィンドウ制御部102からアクティブウィンドウ切り替え通知を受け取ると、ステップS702に処理を進める。即ち、この通知がステップS702に進むトリガーとなる。

【0113】

ステップS702では、監視部103は、ウィンドウが切り替わる前のアクティブウィンドウ（以下、旧アクティブウィンドウ）のハンドルをウィンドウ制御部102より取得する。

【0114】

20

次に、ステップS703において、監視部103は、S702で取得した旧アクティブウィンドウのハンドルを元にウィンドウ情報テーブル104（図4）を検索する。

【0115】

次に、ステップS704において、監視部103は、S703の検索結果に基づいて、旧アクティブウィンドウの関連ウィンドウ表示機能1003を調べ、それが「ON」である場合は、S705に処理を進め、一方、「OFF」の場合は、S706に処理を進める。

【0116】

ステップS705では、監視部103は、関連付け部108に対し、関連付け処理を行うようメッセージを送る。これにより、関連付け部108は、監視部103からのメッセージを受け、旧アクティブウィンドウの関連付け処理を行う（図5）。そして、ステップS705の処理が終了すると、監視部103は、ステップS706に処理を進める。

30

【0117】

ステップS706では、監視部103は、ウィンドウ制御部102より、旧アクティブウィンドウのウィンドウイベントを取得する。

【0118】

次に、ステップS707において、監視部103は、旧アクティブウィンドウのイベントがクローズか否かを判定する。そして、旧アクティブウィンドウのイベントがクローズと判断した場合は、監視部103は、ステップS708に処理を進める。

【0119】

40

ステップS708では、監視部103は、状態記憶部107に対し、旧アクティブウィンドウのクローズに伴う処理を行うようメッセージを送る。これにより、状態記憶部107は、メッセージを受けて旧アクティブウィンドウのクローズに伴う処理を行う。なお、ウィンドウのクローズに伴う処理の詳細は後述する図10に示す。そして、ステップS708の処理が終了すると、監視部103は、ステップS711に処理を進める。

【0120】

一方、ステップS707において、旧アクティブウィンドウのイベントがクローズでないと判定した場合には、監視部103は、ステップS709に処理を進める。

【0121】

ステップS709では、監視部103は、S703の検索結果に基づいて、旧アクティ

50

ブウィンドウの関連ウィンドウ表示機能 1 0 0 3 を調べ、それが「ON」である場合は、S 7 1 0 に処理を進め、一方、「OFF」の場合は、S 7 1 1 に処理を進める。

【0 1 2 2】

ステップ S 7 1 0 では、監視部 1 0 3 は、状態記憶部 1 0 7 に対し、処理対象のアクティブウィンドウの関連ウィンドウ記憶処理を行うようメッセージを送る。これにより、状態記憶部 1 0 7 は、メッセージを受けて、旧アクティブウィンドウの関連ウィンドウの状態を記憶する処理を行う（図 7）。そして、ステップ S 7 1 1 の処理が終了すると、監視部 1 0 3 は、ステップ S 7 1 1 に処理を進める。

【0 1 2 3】

ステップ S 7 1 1 では、監視部 1 0 3 は、ウィンドウ切り替えにより、旧アクティブウィンドウに代わって新しくアクティブになるウィンドウ（以下、新アクティブウィンドウと呼ぶ）のハンドルをウィンドウ制御部 1 0 2 より取得する。

【0 1 2 4】

次に、ステップ S 7 1 2 では、監視部 1 0 3 は、S 7 1 1 で取得した新アクティブウィンドウのハンドルをキーにウィンドウ情報テーブル 1 0 4（図 4）を検索する。

【0 1 2 5】

次に、ステップ S 7 1 3 において、監視部 1 0 3 は、新アクティブウィンドウのウィンドウハンドルがウィンドウ情報テーブル 1 0 4 に存在するか否か判定し、存在すると判定した場合には、そのままステップ S 7 1 4 に処理を進める。

【0 1 2 6】

ステップ S 7 1 4 では、監視部 1 0 3 は、新アクティブウィンドウの関連ウィンドウ表示機能 1 0 0 3 を調べ、「ON」である場合は、ステップ S 7 1 5 に処理を進める。

【0 1 2 7】

ステップ S 7 1 5 では、監視部 1 0 3 は、状態復元部 1 0 6 に対し、新アクティブウィンドウの関連ウィンドウ復元処理を行うようメッセージを送る。これにより、状態復元部 1 0 6 は、メッセージを受けて新アクティブウィンドウに関連するウィンドウの復元処理を行う（図 8）。そして、ステップ S 7 1 5 の処理が終了すると、監視部 1 0 3 は、本フローチャートの処理を終了する。

【0 1 2 8】

一方、ステップ S 7 1 4 は、新アクティブウィンドウの関連ウィンドウ表示機能 1 0 0 3 が「ON」でないと判定した場合には、監視部 1 0 3 は、本フローチャートの処理を終了する。

【0 1 2 9】

また、ステップ S 7 1 3 で、新アクティブウィンドウのウィンドウハンドルがウィンドウ情報テーブル 1 0 4 に存在しないと判定した場合には、ステップ S 7 1 6 に処理を進める。

【0 1 3 0】

ステップ S 7 1 6 では、監視部 1 0 3 は、状態復元部 1 0 6 に対し、新アクティブウィンドウのオープンに伴う処理を行うようメッセージを送る。これにより、状態復元部 1 0 6 は、メッセージを受けて新アクティブウィンドウのオープンに伴う処理を行う。なお、ウィンドウのオープンに伴う処理の詳細は、後述する図 1 0 に示す。そして、ステップ S 7 1 6 の処理が終了すると、監視部 1 0 3 は、本フローチャートの処理を終了する。

【0 1 3 1】

次に、図 9 のステップ S 7 0 8 に示した旧アクティブウィンドウのクローズに伴う処理について、図 1 0 を参照して説明する。

【0 1 3 2】

図 1 0 は、本発明における第 6 の制御処理手順の一例を示すフローチャートであり、図 9 のステップ S 7 0 8 に示した旧アクティブウィンドウのクローズに伴う処理に対応する。なお、このフローチャートの処理は、状態記憶部 1 0 7 により実行されるものである。即ち、外部メモリ 2 1 1 に記録された関連ウィンドウ参照支援プログラムを、CPU 2 0

10

20

30

40

50

1 が、R A M 2 0 3 にロードして実行することにより実現される。

【 0 1 3 3 】

まず、ステップ S 8 0 1 において、状態記憶部 1 0 7 は、旧アクティブウィンドウのウィンドウハンドルをキーに、ウィンドウ情報テーブル 1 0 4 を検索し、旧アクティブウィンドウの関連ウィンドウ表示機能が「 O N 」であるか否かを判定する。

【 0 1 3 4 】

ステップ S 8 0 1 で、旧アクティブウィンドウの関連ウィンドウ表示機能が「 O N 」でないと判定した場合には、状態記憶部 1 0 7 は、そのままステップ S 8 1 5 に処理を進める。

