

(19)日本国特許庁(JP)

(12)特許公報(B2)(11)特許番号
特許第7013176号
(P7013176)

(45)発行日 令和4年1月31日(2022.1.31)

(24)登録日 令和4年1月21日(2022.1.21)

(51)国際特許分類

G 0 6 F 8/61 (2018.01)

F I

G 0 6 F 8/61

請求項の数 7 (全17頁)

(21)出願番号	特願2017-1711179(P2017-1711179)	(73)特許権者	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
(22)出願日	平成29年9月6日(2017.9.6)	(74)代理人	100126240 弁理士 阿部 琢磨
(65)公開番号	特開2019-46360(P2019-46360A)	(74)代理人	100124442 弁理士 黒岩 創吾
(43)公開日	平成31年3月22日(2019.3.22)	(72)発明者	布施 洋 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
審査請求日	令和2年7月31日(2020.7.31)	審査官	金田 孝之

最終頁に続く

(54)【発明の名称】 プログラム、情報処理装置、及び制御方法

(57)【特許請求の範囲】**【請求項1】**

アプリケーションソフトウェアを情報処理装置にインストールするためのプログラムであつて、

前記アプリケーションのベース機能を前記情報処理装置にインストールする第1のインストール処理を実行する第1のインストール手段を有し、

前記第1のインストール手段は、

前記第1のインストール手段によって前記情報処理装置にインストールされた前記アプリケーションをアップデートするための未実行の第2のインストール処理が存在する場合、前記第2のインストール処理を実行する第2のインストール手段に対して前記第2のインストール処理を実行するように指示し、第1のインストール処理が完了していたとしても前記第2のインストール処理が完了するまでは前記アプリケーションのショートカットを作成せず、

未実行の第2のインストール処理が存在しない場合、前記第1のインストール処理が完了すると前記アプリケーションのショートカットを作成することを特徴とするプログラム。

【請求項2】

前記プログラムは、

未実行である前記第2のインストール処理が存在するかどうかを判定する第1の判定手段を更に有し、

前記第1の判定手段は、

前記情報処理装置のOSによって管理されたレジストリ情報に、前記第2のインストール手段を識別する識別情報が存在する場合に、未実行である前記第2のインストール処理が存在すると判定することを特徴とする請求項1に記載のプログラム。

【請求項3】

前記プログラムは、

前記第2のインストール手段によって前記アプリケーションがアップデートされた後、

前記ショートカットが存在するかを判定する第2の判定手段を更に有し、

前記第2の判定手段によって前記ショートカットが存在すると判定された場合は、

前記ショートカットを作成することなく処理を終了することを特徴とする請求項1または2に記載のプログラム。

10

【請求項4】

アプリケーションソフトウェアをインストール可能な情報処理装置であって、

前記アプリケーションのベース機能を前記情報処理装置にインストールする第1のインストール処理を実行する第1のインストール手段を有し、

前記第1のインストール手段は、

前記第1のインストール手段によって前記情報処理装置にインストールされた前記アプリケーションをアップデートするための未実行の第2のインストール処理が存在する場合、

前記第2のインストール処理を実行する第2のインストール手段に対して前記第2のインストール処理を実行するように指示し、第1のインストール処理が完了していたとしても前記第2のインストール処理が完了するまでは前記アプリケーションのショートカットを作成せず、

20

未実行の第2のインストール処理が存在しない場合、前記第1のインストール処理が完了すると前記アプリケーションのショートカットを作成することを特徴とする情報処理装置。

【請求項5】

前記情報処理装置は、

未実行である前記第2のインストール処理が存在するかどうかを判定する第1の判定手段を更に有し、

前記第1の判定手段は、

前記情報処理装置のOSによって管理されたレジストリ情報に、前記第2のインストール手段を識別する識別情報が存在する場合に、未実行である前記第2のインストール処理が存在すると判定することを特徴とする請求項4に記載の情報処理装置。

30

【請求項6】

前記情報処理装置は、

前記第2のインストール手段によって前記アプリケーションがアップデートされた後、

前記ショートカットが存在するかを判定する第2の判定手段を更に有し、

前記第2の判定手段によって前記ショートカットが存在すると判定された場合は、

前記ショートカットを作成することなく処理を終了することを特徴とする請求項4または5に記載のプログラム。

【請求項7】

アプリケーションソフトウェアをインストール可能な情報処理装置の制御方法であって、

前記アプリケーションのベース機能を前記情報処理装置にインストールする第1のインストール処理を実行する第1のインストールステップを有し、

前記第1のインストールステップは、

前記第1のインストールステップによって前記情報処理装置にインストールされた前記アプリケーションをアップデートするための未実行の第2のインストール処理が存在する場合、前記第2のインストール処理を実行する第2のインストールステップに対して前記第2のインストール処理を実行するように指示し、第1のインストール処理が完了していたとしても前記第2のインストール処理が完了するまでは前記アプリケーションのショートカットを作成せず、

40

未実行の第2のインストール処理が存在しない場合、前記第1のインストール処理が完了

50

すると前記アプリケーションのショートカットを作成することを特徴とする情報処理装置の制御方法。

【発明の詳細な説明】

【技術分野】

【0001】

本発明は、アプリケーションソフトウェアをインストールするためのプログラム、情報処理装置、及び制御方法に関する。

【背景技術】

【0002】

近年、アプリケーションソフトウェアの大規模・複雑化に伴い、1つのアプリケーションソフトウェアを動作させるために複数のインストーラを正しい手順で実行する必要のある製品が増えてきた。しかしながら、ITリテラシーの低いユーザにとって正しい手順でインストーラを実行することは必ずしも容易なことではない。

10

【0003】

そこで特許文献1には、サーバーコンピュータにおいて、複数のソフトウェアをインストールする際のインストール順によるコンフリクトチェック（整合性確認）を実施することが記載されている。特許文献1によれば、ユーザはインストーラの実行順序を意識することなく所望のアプリケーションソフトウェアがインストールできるようになる。

