

(19) 日本国特許庁(JP)

(12) 公表特許公報(A)

(11) 特許出願公表番号

特表2006-517694

(P2006-517694A)

(43) 公表日 平成18年7月27日(2006.7.27)

(51) Int. Cl.		F I			テーマコード (参考)
G06F 11/00	(2006.01)	G06F	9/06	630A	5B076
G06F 9/445	(2006.01)	G06F	9/06	640A	5K067
H04B 7/26	(2006.01)	H04B	7/26	M	
		H04B	7/26	Z	

審査請求 未請求 予備審査請求 有 (全 16 頁)

(21) 出願番号 特願2004-563942 (P2004-563942)
 (86) (22) 出願日 平成15年12月19日 (2003.12.19)
 (85) 翻訳文提出日 平成17年8月9日 (2005.8.9)
 (86) 国際出願番号 PCT/US2003/040926
 (87) 国際公開番号 W02004/059438
 (87) 国際公開日 平成16年7月15日 (2004.7.15)
 (31) 優先権主張番号 60/435,486
 (32) 優先日 平成14年12月20日 (2002.12.20)
 (33) 優先権主張国 米国 (US)
 (31) 優先権主張番号 60/435,828
 (32) 優先日 平成14年12月20日 (2002.12.20)
 (33) 優先権主張国 米国 (US)
 (31) 優先権主張番号 10/740,355
 (32) 優先日 平成15年12月18日 (2003.12.18)
 (33) 優先権主張国 米国 (US)

(71) 出願人 595020643
 クアルコム・インコーポレイテッド
 QUALCOMM INCORPORATED
 アメリカ合衆国、カリフォルニア州 92
 121-1714、サン・ディエゴ、モア
 ハウス・ドライブ 5775
 (74) 代理人 100058479
 弁理士 鈴江 武彦
 (74) 代理人 100091351
 弁理士 河野 哲
 (74) 代理人 100088683
 弁理士 中村 誠
 (74) 代理人 100108855
 弁理士 蔵田 昌俊

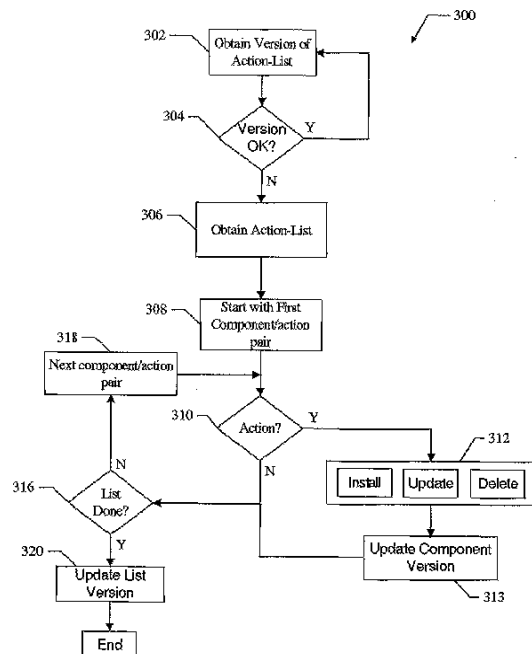
最終頁に続く

(54) 【発明の名称】 デバイス上のコンポーネントを自動的に処理するシステム

(57) 【要約】

【課題】

【解決手段】 デバイス上のコンポーネントを自動的に処理するための方法が提供される。この方法は、アクションリストに関連したバージョン識別子を受信することと、前記バージョン識別子が、格納されたバージョン識別子と等価ではないことを判定することと、前記アクションリストを受信することと、前記アクションリストを解析し、コンポーネント識別子と、関連するアクションとを取得することと、前記コンポーネント識別子によって識別されたコンポーネントについて前記アクションを実行することを含む。



【特許請求の範囲】**【請求項 1】**

デバイス上のコンポーネントを自動的に処理する方法であって、
アクションリストに関連するバージョン識別子を受信することと、
前記バージョン識別子が、格納されたバージョン識別子と等価ではないことを判定することと、
前記アクションリストを受信することと、
前記アクションリストを解析して、コンポーネント識別子と、関連するアクションとを取得することと、
前記コンポーネント識別子によって識別されたコンポーネントについて前記アクションを実行することとを備える。 10

【請求項 2】

請求項 1 に記載の方法において、前記格納された識別子を、前記バージョン識別子に等価になるように更新することを更に備える。

【請求項 3】

請求項 1 に記載の方法において、前記アクションはインストールすることであり、前記実行するステップは、前記デバイス上に前記コンポーネントをインストールすることを備える。