【 0 1 3 5 】

一方、ステップ S 8 0 1 で、旧アクティブウィンドウの関連ウィンドウ表示機能が「 O N 」であると判定した場合には、状態記憶部 1 0 7 は、ステップ S 8 0 2 に処理を進める。

【 0 1 3 6 】

ステップ S 8 0 2 では、状態記憶部 1 0 7 は、旧アクティブウィンドウの関連ウィンドウ情報配列を取得する。

【 0 1 3 7 】

次に、ステップ S 8 0 3 において、状態記憶部 1 0 7 は、旧アクティブウィンドウに関連ウィンドウがあるか否かを判定する。この判定の際、状態記憶部 1 0 7 は、S 8 0 2 で取得した関連ウィンドウ情報配列の要素数を調べ、要素数が 0 より大きい場合に、旧アクティブウィンドウの関連ウィンドウの存在すると判定する。

【 0 1 3 8 】

ステップ S 8 0 3 において、旧アクティブウィンドウが関連ウィンドウが存在しないと判定した場合には、状態記憶部 1 0 7 は、そのままステップ S 8 1 4 に処理を進める。

【 0 1 3 9 】

一方、ステップ S 8 0 3 において、旧アクティブウィンドウが関連ウィンドウが存在すると判定した場合には、状態記憶部 1 0 7 は、ステップ S 8 0 4 に処理を進める。

【 0 1 4 0 】

ステップ S 8 0 4 では、状態記憶部 1 0 7 は、旧アクティブウィンドウのファイルパスを元にファイル管理 D B 1 0 9 (図 1 1) を検索し、関連ファイル I D 1 1 0 3 を取得する。ここで、図 1 1 を参照して、ファイル管理 D B 1 0 9 について説明する。

【 0 1 4 1 】

図 1 1 は、外部メモリ 2 1 1 に記録されているファイル管理 D B 1 0 9 のデータ構造の一例を示す図である。

【 0 1 4 2 】

ファイルパス・U R L 1 1 0 1 は、各ファイルのファイルパスもしくは U R L が格納される。

【 0 1 4 3 】

関連ウィンドウ表示機能 1 1 0 2 は、関連ファイルを自動的に表示するか否かを示すものであり、前回ウィンドウクローズ時の値がセットされており、次回起動時にデフォルト値として用いられる。

【 0 1 4 4 】

関連ファイル I D 1 1 0 3 は、関連ファイル管理 D B 1 1 0 (図 1 2) において、関連するファイル情報を識別するための I D である。

【 0 1 4 5 】

図 1 2 は、外部メモリ 2 1 1 に記録されている関連ファイル管理 D B 1 1 0 のデータ構造を表として示す図である。

【 0 1 4 6 】

関連ファイル I D 1 2 0 1 は、関連ファイルを識別するための I D である。ファイル管理 D B 1 0 9 で関連元となるファイルがこの I D を持っており、この I D が一致するもの

10

20

30

40

50

をそのファイルの関連ファイル情報とする。

【 0 1 4 7 】

ファイルパス・URL 1202は、各ファイルのファイルパスもしくはURLが格納される。アプリケーション1203は、ファイルを開くのに用いるアプリケーションを表すものである。

【 0 1 4 8 】

表示順1204は、関連元のウィンドウを除いて、スクリーン上で上から何番目に表示するかを表すものである。関連元のウィンドウが最上位に表示され、それに続いて1から順に上位に表示される。

【 0 1 4 9 】

ウィンドウ領域1205は、左から順に、X座標、Y座標、幅、高さ、となっており、X座標、Y座標は、ウィンドウの左上の座標を表している。また、幅、高さはウィンドウの幅と高さを表している。

【 0 1 5 0 】

参照位置(X、Y)1206は、ファイル内のどの場所を表示するかを表すものである。左から、X座標、Y座標となっている。今回の実施形態では、スクロールの位置を表している。

【 0 1 5 1 】

以下、図10のフローチャートの説明に戻る。

【 0 1 5 2 】

ステップS805において、状態記憶部107は、S804で関連ファイルIDを取得できたか否かを判定し、関連ファイルIDを取得できなかったと判定した場合には、ステップS808に処理を進める。

【 0 1 5 3 】

一方、ステップS805において、S804で関連ファイルIDを取得できたと判定した場合には、状態記憶部107は、ステップS806に処理を進める。

【 0 1 5 4 】

ステップS806では、状態記憶部107は、関連ファイルIDを発行する。このとき、状態記憶部107は、ファイル管理DB109を調べ、他のファイルの関連ファイルIDと重複しないユニークなIDになるようにする。

【 0 1 5 5 】

次に、ステップS807において、状態記憶部107は、S806で発行したIDを旧アクティブウィンドウの関連ファイルIDとしてファイル管理DB109に保存する。そして、ステップS808に処理を進める。

【 0 1 5 6 】

ステップS808では、状態記憶部107は、S801で取得した関連ウィンドウ情報配列に含まれる関連ウィンドウ全てに対し、ステップS809～S813の処理を繰り返すように制御する。

【 0 1 5 7 】

まず、状態記憶部107は、ステップS809では、状態記憶部107は、処理対象の関連ウィンドウの関連ウィンドウ情報(参照位置、領域、表示順等の情報)をウィンドウ制御部102から取得する。

【 0 1 5 8 】

次に、ステップS810において、状態記憶部107は、S809で取得した関連ウィンドウの情報を関連ファイル管理DB110に保存する。このとき、S804で取得したIDもしくは、S806で発行したIDとファイルパスが一致するものを検索し、存在すれば、関連ウィンドウ情報(参照位置、ウィンドウ領域、表示順等の情報)を上書き保存する。一方、存在しなければ、関連ファイル管理DB110に関連ウィンドウ情報(関連ファイルID、ファイルパス、参照位置、ウィンドウ領域、表示順等)を追加保存する。

【 0 1 5 9 】

10

20

30

40

50

次に、ステップ S 8 1 1 において、状態記憶部 1 0 7 は、処理対象の関連ウィンドウがその他のウィンドウから参照されていないか判定する。この判定の際、状態記憶部 1 0 7 は、処理対象の関連ウィンドウのハンドルをキーにして、その他の関連ウィンドウ情報配列全てを検索する。そして、処理対象の関連ウィンドウのハンドルが他の関連ウィンドウ情報配列に含まれていなかった場合に、処理対象の関連ウィンドウが他のウィンドウから参照されていないと判断する。

【 0 1 6 0 】

そして、ステップ S 8 1 1 において、処理対象の関連ウィンドウがその他のウィンドウから参照されていないと判定した場合には、状態記憶部 1 0 7 は、ステップ S 8 1 2 に処理を進める。