【先行技術文献】

【特許文献】

20

【0004】

【文献】特開2012-48410号公報

【発明の概要】

【発明が解決しようとする課題】

【0005】

しかしながら、複数のインストーラを連続実行している際にユーザがインストール中のアプリケーションソフトウェアを起動してしまうと、インストール処理が失敗する場合がある。

【0006】

例えば、ユーザがサービスパックを含むアプリケーションソフトウェアを購入した場合、製品にはベース用のインストーラ及びサービスパック用のインストーラが同梱されていることがある。一般的にこのような構成のインストーラでは、ベースのインストール処理が完了した後に自動的にサービスパックのインストール処理が開始されることが多い。この場合、ベースのインストール処理が完了した時点でOS(Operating System)のデスクトップ画面上にショートカットが作成される。そのため、サービスパックのインストール中にユーザがショートカットからアプリケーションソフトウェアを容易に起動することが出来てしまう。

30

【0007】

ここでユーザがインストール中のアプリケーションソフトウェアを起動した場合、OSによってアプリケーションソフトウェア実行に必要なモジュールはロードされ、ロックされる。そのため、サービスパックのインストーラによるサービスパック用モジュールの更新ができなくなるため、サービスパックのインストール処理が失敗するという問題が発生する。

40

【0008】

また、サービスパックのインストーラによる更新ができなかったモジュールを、ソフトウェアの再起動後に更新することも可能である。しかしながら、ユーザがサービスパックを含むアプリケーションソフトウェアを購入したにもかかわらず、そのユーザが「Bug-Fix未適用の製品」や「新機能がまだ存在しないベース機能のみの製品」を一時的にでも使用することは好ましくない。

【0009】

50

そこで、本発明は、既にインストールされたアプリケーションソフトウェアについての追加インストールを行うためのインストーラを起動させる前に、前記インストールされたアプリケーションソフトウェアの起動のための処理を実行させないことを目的とする。

【課題を解決するための手段】

【0010】

上記課題を解決するために、第1のインストール手段によって情報処理装置にインストールされたアプリケーションをアップデートするための未実行の第2のインストール処理が存在する場合、第2のインストール処理を実行する第2のインストール手段に対して第2のインストール処理を実行するように指示し、第1のインストール処理が完了していたとしても前記第2のインストール処理が完了するまでは前記アプリケーションのショートカットを作成せず、未実行の第2のインストール処理が存在しない場合、前記第1のインストール処理が完了すると前記アプリケーションのショートカットを作成することを特徴とする。

10

【発明の効果】

【0011】

本発明により、既にインストールされたアプリケーションソフトウェアについての追加インストールを行うためのインストーラを起動させる前に、前記インストールされたアプリケーションソフトウェアの起動のための処理を実行させないようにすることができる。

【図面の簡単な説明】

【0012】

20

【図1】ハードウェア構成例を示す図

【図2】ソフトウェア構成例を示す図

【図3】インストーラに関する設定情報の一例を示す図

【図4】実施例1の処理を示すフローチャート

【図5】実施例2の処理を示すフローチャート

【図6】実施例3の処理を示すフローチャート

【発明を実施するための形態】

【0013】

本発明では、ユーザがインストーラ実行中に、アプリケーションソフトウェアの起動を防ぐためのショートカット作成方法を実現する。

30

【0014】

以下、本発明を実施するための最良の形態について図面を用いて説明する。

【0015】

【ハードウェア構成】

図1は、本発明の一実施形態におけるハードウェア構成例を示す図である。図1では、コンピュータである情報処理装置100のハードウェア構成例を示している。情報処理装置100は、ユーザ操作入力を受信する入力デバイスであるキーボード105とポインティングデバイス106が設けられている。

【0016】

キーボード105は、情報処理装置100に対して文字や数字などの情報を入力する装置である。キーボードは、例えば仮名や英文字などを入力するキーや数字を入力するためのテンキー、各種機能キー、カーソルキー及びその他のキーを備える。

40

【0017】

ポインティングデバイス106は、例えばマウスなどを備える。ユーザが、G U I (G r a p h i c a l U s e r I n t e r f a c e)などを用いて情報処理装置100を操作する場合、表示装置上に表示されたボタンやアイコンなどをマウスでクリックすることにより、所定の情報の入力を行うことができる。例えば、情報処理装置100が、インストールされたアプリケーションソフトウェアを表すアイコンを表示部102上に表示し、ユーザがアイコンをマウスでクリックすることに応じて、C P U 101がアプリケーションソフトウェアを起動させる。さらに、ユーザに視覚的な出力情報フィードバックを与える

50

表示部 102 を備える。

【0018】

表示部 102 は、例えば、C R T (C a t h o d e R a y T u b e) ディスプレイ、液晶ディスプレイ、又はプラズマディスプレイを備える画面上に各種情報を表示する。表示部 102 は、キー ボードやマウスの入力結果や、インストーラやアプリケーションソフトウェアが提供する画面（例えばユーザインターフェース）等を表示する。

【0019】

さらに、情報処理装置 100 は、動作を制御するための手段である C P U 101、R A M 103、R O M 104、H D D 107 を備える。本実施形態に特有の動作として、C P U 101 は、情報処理装置 100 上では図 2 に示す複数のインストーラ 200・210 の動作を制御する。C P U 101 は、情報処理装置 100 の各インストーラを実行し、所定のプログラムに従って、各種の演算、情報処理、情報処理装置 100 全体の制御処理を行う。

10

【0020】

R A M 103、R O M 104、H D D 107 は本実施形態における各種プログラムや実行情報を保管する。また、D V D / C D に記録されたデータを読み出すための D V D / C D - R O M ドライバ 108 も設ける。R A M 103 は、C P U 101 により O S (O p e r a t i n g S y s t e m) や本実施例を実現するための制御プログラムを含むその他の制御プログラムがロードされ実行される。また、制御プログラムを実行するために用いられる各種作業領域、一時待避領域として機能する。本実施形態では、R A M 103 には、C P U 101 がインストーラの起動やシステム内部の処理などを行うためのエリアが確保される。R O M 104 は、C P U 101 の各種制御を実行する各種制御プログラム、データ及びパラメータ等を記憶する記憶装置である。