【請求項 4】

請求項 1 に記載の方法において、前記アクションは削除することであり、前記実行するステップは、前記デバイスから前記コンポーネントを削除することを備える。 20

【請求項 5】

請求項 1 に記載の方法において、前記実行するステップは、前記デバイス上の前記コンポーネントの状態を変更することを備える。

【請求項 6】

請求項 1 に記載の方法において、前記解析するステップは、前記コンポーネント識別子に関連したコンポーネントバージョンが、格納されたコンポーネントバージョンと等価ではないことを判定することを更に備える。

【請求項 7】

請求項 6 に記載の方法において、前記格納されたコンポーネントバージョンを、前記コンポーネントバージョンに等価になるように更新することを更に備える。 30

【請求項 8】

請求項 1 に記載の方法において、前記バージョン識別子、前記アクションリスト、及び前記コンポーネントをサーバから受信することを更に備える。

【請求項 9】

請求項 1 に記載の方法において、前記デバイスは無線デバイスである。

【請求項 10】

デバイス上のコンポーネントを自動的に処理する装置であって、
アクションリストに関連するバージョン識別子を受信するロジックと、
前記バージョン識別子を、格納されたバージョン識別子と比較するロジックと、 40
前記バージョン識別子が前記格納されたバージョン識別子と等価でなければ、前記アクションリストを受信するロジックと、
前記アクションリストを解析して、コンポーネント識別子と、関連するアクションとを取得するロジックと、
前記コンポーネント識別子によって識別されたコンポーネントについて前記アクションを実行するロジックとを備える。

【請求項 11】

請求項 10 に記載の装置において、前記格納された識別子を、前記バージョン識別子に等価になるように更新するロジックを更に備える。

【請求項 12】

請求項 10 に記載の装置において、前記アクションを実行するロジックは、前記デバイス上に前記コンポーネントをインストールするロジックを備える。

【請求項 13】

請求項 10 に記載の装置において、前記アクションを実行するロジックは、前記デバイスから前記コンポーネントを削除するロジックを備える。

【請求項 14】

請求項 10 に記載の装置において、前記アクションを実行するロジックは、前記デバイス上の前記コンポーネントの状態を変更するロジックを備える。

【請求項 15】

請求項 10 に記載の装置において、前記アクションリストを解析するロジックは、前記コンポーネント識別子に関連したコンポーネントバージョンが、格納されたコンポーネントバージョンと等価ではないことを判定するロジックを更に備える。

10

【請求項 16】

請求項 15 に記載の装置において、前記格納されたコンポーネントバージョンを、前記コンポーネントバージョンに等価になるように更新するロジックを更に備える。

【請求項 17】

請求項 10 に記載の装置において、前記バージョン識別子、前記アクションリスト、及び前記コンポーネントをサーバから受信するロジックを更に備える。

【請求項 18】

請求項 10 に記載の装置において、前記デバイスは無線デバイスである。

20

【請求項 19】

デバイス上のコンポーネントを自動的に処理する装置であって、
アクションリストに関連するバージョン識別子を受信する手段と、
前記バージョン識別子が、格納されたバージョン識別子と等価ではないことを判定する手段と、

前記アクションリストを受信する手段と、

前記アクションリストを解析して、コンポーネント識別子と、関連するアクションとを取得する手段と、

前記コンポーネント識別子によって識別されたコンポーネントについて前記アクションを実行する手段とを備える。

30

【請求項 20】

請求項 19 に記載の装置において、前記格納された識別子を、前記バージョン識別子に等価になるように更新する手段を更に備える。

【請求項 21】

請求項 19 に記載の装置において、前記実行する手段は、前記デバイス上に前記コンポーネントをインストールする手段を備える。

【請求項 22】

請求項 19 に記載の装置において、前記実行する手段は、前記デバイスから前記コンポーネントを削除する手段を備える。

【請求項 23】

請求項 19 に記載の装置において、前記実行する手段は、前記デバイス上の前記コンポーネントの状態を変更する手段を備える。

40

【請求項 24】

請求項 19 に記載の装置において、前記解析する手段は、前記コンポーネント識別子に関連したコンポーネントバージョンが、格納されたコンポーネントバージョンと等価ではないことを判定する手段を更に備える。