10

【 0 1 6 1 】

ステップ S 8 1 2 では、状態記憶部 1 0 7 は、ウィンドウ制御部 1 0 2 にメッセージを送り、処理対象の関連ウィンドウをクローズし、処理対象の関連ウィンドウの情報をウィンドウ情報テーブル 1 0 9 から削除する。そして、ステップ S 8 0 9 に処理を戻し、次のウィンドウに処理対象を移す。

【 0 1 6 2 】

一方、ステップ S 8 1 1 において、処理対象の関連ウィンドウがその他のウィンドウから参照されていると判定した場合には、状態記憶部 1 0 7 は、ステップ S 8 1 3 に処理を進める。

【 0 1 6 3 】

20

ステップ S 8 1 3 では、状態記憶部 1 0 7 は、ウィンドウ制御部 1 0 2 にメッセージを送り、処理対象の関連ウィンドウをクローズせず、最小化する。そして、ステップ S 8 0 9 に処理を戻し、次のウィンドウに処理対象を移す。

【 0 1 6 4 】

そして、S 8 0 1 で取得した関連ウィンドウ情報配列に含まれる関連ウィンドウ全てに対し、ステップ S 8 0 9 ~ S 8 1 3 の処理を終了すると、状態記憶部 1 0 7 は、ステップ S 8 1 4 に処理を進める。

【 0 1 6 5 】

ステップ S 8 1 4 では、状態記憶部 1 0 7 は、旧アクティブウィンドウの関連ウィンドウ情報配列を R A M 2 0 3 から破棄し、ステップ S 8 1 5 に処理を進める。

30

【 0 1 6 6 】

ステップ S 8 1 5 では、状態記憶部 1 0 7 は、旧アクティブウィンドウのウィンドウ情報をウィンドウ情報テーブル 1 0 4 から削除し、本フローチャートの処理を終了する。

【 0 1 6 7 】

次に、図 9 のステップ S 7 1 6 に示した新アクティブウィンドウのオープンに伴う処理について、図 1 3 を参照して説明する。

【 0 1 6 8 】

図 1 3 は、本発明における第 7 の制御処理手順の一例を示すフローチャートであり、図 9 のステップ S 7 1 6 に示した新アクティブウィンドウのオープンに伴う処理に対応する。なお、このフローチャートの処理は、状態復元部 1 0 6 により実行されるものである。即ち、外部メモリ 2 1 1 に記録された関連ウィンドウ参照支援プログラムを、C P U 2 0 1 が、R A M 2 0 3 にロードして実行することにより実現される。

40

【 0 1 6 9 】

まず、ステップ S 9 0 1 において、状態復元部 1 0 6 は、新アクティブウィンドウで開くファイルのパスを取得する。

【 0 1 7 0 】

次に、ステップ S 9 0 2 において、状態復元部 1 0 6 は、S 9 0 1 で取得したファイルパスをキーにファイル管理 D B 1 0 9 を検索する。

【 0 1 7 1 】

次に、ステップ S 9 0 3 において、状態復元部 1 0 6 は、S 9 0 2 でのファイル管理 D

50

B 検索処理の結果、S 9 0 1 で取得したファイルパスに対応するファイルがファイル管理 DB 1 0 9 に登録されているか否かを判定する。

【 0 1 7 2 】

そして、ステップ S 9 0 3 で、S 9 0 1 で取得したファイルパスに対応するファイルがファイル管理 DB 1 0 9 に登録されていると判定した場合には、状態復元部 1 0 6 は、ステップ S 9 0 6 に処理を進める。

【 0 1 7 3 】

一方、ステップ S 9 0 3 で、S 9 0 1 で取得したファイルパスに対応するファイルがファイル管理 DB 1 0 9 に登録されていないと判定した場合には、状態復元部 1 0 6 は、ステップ S 9 0 4 に処理を進める。

【 0 1 7 4 】

ステップ S 9 0 4 では、状態復元部 1 0 6 は、関連ウィンドウ表示機能設定画面 1 4 0 0 (図 1 4) を表示し、ユーザに回答を求める。

【 0 1 7 5 】

図 1 4 は、関連ウィンドウ表示機能設定画面 1 4 0 0 の一例を示す図である。

【 0 1 7 6 】

図 1 4 において、1 4 0 1、1 4 0 2 はラジオボタンとなっており、いずれかが必ず選択状態となっている。

【 0 1 7 7 】

1 4 0 1 は、「関連ウィンドウ表示機能をONにする」ラジオボタンであり、このボタンが選択状態で、OK ボタン 1 4 0 3 が押された場合は、アクティブウィンドウの関連ウィンドウ表示機能がONとなる。

【 0 1 7 8 】

1 4 0 2 は、「関連ウィンドウ表示機能をOFFにする」ラジオボタンであり、このボタンが選択状態でOK ボタン 1 4 0 3 が押された場合は、アクティブウィンドウの関連ウィンドウ表示機能がOFFとなる。

【 0 1 7 9 】

以下、図 1 3 のフローチャートの説明に戻る。

【 0 1 8 0 】

関連ウィンドウ表示機能設定画面 1 4 0 0 でOK ボタン 1 4 0 3 が押下されると、状態復元部 1 0 6 は、ステップ S 9 0 5 に処理を進める。

【 0 1 8 1 】

ステップ S 9 0 5 では、状態復元部 1 0 6 は、新アクティブウィンドウのファイルパス、関連ウィンドウ表示機能等のファイル情報をファイル管理 DB 1 0 9 に登録する。この際、状態復元部 1 0 6 は、関連ウィンドウ表示機能設定画面 1 4 0 0 で、「ONにする」ボタン 1 4 0 1 が選択された状態でOK ボタン 1 4 0 3 が押下された場合、関連ウィンドウ表示機能 1 1 0 2 を「ON」に登録する。一方、「OFFにする」ボタン 1 4 0 2 が選択された状態でOK ボタン 1 4 0 3 が押下された場合、状態復元部 1 0 6 は、関連ウィンドウ表示機能 1 1 0 2 を「OFF」に登録する。

【 0 1 8 2 】

ステップ S 9 0 5 の処理が終了すると、状態復元部 1 0 6 は、ステップ S 9 0 6 に処理を進める。

【 0 1 8 3 】

ステップ S 9 0 6 では、状態復元部 1 0 6 は、ファイル管理 DB 1 0 9 に登録されている新アクティブウィンドウの関連ウィンドウ表示機能 1 1 0 2 が「ON」か否かを判定し、「ON」でない場合、ステップ S 9 1 7 に処理を進める。

【 0 1 8 4 】

一方、ステップ S 9 0 6 において、ファイル管理 DB 1 0 9 に登録されている新アクティブウィンドウの関連ウィンドウ表示機能 1 1 0 2 が「ON」の場合、ステップ S 9 0 7 に処理を進める。

10

20

30

40

50

【 0 1 8 5 】

ステップ S 9 0 7 では、状態復元部 1 0 6 は、R A M 2 0 3 上に新アクティブウィンドウの関連ウィンドウ情報配列を作成する。

【 0 1 8 6 】

次に、ステップ S 9 0 8 において、状態復元部 1 0 6 は、新アクティブウィンドウの関連ファイル I D がファイル管理 D B 1 0 9 に存在するか判定し、存在しないと判定した場合には、ステップ S 9 1 7 に処理を進める。