20

【0021】

H D D 107 は、読み書き可能な記憶媒体と、その記憶媒体に対してプログラムやデータを読み書きするための駆動装置と、を備える。本実施形態では、H D D 107 は、ハードディスクを備えるローカルディスクである。なお、H D D 107 が備える記憶媒体が、光磁気ディスク、磁気ディスク、半導体メモリ等であってもよい。H D D 107 には、図 2 に示す複数のインストーラ 200・210 が記憶されている。また、H D D 107 には、インターフェイスデバイス I / O 109 を制御し、ネットワークを介した通信を維持する通信プログラムが記憶されている。また、H D D 107 には、メモリ管理やファイル入出力管理などのクライアントを動作させるための基本アプリケーションソフトウェアである O S などが記憶されている。

30

【0022】

さらに、外部機器との通信を行うインターフェイスデバイス I / O 109 を備え、プログラム実行を行う C P U 101 を備える。なお、周辺機器との接続形態は有線 / 無線を問わない。インターフェイスデバイス I / O 109 は、ネットワーク（例えば、L A N やインターネット）を介して他の端末（W e b サーバ、メールサーバなど）に接続する。インターネットを介した通信を行う場合には、例えば、T C P / I P 等のプロトコルが用いられる。

40

【0023】

なお、本実施例で説明される処理は、R O M 104、またはH D D 107 に保存されたプログラムや実行情報、一時データをR A M 103 にロードし、C P U 101 で実行することによって実現される。

【0024】

[ソフトウェア構成]

図 2 は、ソフトウェアの構成例を示す図である。

【0025】

情報処理装置 100 は、複数のインストーラ（第一のインストーラ 200、第二のインストーラ 210）を備える。なお、本実施例では、インストーラ 210 はインストーラ 200 と同様の構成であり重複する情報は省略する。

50

【0026】

ここで、第一のインストーラ200および第二のインストーラ210は、アプリケーションソフトウェアの制作者によってDVDやCD等に記憶され、或いは、アーカイブファイルに変換されWebページ等にアップロードされることが一般的である。また、インストーラの配布方式は、単数のインストーラを別々に配布する単数インストーラ形式や、複数のインストーラをあわせて配布する複数インストーラ形式の2パターンが存在する。例えば、アプリケーションソフトウェアのサービスパックがリリースされて、ベースとサービスパックの2つのインストーラを1つにまとめて配布する場合、複数インストーラ形式となる。なお、ユーザのPCに既にベースのインストールしており、後にリリースされたサービスパックを追加でインストールする場合には、サービスパックのインストーラが単数インストーラ形式で配布されることとなる。図2で示すような第一のインストーラ200および第二のインストーラ210は、複数インストーラ形式で配布されたインストーラセットであると言える。

10

【0027】

情報処理装置100が備える各構成について説明する。

【0028】

インストーラ200は、アプリケーションソフトウェアのインストール実行するための各種のユーザインターフェースを備え、情報処理装置100のデスクトップ上でユーザからの要求を受け付けて、各機能処理の実行を総括的に管理している。

20

【0029】

インストーラ200は、表示部201および制御部202を有する。各インストーラを構成するコンポーネントについて説明する。

【0030】

表示部201は、インストーラ200のユーザインターフェースの構築を行う。また、GUIを介して、ユーザからの各種入力操作の受け付けを処理する。

【0031】

制御部202は、インストーラ制御に関する処理全般の機能を備える。例えば、ユーザからの各種入力操作の受け付けを基に、図示を省略するアプリケーションソフトウェアのインストール、アンインストール、修復インストールやアプリケーションの設定や構築を行う。また、制御部202は、情報処理装置100のインストーラに関する設定情報を解析する。設定情報とは、例えば、インストーラ情報300や各インストーラが管理するレジストリ情報320等を示す。設定情報の詳細については、図3を用いて後述する。

30

【0032】

さらに、制御部202は、アプリケーションソフトウェアのインストールに際して、既にインストールされたアプリケーションソフトウェアの起動のためのユーザ入力を受け付け可能にする処理を実行する。その処理の一例として、制御部202は、情報処理装置のOSが提供するデスクトップ画面上に、アプリケーションソフトウェアの起動のためのユーザ入力を受け付けるショートカットを作成する。また処理の他の例として、制御部202は、メニュー画面の一覧にアプリケーションソフトウェアを表示させたり、ファイルに対する拡張子の関連付けを行ったりしてもよい。

40

【0033】

なお、インストーラ200は、アプリケーションソフトウェアの基本機能を情報処理装置にインストールするためのインストーラである。また、表示部211および制御部212を有するインストーラ210は、アプリケーションソフトウェアの追加インストールするためのサービスパックのインストーラである。サービスパックのインストーラにおける追加インストールでは、例えば、ベースのインストーラにおける基本機能についてのバグ修正やセキュリティ強化、ベースの機能をバージョンアップした機能の追加などが行われる。一方、図5で後述する実施例2におけるインストーラ210は、アプリケーションソフトウェアの追加インストールとして、オプション機能をインストールするためのインストーラとして扱う。オプション機能のインストーラにおいては、ベースの機能には含まれな

50

い付帯機能であり、ユーザがカスタマイズして追加可能な機能がインストールされる。サービスパックのインストーラの制御部212は、ベースのインストーラの制御部202と同様に、既にインストールされたアプリケーションソフトウェアの起動のためのユーザ入力を受け付け可能にする処理を実行可能である。例えば、制御部212はショートカット作成の機能などを有している。一方、オプション機能のインストーラの制御部212は、既にインストールされたアプリケーションソフトウェアの起動のためのユーザ入力を受け付け可能にする処理を実行する機能を有していないため、ショートカットの作成などはできない。

【0034】

本実施例で説明される処理は、ROM104またはHDD107に保存されたプログラムが、RAM103にロードされて、CPU101で実行されることによって実現される。

10

【0035】

[インストーラに関する設定情報]