【 0 1 8 7 】

一方、ステップ S 9 0 8 において、新アクティブウィンドウの関連ファイル I D がファイル管理 D B 1 0 9 に存在すると判定した場合には、状態復元部 1 0 6 は、新アクティブウィンドウの関連ファイル I D をファイル管理 D B 1 0 9 から取得し、ステップ S 9 0 9 に処理を進める。

10

【 0 1 8 8 】

ステップ S 9 0 9 では、状態復元部 1 0 6 は、ステップ S 9 0 9 で取得した関連ファイル I D をキーに関連ファイル管理 D B 1 1 0 を検索する。

【 0 1 8 9 】

次に、ステップ S 9 1 0 において、状態復元部 1 0 6 は、S 9 0 9 の検索処理で関連ファイル I D が一致した関連ファイル全てに対し、ステップ S 9 1 1 ~ S 9 1 6 の処理を繰り返す。

【 0 1 9 0 】

20

ステップ S 9 1 1 で、状態復元部 1 0 6 は、関連ファイル管理 D B 1 1 0 から処理対象の関連ファイルの関連ファイル情報を取得する。

【 0 1 9 1 】

次に、ステップ S 9 1 2 において、状態復元部 1 0 6 は、S 9 1 1 で取得した関連ファイル情報のうち、ファイルパスをキーに、ウィンドウ情報テーブル 1 0 4 を検索する。そして、上記 S 9 1 1 で取得した関連ファイル情報のファイルパスがウィンドウ情報テーブル 1 0 4 に存在する場合には、該ファイルパスに対応するファイルのウィンドウがオープン済みであると判断し、ステップ S 9 1 4 に処理を進める。

【 0 1 9 2 】

一方、ステップ S 9 1 2 で、状態復元部 1 0 6 は、上記 S 9 1 1 で取得した関連ファイル情報のファイルパスがウィンドウ情報テーブル 1 0 4 に存在しない場合には、状態復元部 1 0 6 は、該関連ファイル情報に対応するファイルのウィンドウがオープン済みでないと判断し、ステップ S 9 1 3 に処理を進める。

30

【 0 1 9 3 】

ステップ S 9 1 3 では、状態復元部 1 0 6 は、S 9 1 1 で取得した関連ファイル情報のアプリケーション 1 2 0 3 でファイルパス 1 2 0 2 の関連ファイルを開いて、新規ウィンドウをオープンする。、該関連ファイルのウィンドウ情報をウィンドウ情報テーブル 1 0 4 に追加する。そして、ステップ S 9 1 4 に処理を進める。

【 0 1 9 4 】

ステップ S 9 1 4 では、状態復元部 1 0 6 は、S 9 1 1 で取得した関連ファイル情報に対応するウィンドウのウィンドウハンドルをウィンドウ情報テーブル 1 0 4 から取得する。

40

【 0 1 9 5 】

次に、ステップ S 9 1 5 において、状態復元部 1 0 6 は、S 9 1 1 で取得した関連ファイル情報のうち、ファイルパス、表示順、ウィンドウ領域、参照位置、S 9 1 4 で取得したウィンドウハンドルを、S 9 0 7 で作成した関連ウィンドウ情報配列にセットする。

【 0 1 9 6 】

次に、ステップ S 9 1 6 において、状態復元部 1 0 6 は、S 9 1 6 で関連ウィンドウ情報配列にセットした関連ウィンドウ情報を元に、処理対処の関連ウィンドウの状態を復元する。そして、ステップ S 9 1 1 に処理を戻し、次の関連ファイルに処理対象を移す。

【 0 1 9 7 】

50

そして、状態復元部 106 は、S909 の検索処理で関連ファイル ID が一致した関連ファイル全てに対し、ステップ S911 ~ S916 の処理を終了すると、ステップ S917 に処理を進める。

【0198】

ステップ S917 では、状態復元部 106 は、ウィンドウ情報テーブル 104 に、新アクティブウィンドウのウィンドウハンドル、ファイルパス、関連ウィンドウ表示機能、関連ウィンドウ情報配列ポインタ等のウィンドウ情報を追加して、本フローチャートの処理を終了する。

【0199】

次に、図 15、図 16 を参照して、本実施形態におけるファイルの関連イメージ及びアクティブウィンドウの切り替えイメージを具体的に説明する。

10

【0200】

図 15 は、本実施形態におけるファイルの関連イメージの一例を示す模式図である。

【0201】

「提案書 1」1510 と「提案書 2」1520 は、本実施形態において、関連ウィンドウ表示機能が「ON」の関連元ウィンドウとしての役割を持つものである。

【0202】

そして、各々には関連ウィンドウが存在し、各関連ウィンドウのウィンドウハンドルやファイルパス、参照位置、ウィンドウ領域、表示順等のウィンドウ状態に関する情報を持っている。

20

【0203】

「提案書 1」1510 は、「提案書サンプル」1511、「類似特許 1」1512 と関連付けられている。

【0204】

「提案書 2」1520 は、「提案書サンプル」1521、「類似特許 2」1522、「図表フォルダ」1523 と関連付けられている。

【0205】

なお、「提案書サンプル」1511 と、「提案書サンプル」1521 は、同一ファイルであるが、参照位置、ウィンドウ領域、表示順等の情報がそれぞれ異なっている。即ち、「提案書サンプル」は、「提案書 1」1510、「提案書 2」1520 の双方に関連付けられ、それぞれ「提案書サンプル」1511、「提案書サンプル」1521 として図示されている。

30

【0206】

次に図 15 での関連を元にした、アクティブウィンドウ切り替え時のスクリーン上のイメージを図 16 を用いて説明する。

【0207】

図 16 は、アクティブウィンドウ切り替え時の画面上のイメージを具体的に示す模式図である。

【0208】

図 16 に示すように、マルチウィンドウ表示装置のスクリーン（画面）上には、「提案書 1.doc」、「類似特許 1.doc」、「提案書サンプル.pdf」、「提案書 2.doc」、「類似特許 2.doc」、「図表フォルダ」、以上のウィンドウが開かれている。

40

【0209】

なお、ユーザは、「提案書 1.doc」、「提案書 2.doc」を関連ウィンドウ表示機能が ON の状態で開いている。

【0210】

まず、スクリーンイメージ 1610 において、ユーザは、「提案書 1.doc」を編集している。その際、「提案書 1.doc」の関連ウィンドウとして「提案書サンプル.doc」、「類似特許 1.pdf」もスクリーン上に表示されている。

【0211】

50

その後、ユーザが「提案書 1.doc」を最小化し、「提案書 2.doc」をアクティブにすると、マルチウィンドウ表示装置は、スクリーンイメージ 1 6 2 0 のように関連ウィンドウを参照支援制御する。

【 0 2 1 2 】

即ち、「提案書 1.doc」が最小化されることで、それに関連付けられている「類似特許 1.pdf」、「提案書サンプル.doc」のウィンドウ状態（表示状態）が、関連ウィンドウ情報配列 1 0 5 に記憶された上で、最小化される。

【 0 2 1 3 】

そして、「提案書 2.doc」がアクティブになることで、それに関連付けられている「図表」、「提案書サンプル.pdf」、「類似特許 2.doc」がスクリーン上に表示される。スクリーンイメージ 1 6 1 0 の時とスクリーンイメージ 1 6 2 0 の時とでは、アクティブウィンドウが異なるため、同じ「提案書サンプル.pdf」でも、ウィンドウ領域、参照位置、表示順が異なって表示されている。

【 0 2 1 4 】

そして再び、ユーザが「提案書 1.doc」をアクティブにすることで、マルチウィンドウ表示装置は、スクリーンイメージ 1 6 3 0 のように制御する。

【 0 2 1 5 】

即ち、スクリーンの状態が、前回、「提案書 1.doc」を閉じる時に記憶しておいたウィンドウ状態に戻るため、スクリーンイメージ 1 6 3 0 とスクリーンイメージ 1 6 1 0 は同じウィンドウ状態となっている。

【 0 2 1 6 】

そして、「提案書サンプル.doc」のウィンドウ状態も、スクリーンイメージ 1 6 2 0 の状態から、スクリーンイメージ 1 6 1 0 の時の状態に戻っている。

【 0 2 1 7 】

以上説明したように、本実施形態のマルチウィンドウ表示装置の関連ウィンドウ参照支援制御によれば、監視部 1 0 3 が、ウィンドウの監視を行うことで、アクティブウィンドウが切り替わることを検知する。これを受けて、アクティブウィンドウの関連ウィンドウ表示機能が ON の場合には、関連付け部 1 0 8 が、ウィンドウの関連付けを行う。そして、アクティブウィンドウに予め関連付けられたウィンドウの状態を内部メモリ（RAM 2 0 3）上の関連ウィンドウ情報配列にセットして記憶する。

【 0 2 1 8 】

また、ウィンドウのクローズによるアクティブウィンドウの切り替えの場合、状態記憶部 1 0 7 が、そのウィンドウに予め関連付けられているウィンドウの状態を関連ファイル管理 DB に記憶する。その際、その他のウィンドウから参照されていない場合には、関連ウィンドウもクローズする。

【 0 2 1 9 】

そして、ウィンドウ切り替えによって新たにアクティブになるウィンドウに予め関連付けられたウィンドウの状態を、関連ウィンドウ情報配列を元に状態復元部 1 0 6 が復元する。

【 0 2 2 0 】

また、ファイルのオープンによるウィンドウの切り替えの場合、新たにアクティブになるウィンドウに予め関連付けられているウィンドウの状態を関連ファイル管理 DB の情報を元に復元する。

【 0 2 2 1 】

このように、本実施形態により、ウィンドウが切り替わる等の状況の変化に合わせて、それに関連するウィンドウ（ファイル）の表示状態も変化させることができる。

【 0 2 2 2 】

なお、上記したように実施形態では、関連ウィンドウの参照位置を含む関連ウィンドウ情報を、関連ウィンドウ情報配列 1 0 5 や、関連ファイル管理 DB に記憶しておき、アクティブウィンドウを切り替えたときや、ウィンドウをクローズしたとき、ウィンドウをオ

10

20

30

40

50

ーブンしたときなど、上記関連ウィンドウ情報に基づいて、新たにアクティブになったウィンドウの関連ウィンドウの参照位置を元に戻す構成を有する。

【0223】

さらに、関連ウィンドウ情報として、関連ウィンドウ内の選択状態や、カーソル位置、ボタン選択状態等も記憶しておき、関連元のウィンドウがアクティブになった際、上記関連ウィンドウ情報に基づいて、関連ウィンドウの選択状態や、カーソル位置、ボタン選択状態等も復元するように構成してもよい。

【0224】

なお、上述した各種データの構成及びその内容はこれに限定されるものではなく、用途や目的に応じて、様々な構成や内容で構成されることは言うまでもない。

10

【0225】

以上、一実施形態について示したが、本発明は、例えば、システム、装置、方法、プログラムもしくは記録媒体等としての実施態様をとることが可能であり、具体的には、複数の機器から構成されるシステムに適用しても良いし、また、一つの機器からなる装置に適用しても良い。

【0226】

以下、図17に示すメモリマップを参照して本発明に係るマルチウィンドウ表示装置で読み取り可能なデータ処理プログラムの構成について説明する。

【0227】

図17は、本発明に係るマルチウィンドウ表示装置で読み取り可能な各種データ処理プログラムを記録した記録媒体（記憶媒体）のメモリマップを説明する図である。

20

【0228】

なお、特に図示しないが、記録媒体に記憶されるプログラム群を管理する情報、例えばバージョン情報、作成者等も記憶され、かつ、プログラム読み出し側のOS等に依存する情報、例えばプログラムを識別表示するアイコン等も記憶される場合もある。

【0229】

さらに、各種プログラムに従属するデータも上記ディレクトリに管理されている。また、インストールするプログラムやデータが圧縮されている場合に、解凍するプログラム等も記憶される場合もある。

【0230】

30

本実施形態における図3、図5、図7、図8、図9、図10、図13に示す機能が外部からインストールされるプログラムによって、ホストコンピュータにより遂行されていてもよい。そして、その場合、CD-ROMやフラッシュメモリやFD等の記録媒体により、あるいはネットワークを介して外部の記録媒体から、プログラムを含む情報群を出力装置に供給される場合でも本発明は適用されるものである。

【0231】

以上のように、前述した実施形態の機能を実現する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、システムあるいは装置に供給し、そのシステムあるいは装置のコンピュータ（またはCPUやMPU）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も、本発明の目的が達成されることは言うまでもない。

40

【0232】

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本発明の新規な機能を実現することになり、そのプログラムコードを記憶した記録媒体は本発明を構成することになる。

【0233】

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、フレキシブルディスク、ハードディスク、光ディスク、光磁気ディスク、CD-ROM、CD-R、DVD-ROM、磁気テープ、不揮発性のメモリカード、ROM、EEPROM、シリコンディスク等を用いることができる。

【0234】

50

また、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により、前述した実施形態の機能が実現されるだけでなく、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働しているOS（オペレーティングシステム）等が実際の処理の一部または全部を行い、その処理によって前述した実施形態の機能が実現される場合も含まれることは言うまでもない。

【0235】

さらに、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込まれた後、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CPU等が実際の処理の一部または全部を行い、その処理によって前述した実施形態の機能が実現される場合も含まれることは言うまでもない。