図3は、インストーラに関する設定情報の一例を示している。

【0036】

インストーラに関する設定情報には、インストーラが保持するインストーラ情報およびOSが保持するレジストリ情報が含まれる。これらの情報は各インストーラのインストール処理において、併用して利用される。

【0037】

初めに、図3(a)で示すインストーラ情報300について説明する。インストーラ情報300は、インストーラ200のインストール処理に必要な設定情報であり、インストーラ200が保持する情報である。設定情報は、例えばInstallerVersion301やProductCode302、UpgradeCode等を含む。同様に、図3(b)で示すインストーラ情報310は、インストーラ210のインストール処理に必要な設定情報であり、インストーラ210が保持する情報である。なお、本実施例では、インストーラ情報310はインストーラ情報300と同様の構成であり、重複する情報の説明は省略する。

20

【0038】

インストーラ情報300の各項目について説明する。InstallerVersion301は、インストーラ210がインストール可能なアプリケーションソフトウェアのバージョンを示す。一般的にアプリケーションソフトウェアのバージョンは、メジャーバージョン、マイナーバージョン、リビジョン(サービスパックのバージョン)の3つの値で構成される。例えば、メジャーバージョンの値が「3」、マイナーバージョンの値が「0」、リビジョンの値が「0」であれば、アプリケーションソフトウェアのバージョンは「3.0.0」となる。本実施例におけるInstallerVersion301の値は「3.0.0」であるため、インストーラ200によってバージョン3.0.0のアプリケーションソフトウェアがインストールされる。また、インストーラ情報310のInstallerVersion311の値は「3.0.1」であるため、インストーラ210によってバージョン3.0.1のアプリケーションソフトウェアがインストールされる。

30

【0039】

ProductCode302は、アプリケーションソフトウェアの識別などに用いられる固有の番号や文字列である。この値はインストーラ200によって、レジストリ情報320のProductCode322へ書き込まれる。また、一般的に、リリース済みのアプリケーションソフトウェアに適用可能なサービスパックをリリースする場合、ProductCode302およびUpgradeCodeは変更しない。即ち、サービスパックをインストールした場合、レジストリ情報320に書き込まれるProductCode322の値およびUpgradeCodeの値は変更されない。なお、第二のインストーラ210がオプション機能のインストーラである場合、ProductCodeやUpgradeCodeは、第一のインストーラ200が保持する値と異なる(不図示)。値が異なる場合、第一のインストーラ200と第二のインストーラ210によってインス

40

50

トールされるアプリケーションソフトウェアは、それぞれアップグレードの依存関係がないことになる。なお、Product Version 323 および Product Code 324 については、実施例 2 で説明する。

【0040】

図3を例にまとめると、インストーラ情報300を有するインストーラ200はベースであり、インストーラ情報310を有するインストーラ210は、ベースに対するサービスパックであると言える。

【0041】

次に、図3(c)で示すレジストリ情報320について説明する。レジストリ情報320は、情報処理装置100のOSが保持するレジストリ情報一覧の一例を示す。

10

【0042】

レジストリ情報320は、情報処理装置100にインストールされているアプリケーションソフトウェアの情報であり、ROM104に保存されている。例えば、レジストリ情報320は、アプリケーションソフトウェアのProduct Version 321やProduct Code 322、Upgrade Code、インストールパス等を含む。また、レジストリ情報320は複数のアプリケーションソフトウェアを管理する。図3を例にあげると、レジストリ情報320は、ベースのアプリケーションソフトウェアとベースに対するオプション機能のアプリケーションソフトウェアの2つのアプリケーションソフトウェアを含む。具体的には、Product Version 321、Product Code 322はベースのアプリケーションソフトウェアのレジストリ情報に関する情報である。また、Product Version 323、Product Code 324はベースに対するオプション機能のアプリケーションソフトウェアのレジストリ情報に関する情報である。なお、レジストリ情報320に含まれる2つのProduct Version、Product Codeにおいて、重複する情報の説明は省略する。

20

【0043】

レジストリ情報320の各項目について説明する。Product Version 321は、情報処理装置100にインストールされているベースのアプリケーションソフトウェアのバージョンを示す。例えば、Product Version 321の値が「3.0.0」であれば、バージョン3.0.0のアプリケーションソフトウェアがインストールされていることになる。

30

【0044】

ここで、サービスパックがインストールされている場合は、リビジョンの値が「1」以上となる。したがって、図3(c)で示す例では、リビジョンの値が「0」であるため、サービスパックはインストールされていないと判断できる。

【0045】

Product Code 322は、アプリケーションソフトウェアの識別などに用いられる固有の番号や文字列である。レジストリ情報320にアプリケーションソフトウェアのProduct Code 302が存在すれば、アプリケーションソフトウェアがインストールされているということになる。

40

【0046】

なお、インストーラ情報、および、レジストリ情報が保持する情報は、本実施例に限ったものとしない。

【0047】

[実施例1における各インストーラの処理]

実施例1では、第一のインストーラ200と第二のインストーラ210が連続実行される場合における課題の解決手法を説明する。なお、実施例1では第一のインストーラ200をベースのインストーラ、第二のインストーラ210をサービスパックのインストーラとして扱う。

【0048】

まず、実施例1における課題を説明する。ベース用とサービスパック用の2つのインスト

50

ーラが連続で実行される場合にサービスパックのインストール処理が失敗するケースがある。具体的には、2つのインストーラを連続実行する場合、まず初めにベースのインストーラによって、OSのデスクトップ画面上にショートカットが作成される。その後サービスパックのインストーラが実行されることとなるが、ベースのインストーラ実行が完了した時点でユーザはOSのデスクトップ画面上に作成されたショートカットからアプリケーションソフトウェアを起動するための指示をすることができる。ここで、サービスパックのインストーラ実行中にアプリケーションソフトウェアが起動された場合、サービスパックのインストーラはアプリケーションソフトウェアのモジュールを更新できずにインストール処理が失敗することがある。

【0049】

10

ここでは前述の課題を解決するためのインストーラ構成に応じたショートカット作成のタイミングの動的な変更手法について説明する。

【0050】

【実施例1における各インストーラの処理】

図4は、ベースのインストーラ200およびサービスパックのインストーラ210の処理のフローチャートである。図1の情報処理装置100上で動作するインストーラ200・210を例に処理の流れを説明する。図4に示すフローチャートでは、インストーラにおける動的なショートカット作成処理を実施する手法を示す。また、S1001～S1010の各ステップはインストーラ200による処理であり、S1101～S1103の各ステップはインストーラ210による処理である。

20

【0051】

図4(a)は、ベースのインストーラであるインストーラ200による処理の流れを示すフローチャートである。

【0052】

インストーラ200が起動されると、S1001にて、制御部202はインストーラ200でインストール可能なアプリケーションソフトウェアが未インストールかを判定する。はい(未インストールである)の場合、S1002へ進み、いいえ(未インストールでない)の場合、S1003へ進む。

【0053】

S1001における判定のロジックについて具体的に説明する。制御部202は、レジストリ情報320を取得する。そして制御部202は、レジストリ情報320とインストーラ情報300を基に、インストーラ情報300が保持するProductCode302がレジストリ情報320に存在するかを解析する。レジストリ情報320のProductCode322にProductCode302が存在する場合、ベースのアプリケーションソフトウェアは未インストールでないと判定し、S1003へ進む。一方、レジストリ情報320のProductCode322にProductCode302が存在しない場合、または、レジストリ情報320が存在しない場合、ベースのアプリケーションソフトウェアは未インストールであると判定し、S1002へ進む。

30

【0054】

次に、S1002にて、制御部202はベースのアプリケーションソフトウェアのインストール実行フラグをONにする。

40

【0055】

次に、S1003にて、複数インストーラ形式のインストーラセットに未インストールのサービスパックが存在するかを判定する。はい(存在する)の場合、S1004へ進み、いいえ(存在しない)の場合、S1005へ進む。

【0056】

S1003における判定のロジックについて具体的に説明する。初めに制御部202は、サービスパックのインストーラ情報310を取得する。ここでインストーラ情報310が取得できない場合、サービスパックのインストーラ210は存在しないと判定し、S1005へ進む。次に制御部202は、レジストリ情報320を取得する。そして制御部202

50

2は、サービスパックのインストーラ情報310とレジストリ情報320とを基に、インストーラ情報310が保持するProductCode312がレジストリ情報320に存在するかを解析する。レジストリ情報320のProductCode322にProductCode302が存在しない場合、または、レジストリ情報320が存在しない場合、未インストールのサービスパックが存在すると判定し、S1004へ進む。一方、レジストリ情報320のProductCode322にProductCode302が存在する場合、さらにInstallerVersion311とProductVersion321とを比較する。ここでは、例えば、それぞれのリビジョンの値を比較することとなる。比較した結果、InstallerVersion311の方が大きい場合、未インストールのサービスパックが存在すると判定し、S1004へ進む。一方、InstallerVersion311の方が小さい或いは同じ場合、未インストールのサービスパックが存在しないと判定し、S1005へ進む。

【0057】

次に、S1004にて、制御部202はサービスパックのアプリケーションソフトウェアのインストール実行フラグをONにする。

【0058】

次に、S1005にて、制御部202はインストールするアプリケーションソフトウェアが1つ以上存在するかを判定する。ここで、S1005における判定のロジックには「ベースのアプリケーションソフトウェアのインストール実行フラグ」と「サービスパックのアプリケーションソフトウェアのインストール実行フラグ」とを用いる。少なくともいずれかのフラグがONであれば、何らかのアプリケーションソフトウェアをインストールすると判定する。はい(存在する)の場合、S1006へ進み、いいえ(存在しない)の場合、本フローチャートの処理を終了する。

【0059】

次に、S1006にて、制御部202はベースのアプリケーションソフトウェアのインストール実行フラグがONかを判定する。はい(インストール実行フラグがON)の場合、S1007へ進み、いいえ(インストール実行フラグがOFF)の場合、S1008へ進む。

【0060】

次に、S1007にて、制御部202はアプリケーションソフトウェアの基本機能をインストールする。S1007で実行されるインストール処理は、アプリケーションソフトウェアのファイルコピーやレジストリ登録である。なお、S1007で実行されるインストールの完了後には、制御部202はショートカットを作成しないものとする。後述するサービスパックのインストーラによる追加機能のインストールの完了後に、ショートカットが作成されることとなる。

【0061】

次に、S1008にて、制御部202はサービスパックのアプリケーションソフトウェアのインストール実行フラグがONかを判定する。はい(インストール実行フラグがON)の場合、S1009へ進み、いいえ(インストール実行フラグがOFF)の場合、S1010へ進む。

【0062】

次に、S1009にて、制御部202はサービスパックをインストールするためのインストーラ210を起動させ、インストーラ210による処理(図4(b))を実行する。つまりS1009では、前記アプリケーションソフトウェアのインストールに際して、該インストールされたアプリケーションソフトウェアの起動のためのユーザ入力を受け付け可能にする処理を実行することなく、制御部202はインストーラ210を起動する。

【0063】

次に、S1010にて、制御部202はOSのデスクトップ画面上にアプリケーションソフトウェアのショートカットを作成し、インストーラ200の処理を終了する。

【0064】

10

20

30

40

50

図4(b)は、サービスパックのインストーラであるインストーラ210による処理の流れを示すフローチャートである。

【0065】

S1009でインストーラ210がインストーラ200によって起動されると、S1101にて、制御部212はアプリケーションソフトウェアの追加機能をインストールする。S1101で実行されるインストール処理は、アプリケーションソフトウェアのファイルコピーやレジストリ登録の処理であって、ショートカット作成の処理を含まないものとする。

【0066】

次に、S1102にて、制御部212はOSのデスクトップ画面上にアプリケーションソフトウェアのショートカットが存在するかを判定する。はい(存在する)の場合、インストーラ210の処理を終了する。いいえ(存在しない)の場合、S1103へ進む。