10

【0236】

また、本発明は、複数の機器から構成されるシステムに適用しても、1つの機器からなる装置に適用してもよい。また、本発明は、システムあるいは装置にプログラムを供給することによって達成される場合にも適用できることは言うまでもない。この場合、本発明を達成するためのソフトウェアによって表されるプログラムを格納した記録媒体を該システムあるいは装置に読み出すことによって、そのシステムあるいは装置が、本発明の効果を享受することが可能となる。

【0237】

さらに、本発明を達成するためのソフトウェアによって表されるプログラムをネットワーク上のサーバ、データベース等から通信プログラムによりダウンロードして読み出すことによって、そのシステムあるいは装置が、本発明の効果を享受することが可能となる。

20

【0238】

なお、上述した各実施形態およびその変形例を組み合わせた構成も全て本発明に含まれるものである。

【0239】

以上示したように、本実施形態のマルチウィンドウ表示装置によれば、ウィンドウが切り替わる等の状況の変化に合わせて、それに関連するウィンドウ（ファイル）の表示状態を変化させることで、ウィンドウ切り替え時の操作者の負荷を軽減し、関連ウィンドウの参照作業の効率を向上させることができる。

30

【図面の簡単な説明】

【0240】

【図1】本発明の一実施形態を示すマルチウィンドウ表示装置の構成の一例を示すブロック図である。

【図2】図1に示したマルチウィンドウ表示装置に適用可能なコンピュータ（情報処理装置）のハードウェア構成の一例を示すブロック図である。

【図3】本発明における第1の制御処理手順の一例を示すフローチャートである。

【図4】ウィンドウ情報テーブル、関連ウィンドウ情報配列の一例を示す図である。

【図5】本発明における第2の制御処理手順の一例を示すフローチャートである。

【図6】関連付け確認画面の一例を示す図である。

40

【図7】本発明における第3の制御処理手順の一例を示すフローチャートである。

【図8】本発明における第4の制御処理手順の一例を示すフローチャートである。

【図9】本発明における第5の制御処理手順の一例を示すフローチャートである。

【図10】本発明における第6の制御処理手順の一例を示すフローチャートである。

【図11】ファイル管理DBのデータ構造の一例を示す図である。

【図12】関連ファイル管理DBのデータ構造を表として示す図である。

【図13】本発明における第7の制御処理手順の一例を示すフローチャートである。

【図14】関連ウィンドウ表示機能設定画面の一例を示す図である。

【図15】本実施形態におけるファイルの関連イメージの一例を示す模式図である。

【図16】アクティブウィンドウ切り替え時の画面上のイメージを具体的に示す模式図で

50

ある。

【図 17】本発明に係るマルチウィンドウ表示装置で読み取り可能な各種データ処理プログラムを記録した記録媒体（記憶媒体）のメモリマップを説明する図である。

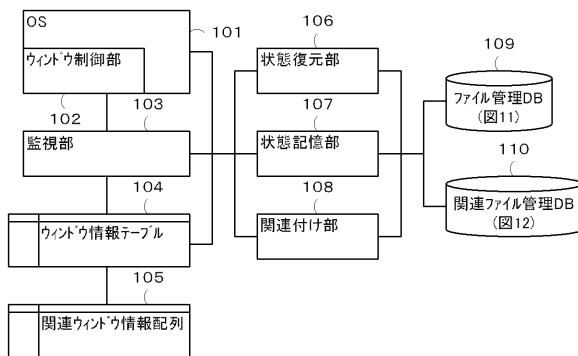
【符号の説明】

【0241】

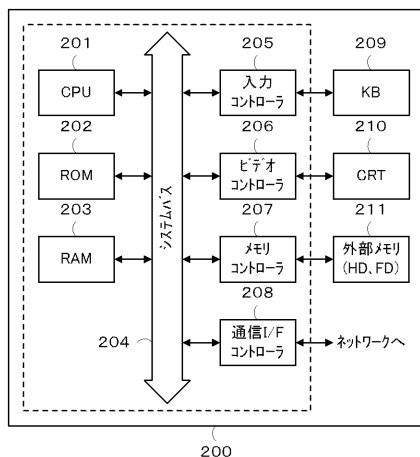
- 101 OS
- 102 ウィンドウ制御部
- 103 監視部
- 104 ウィンドウ情報テーブル
- 105 関連ウィンドウ情報配列
- 106 状態復元部
- 107 状態記憶部
- 108 関連付け部
- 109 ファイル管理DB
- 110 関連ファイル管理DB
- 1300 関連付け確認画面
- 1400 関連ウィンドウ表示機能設定画面

10

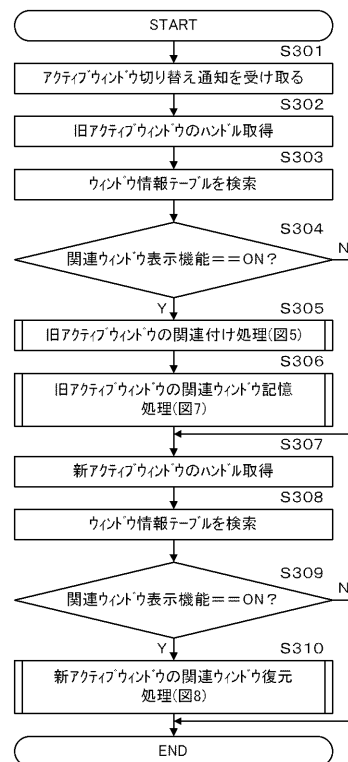
【図 1】



【図 2】



【図 3】



【図 4】

1001 1002 1003 1004

実行時にメモリ上で
持つ情報例
ウィンドウ情報テーブル
104

ウィンドウハンドル	ファイル名・URL	関連ウィンドウ表示機能	関連ウィンドウ情報配列ポインタ
0 × 11111111	document\提案書1.doc	ON	0 × 00010000
0 × 22222222	document\提案書2.doc	ON	0 × 00020000
0 × 33333333	document\類似特許1.doc	OFF	NULL
0 × 55555555	document\提案書サブルーブル.pdf	OFF	NULL
0 × 44444444	document\類似特許2.doc	OFF	NULL
0 × 66666666	document\図表ファイル	OFF	NULL
.	.	.	.
.	.	.	.
.	.	.	.