10

【0067】

次に、S1103にて、制御部212はOSのデスクトップ画面上にアプリケーションソフトウェアのショートカットを作成し、インストーラ210の処理を終了する。

【0068】

なお、S1101でのサービスパックのインストール処理が完了した後、サービスパックのインストーラではなくベースのインストーラがショートカットを作成しても良い。

20

【0069】

また、事前にベースのアプリケーションソフトウェアのみをインストール済みの場合、すでにOSのデスクトップ画面上にアプリケーションソフトウェアのショートカットが作成されている。しかしながら、ユーザがOSのデスクトップ画面上からショートカットを削除していることがある。そのため、サービスパックのインストーラが実行するS1102、S1103の処理で、ベースのインストーラが作成していたショートカットを修復する必要があるかを判定し、必要があれば修復するようにしてよい。

【0070】

以上、インストーラ構成に応じたショートカット作成タイミングの動的な変更手法について説明した。これにより、ベース用とサービスパック用の2つのインストーラが連続で実行される場合、ベースのインストーラでショートカットを作成せずに、サービスパックのインストーラでショートカットを作成できるようになる。従って、インストーラの構成に関わらず、ショートカット作成のタイミングが一律してインストール処理の最後に実施されるようになり、ユーザがインストール中にショートカットからアプリケーションソフトウェアを起動することがなくなる。その結果、インストール中におけるアプリケーションソフトウェアの起動操作を抑止することができるようになる。

30

【0071】

[実施例2における各インストーラの処理]

実施例2では、第一のインストーラ200と第二のインストーラ210が連続実行される場合における課題の解決手法を説明する。なお、実施例2では第一のインストーラ200をベースのインストーラ、第二のインストーラ210をオプション機能のインストーラとして扱う。

40

【0072】

実施例2における課題を説明する。オプション機能のインストーラとはベースのアプリケーションソフトウェアに対し、オプション機能をインストールできるインストーラである。一般的にオプション機能のインストーラは、ベースに対するオプション機能をインストールするだけであり、ベースのアプリケーションソフトウェアのショートカットを作成する処理は存在しない。そのため、実施例1の課題と同様に、ベース用とオプション機能用の2つのインストーラが連続で実行される場合、オプション機能が有効になる前にユーザがアプリケーションソフトウェアを起動してしまう可能性がある。

【0073】

実施例2では前述の課題を解決するためのインストーラ構成に応じたショートカット作成

50

タイミングの動的な変更手法について説明する。

【0074】

[実施例2における各インストーラの処理]

図5は、ベースのインストーラ200およびオプション機能のインストーラ210の処理のフロー図である。図1の情報処理装置100上で動作するインストーラ200・210を例に処理の流れを説明する。本フローでは、インストーラにおける動的なショートカット作成処理を実施する手法を示す。また、S1001～S1010の各ステップはインストーラ200による処理であり、S1101～S1103の各ステップはインストーラ210による処理である。なお、図4に示したフローチャートのステップと重複する部分についての説明を省略する。

10

【0075】

図5(a)は、ベースのインストーラであるインストーラ200による処理の流れを示すフローチャートである。

【0076】

S2003にて、複数インストーラ形式のインストーラセットに未インストールのオプション機能が存在するかを判定する。はい(存在する)の場合、S2004へ進み、いいえ(存在しない)の場合、S2005へ進む。

【0077】

S2003における判定のロジックについて具体的に説明する。初めに制御部202は、オプション機能のインストーラ情報310を取得する。ここでインストーラ情報310が取得できない場合、オプション機能のインストーラ210は存在しないと判定し、S2005へ進む。次に制御部202は、レジストリ情報320を取得する。そして制御部202は、オプション機能のインストーラ情報310とレジストリ情報320を基に、インストーラ情報310が保持するProductCode312がレジストリ情報320に存在するかを解析する。

20

【0078】

レジストリ情報320のProductCode324にProductCode302が存在しない場合、または、レジストリ情報320が存在しない場合、未インストールのオプション機能が存在すると判定し、S2004へ進む。一方、レジストリ情報320のProductCode324にProductCode302が存在する場合、インストーラ情報310のInstallerVersion311とレジストリ情報320のProductVersion323とを比較する。ここでは、例えば、それぞれのリビジョンの値を比較することとなる。比較した結果、InstallerVersion311の方が大きい場合、未インストールのオプション機能が存在すると判定し、S2004へ進む。一方、InstallerVersion311の方が小さい或いは同じ場合、未インストールのオプション機能が存在しないと判定し、S2005へ進む。

30

【0079】

次に、S2004にて、制御部202はオプション機能のアプリケーションソフトウェアのインストール実行フラグをONにする。

【0080】

次に、S2005にて、制御部202はインストールするアプリケーションソフトウェアが1つ以上存在するかを判定する。ここで、S2005における判定には「ベースのアプリケーションソフトウェアのインストール実行フラグ」と「オプション機能のアプリケーションソフトウェアのインストール実行フラグ」を用いる。何れかのフラグがONであれば、何らかのアプリケーションソフトウェアをインストールすると判定される。はい(存在する)の場合、S1006へ進み、いいえ(存在しない)の場合、インストーラ200の処理を終了する。

40

【0081】

次に、S2008にて、制御部202はオプション機能のアプリケーションソフトウェアのインストール実行フラグがONかを判定する。はい(インストール実行フラグがON)

50

の場合、S2009へ進み、いいえ（インストール実行フラグがOFF）の場合、S2010へ進む。

【0082】

次に、S2009にて、制御部202はオプション機能のアプリケーションソフトウェアをインストールするためのインストーラ210を起動させ、インストーラ210による処理（図5（b））を実行する。

【0083】

次に、S2010にて、制御部202はOSのデスクトップ画面上にアプリケーションソフトウェアのショートカットを作成し、インストーラ200の処理を終了する。

【0084】

図5（b）は、オプション機能のインストーラであるインストーラ210による処理の流れを示すフローチャートである。

【0085】

S2009でインストーラ210がインストーラ200によって起動されると、S2101にて、制御部212はベースに対して適用可能なオプション機能をインストールし、インストーラ210の処理を終了する。S2101で実行されるインストール処理は、アプリケーションソフトウェアのファイルコピーやレジストリ登録の処理であって、ショートカット作成の処理を含まないものとする。S2101でのオプション機能のインストール処理が完了した後、S2010でインストーラ200によってショートカットが作成される。

【0086】

以上、インストーラ構成に応じたショートカット作成タイミングの動的な変更手法について説明した。これにより、ベース用とオプション機能用の2つのインストーラが連続で実行される場合、オプション機能のインストーラが終了した後にベースのインストーラでショートカットを作成できるようになる。従って、インストーラの構成に関わらず、ショートカット作成のタイミングが一律してインストール処理の最後に実施されるようになり、ユーザがインストール中にショートカットからアプリケーションソフトウェアを起動することがなくなる。その結果、インストール中におけるアプリケーションソフトウェアの起動操作を抑止することができるようになる。

【0087】

〔実施例3におけるインストーラの処理〕

実施例3では、第二のインストーラ210が単体実行される場合における課題の解決手法を説明する。なお、実施例3では第二のインストーラ210をサービスパックのインストーラとして扱う。

【0088】

実施例3における課題を説明する。事前にベースのアプリケーションソフトウェアがインストールされている状態では、OSのデスクトップ画面上にショートカットが作成済みになっている。この状態でサービスパックのインストールのみを単体で実行した場合、ユーザはサービスパックのインストーラ実行中にOSのデスクトップ画面上に存在するショートカットから容易にアプリケーションソフトウェアが起動できる。そのため、実施例1や実施例2の課題と同様にサービスパックのインストーラを単体で実行する場合、サービスパックのインストールが完了する前にユーザがアプリケーションソフトウェアを起動してしまう可能性がある。

【0089】

実施例3では、前述の課題を解決するためのショートカット存在の有無に応じたショートカット作成手法について説明する。

【0090】

〔実施例3における各インストーラの処理〕

図6は、サービスパックのインストーラ210の処理の流れを示すフローチャートである。図1の情報処理装置100上で動作するインストーラ210を例に処理の流れを説明す

10

20

30

40

50

る。本フローでは、インストーラにおけるショートカット存在の有無に応じたショートカット作成処理を実施する手法を示す。

【0091】

インストーラ210は起動処理が開始されると、S3001にて、制御部212はOSのデスクトップ画面上にアプリケーションソフトウェアのショートカットが存在するかを判定する。はい(存在する)の場合、S3002へ進み、いいえ(存在しない)の場合、S3003へ進む。

【0092】

次に、S3002にて、制御部212はOSのデスクトップ画面上に存在するアプリケーションソフトウェアのショートカットを削除する。

10

【0093】

次に、S3003にて、制御部212はベースに対して適用可能なサービスパックのアプリケーションソフトウェアをインストールする。S3003で実行されるインストール処理は、アプリケーションソフトウェアのファイルコピーやレジストリ登録の処理であり、ショートカット作成の処理を含まないものとする。

【0094】

次に、S3004にて、制御部212はOSのデスクトップ画面上にアプリケーションソフトウェアのショートカットを作成し、インストーラ210の処理を終了する。

【0095】

なお、S3002でショートカットを削除するのではなく、アプリケーションを起動できないことを示すようなショートカットの表示に一時的に変更しても良い。

20

【0096】

また、実施例3におけるインストーラ210は、サービスパックのインストーラに限ったものとしない。また、実施例3におけるソフトウェア構成は、情報処理装置100には第一のインストーラ200を含まずに、第二のインストーラ210のみを含んでもよい(不図示)。

【0097】

以上、ショートカット存在の有無に応じたショートカット作成手法について説明した。これにより、サービスパックのインストーラが単体で実行された場合、サービスパックのインストール中はショートカットが削除され、インストール処理の最後にショートカットが作成されるようになる。そのため、ユーザがサービスパックのインストール中にショートカットからアプリケーションソフトウェアを起動することがなくなる。その結果、インストール中におけるアプリケーションソフトウェアの起動操作を抑止することができるようになる。

30

【0098】

(他の実施例)

本発明は、上述した実施形態を適宜組み合わせることにより構成された装置あるいはシステムやその方法も含まれるものとする。

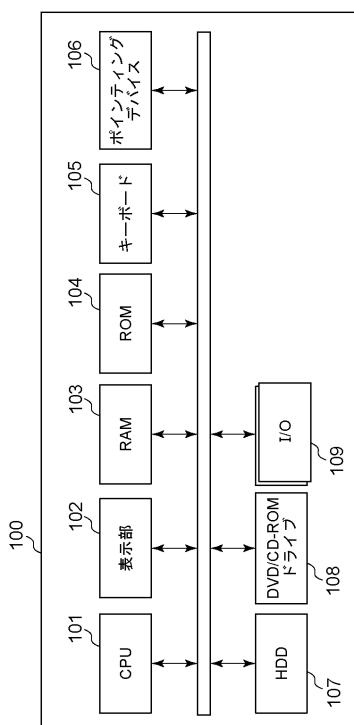
【0099】

ここで、本発明は、上述した実施形態の機能を実現する1つ以上のソフトウェア(プログラム)を実行する主体となる装置あるいはシステムである。また、その装置あるいはシステムで実行される上述した実施形態を実現するための方法も本発明の1つである。また、そのプログラムは、ネットワークまたは各種記憶媒体を介してシステムあるいは装置に供給され、そのシステムあるいは装置の1つ以上のコンピューター(CPUやMPU等)によりそのプログラムが読み出され、実行される。つまり、本発明の1つとして、さらにそのプログラム自体、あるいは当該プログラムを格納したコンピュータにより読み取り可能な各種記憶媒体も含むものとする。また、上述した実施形態の機能を実現する回路(例えば、ASIC)によっても、本発明は実現可能である。