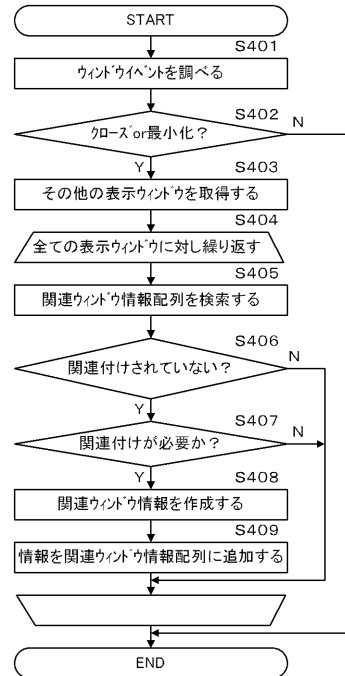
関連ウィンドウ
情報配列
105

1005 1006 1007 1008 1009

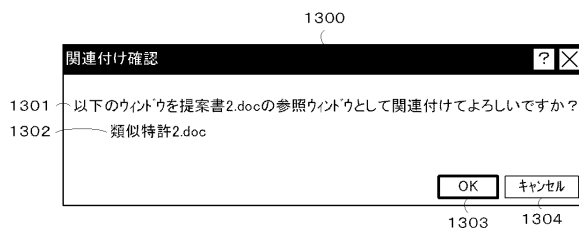
ウィンドウハンドル	ファイル名・URL	表示順	ウィンドウ領域(x,y,width,height)	参照位置(X,Y)
0 × 33333333	document\類似特許1.doc	2	510,10,400,300	0,100
0 × 55555555	document\提案書サブルーブル.pdf	1	550,200,400,600	50,0

ウィンドウハンドル	ファイル名・URL	表示順	ウィンドウ領域(x,y,width,height)	参照位置(X,Y)
0 × 44444444	document\類似特許2.doc	3	530,0,400,300	200,200
0 × 55555555	document\提案書サブルーブル.pdf	1	530,0,400,300	200,200
0 × 66666666	document\図表ファイル	2	600,20,430,350	350,10

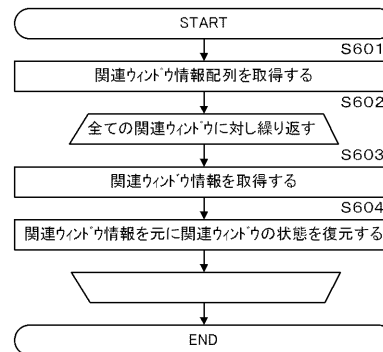
【図 5】



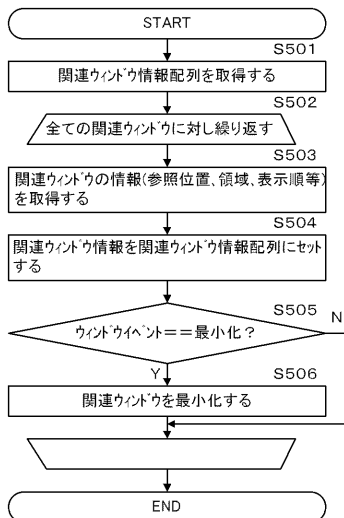
【図 6】



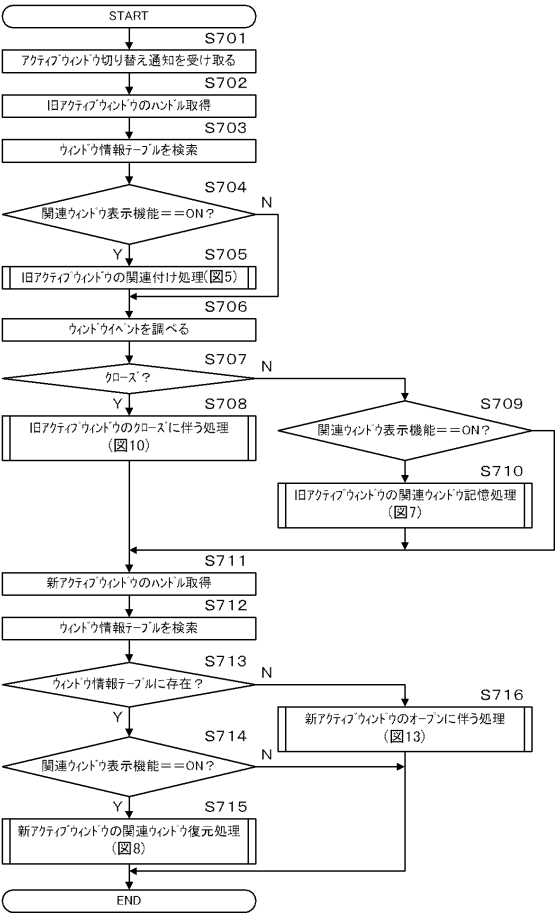
【図 8】



【図 7】



【図 9】

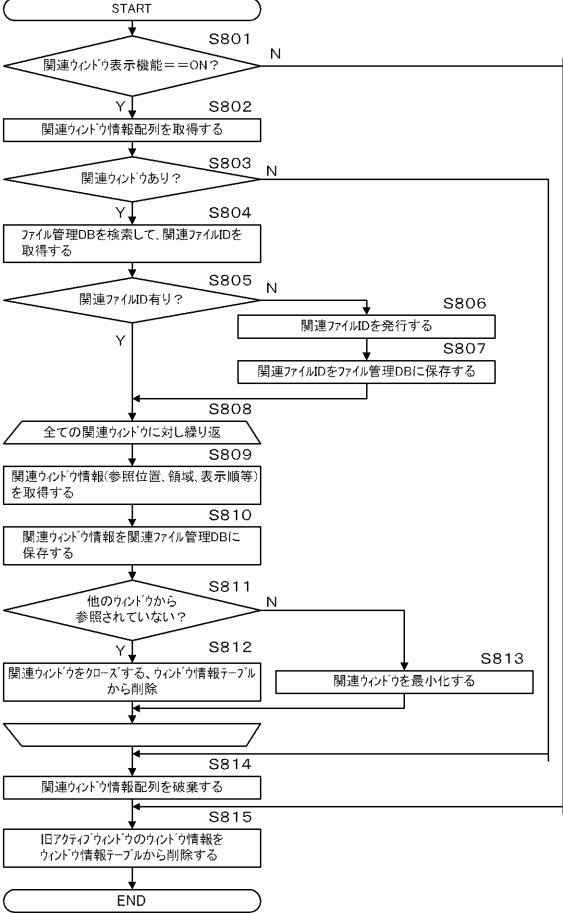


【図 11】

ファイル管理データベース1109

1101	1102	1103
ファイル名orURL	関連ウィンドウ表示機能	関連ファイルID
document¥提案書1.doc	ON	1
document¥提案書2.doc	ON	2
document¥類似特許1.pdf	OFF	
document¥類似特許2.pdf	OFF	
document¥提案書サンプル.pdf	OFF	
document¥図表.xls	OFF	
...
...
...