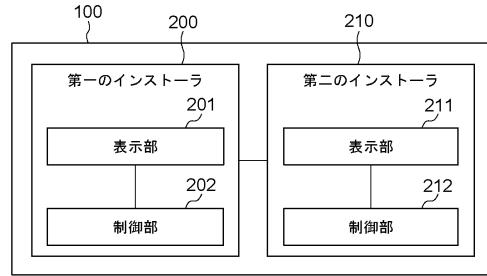
40

【図面】

【図 1】



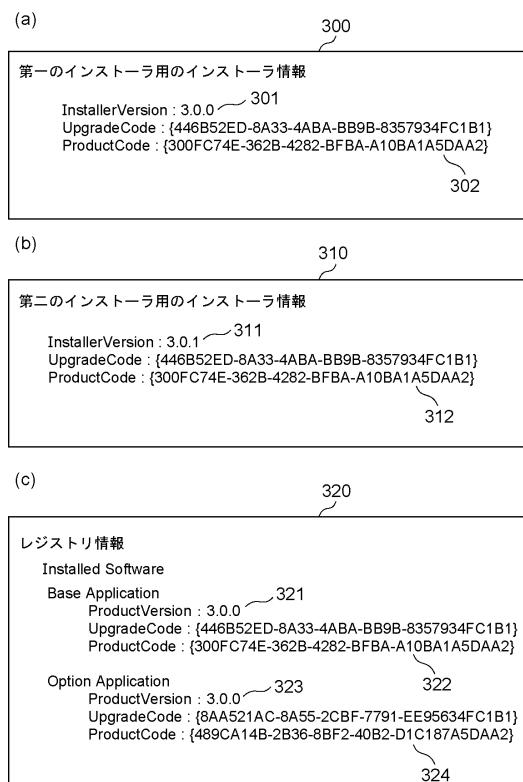
【図 2】



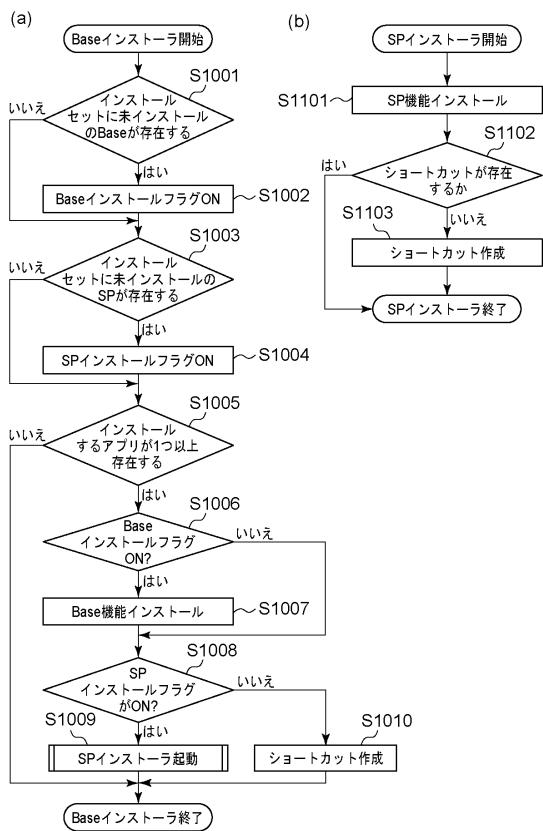
10

20

【図 3】



【図 4】

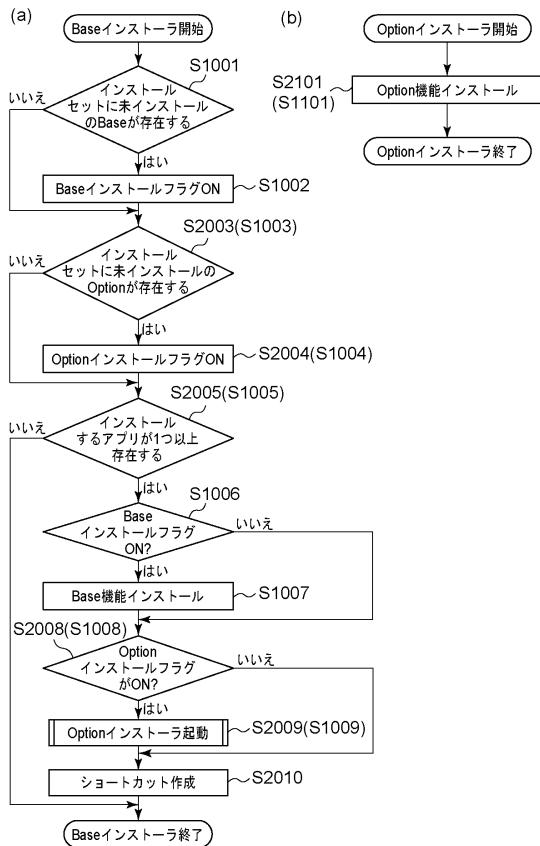


30

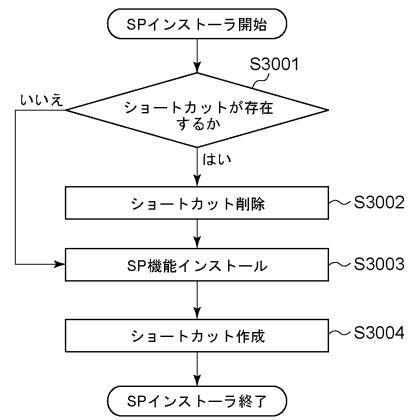
40

50

【図 5】



【図 6】



10

20

30

40

50

フロントページの続き

(56)参考文献 特開2013-214247 (JP, A)
米国特許第08910140 (US, B1)

(58)調査した分野 (Int.Cl., DB名)
G 0 6 F 8 / 0 0 - 8 / 3 8
G 0 6 F 8 / 6 0 - 8 / 7 7
G 0 6 F 9 / 4 4 - 9 / 4 4 5
G 0 6 F 9 / 4 5 1