【図 10】

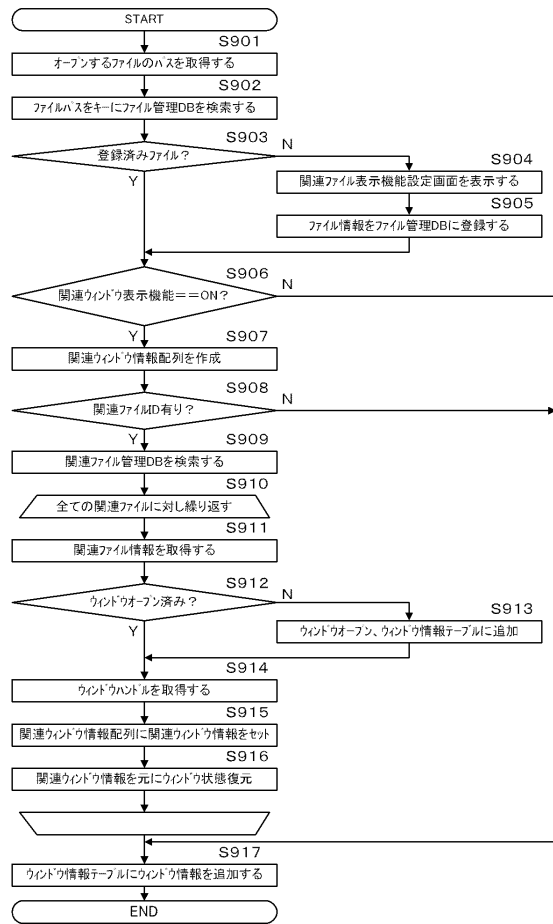


【図 12】

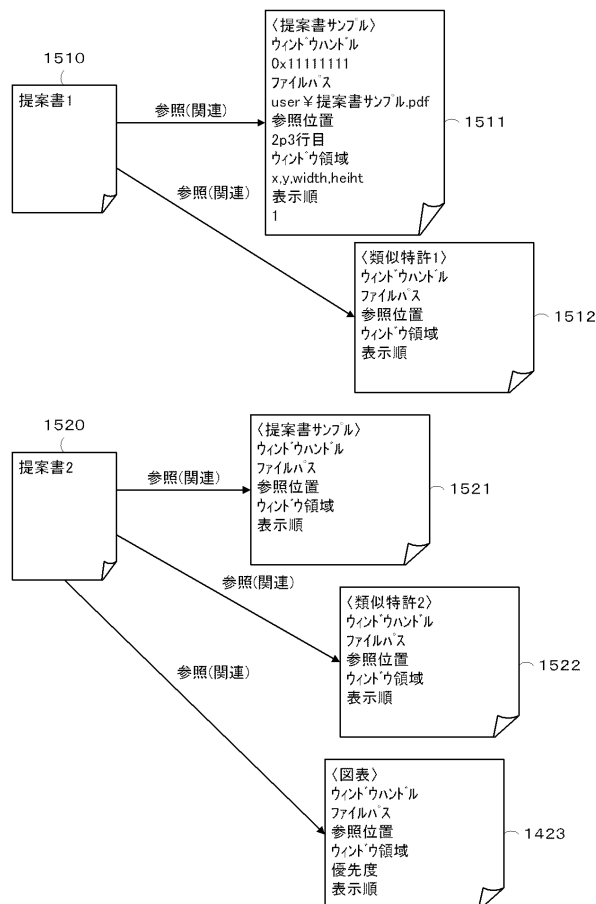
関連ファイル管理データベース1110

1201	1202	1203	1204	1205	1206
関連ファイルID	ファイル名orURL	アプリケーション	表示順	ウィンドウ領域(x,y,width,height)	参照位置(x,y)
1	document¥類似特許1.doc	word		2510,0,400,300	0,100
1	document¥提案書サンプル.pdf	pdf reader		1550,200,400,600	50,0
2	document¥類似特許2.doc	word		3530,10,400,300	200,200
2	document¥図表ファイル.xls	1537ローラ		1530,400,400,300	200,200
2	document¥提案書サンプル.pdf	pdf reader		2600,20,430,350	350,10
...
...
...

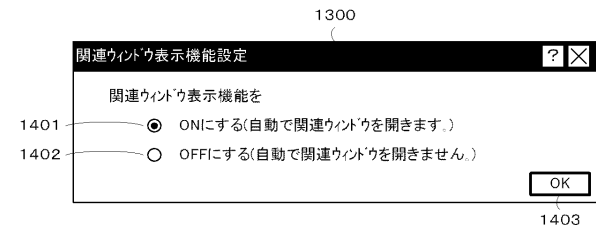
【図 13】



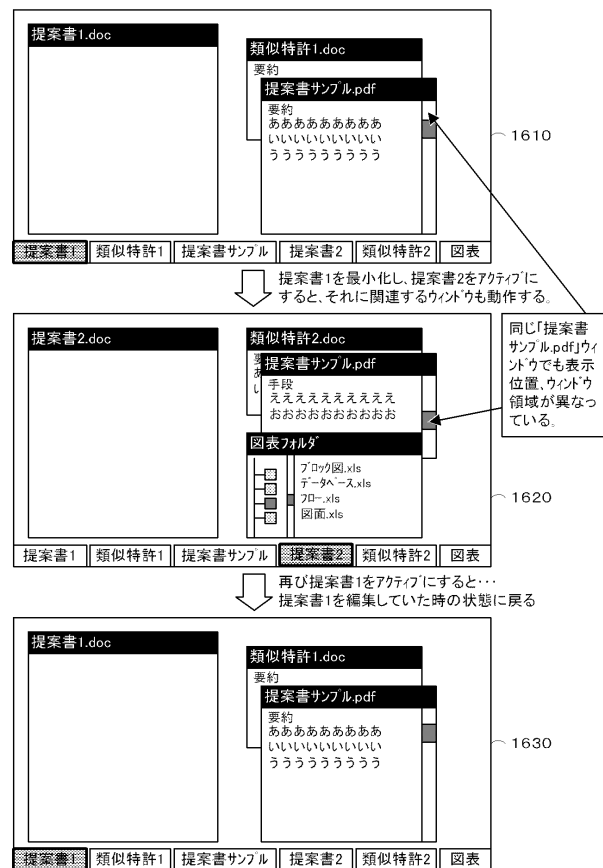
【図 15】



【図 14】



【図 16】



【図 17】

FD/CD-ROM等の記憶媒体	
ディレクトリ情報	
第1の処理プログラム	図3に示すフローチャートのステップに対応するプログラムコード群
第2の処理プログラム	図5に示すフローチャートのステップに対応するプログラムコード群
第3の処理プログラム	図7に示すフローチャートのステップに対応するプログラムコード群
第4の処理プログラム	図8に示すフローチャートのステップに対応するプログラムコード群
第5の処理プログラム	図9に示すフローチャートのステップに対応するプログラムコード群
第6の処理プログラム	図10に示すフローチャートのステップに対応するプログラムコード群
第7の処理プログラム	図13に示すフローチャートのステップに対応するプログラムコード群
記憶媒体のメモリマップ	

フロントページの続き

- (72)発明者 土井 直樹
東京都港区三田3丁目9番7号 キヤノンソフトウェア株式会社内
- (72)発明者 矢口 幸弘
東京都港区三田3丁目9番7号 キヤノンソフトウェア株式会社内

審査官 星野 昌幸

- (56)参考文献 特開平10-124283(JP,A)
国際公開第2006/107324(WO,A1)
特開平07-129355(JP,A)
特開平10-260810(JP,A)
特開平04-199093(JP,A)

- (58)調査した分野(Int.Cl., DB名)
G06F 3/048