【請求項 25】

請求項 24 に記載の装置において、前記格納されたコンポーネントバージョンを、前記コンポーネントバージョンに等価になるように更新する手段を更に備える。

【請求項 26】

50

請求項 19 に記載の装置において、前記バージョン識別子、前記アクションリスト、及び前記コンポーネントをサーバから受信する手段を更に備える。

【請求項 27】

請求項 19 に記載の装置において、前記デバイスは無線デバイスである。

【請求項 28】

デバイス内のロジックを処理することによって実行された時、前記デバイス上のコンポーネントを自動的に処理するように動作するコンピュータ読取可能媒体であって、

アクションリストに関連するバージョン識別子を受信する命令と、

前記バージョン識別子が、格納されたバージョン識別子と等価ではないことを判定する命令と、

前記アクションリストを受信する命令と、

前記アクションリストを解析して、コンポーネント識別子と、関連するアクションとを取得する命令と、

前記コンポーネント識別子によって識別されたコンポーネントについて前記アクションを実行する命令とを備える。

【請求項 29】

請求項 28 に記載のコンピュータ読取可能媒体において、前記格納された識別子を、前記バージョン識別子に等価になるように更新する命令を更に備える。

【請求項 30】

請求項 28 に記載のコンピュータ読取可能媒体において、前記実行する命令は、前記デバイス上に前記コンポーネントをインストールする命令を備える。

【請求項 31】

請求項 28 に記載のコンピュータ読取可能媒体において、前記実行する命令は、前記デバイスから前記コンポーネントを削除する命令を備える。

【請求項 32】

請求項 28 に記載のコンピュータ読取可能媒体において、前記実行する命令は、前記デバイス上の前記コンポーネントの状態を変更する命令を備える。

【請求項 33】

請求項 28 に記載のコンピュータ読取可能媒体において、前記解析する命令は、前記コンポーネント識別子に関連したコンポーネントバージョンが、格納されたコンポーネントバージョンと等価ではないことを判定する命令を更に備える。

【請求項 34】

請求項 33 に記載のコンピュータ読取可能媒体において、前記格納されたコンポーネントバージョンを、前記コンポーネントバージョンと等価になるように更新する命令を更に備える。

【請求項 35】

請求項 28 に記載のコンピュータ読取可能媒体において、前記バージョン識別子、前記アクションリスト、及び前記コンポーネントをサーバから受信する命令を更に備える。

【請求項 36】

請求項 28 に記載のコンピュータ読取可能媒体において、前記デバイスは無線デバイスである。

【発明の詳細な説明】

【技術分野】

【0001】

本発明は、一般に、デバイス上のコンポーネントの処理に関し、更に詳しくは、デバイス上のコンポーネントを自動的にインストールし、更新し、又は削除するシステムに関する。

【背景技術】

【0002】

データネットワークは、様々なネットワーク可能なデバイスを用いて、多数のユーザが

10

20

30

40

50

互いに通信するための方法を提供する。例えば、無線通信ネットワークでは、様々なネットワーク可能なポータブル電話によって、ユーザは、極めて遠い距離を介して互いに通信することが可能となる。ネットワーク可能なデバイスは、一般に、このデバイスがどのように動作するか、そして究極的には、ネットワーク全体の性能がどのように良くなるのかを制御するインストールされた様々なコンポーネントを用いて設定される。

【0003】

ある環境では、ネットワークオペレータは、ネットワーク可能なデバイスの設定を、アップグレード、インストール、削除、さもなければ変更することを望んでいる。例えば、システムソフトウェアが改善されると、ネットワークオペレータは、このネットワークをより効率的に動作させるために、ネットワーク内の全てのデバイスにインストールされた新たなコンポーネントを持ちたいと思う。例えば、ネットワークオペレータは、デバイス上にアプリケーションソフトウェア、バイナリ実行可能な、あるいはその他の情報をインストールし、サービス改善又はアップグレードを提供したいと思うかもしれない。別の状況では、ネットワークオペレータは、特定の種類のデバイスの必要性に基づいて、企業アプリケーション又はその他のデバイスアップグレードアプリケーションをインストールすることを望むかもしれない。更に、もしも現在インストールされているシステムコンポーネントに問題が存在すると判定された場合、デバイスユーザがこの問題が存在することを気付いているかに関わらず、ネットワークオペレータは、この問題を修復するために、アップグレードされたコンポーネントをインストールしたいと思うであろう。例えば、もしもユーザがマルチメディアコンテンツを見ることを可能にする現在のデバイスコンポーネントに問題が存在するのであれば、ネットワークオペレータは、このコンポーネントを、問題の無い新たなコンポーネントに交換するために、ネットワーク上の全てのデバイスをアップグレードしたいと思うであろう。

【0004】

デバイスにおけるコンポーネントをインストール、削除、あるいは変更するために使用される一つの技術は、デバイスユーザが、このデバイスを修理センタに返還することを待つことである。一旦修理センタに行くと、このデバイスが最も最近のコンポーネントバージョンを持つことができるように、修理要員が、このデバイスを再プログラムすることができる。あいにく、もしもデバイスユーザがこの問題に気がつかなければ、あるいは、彼らがこのデバイスをどうやって使用するのかにこの問題が現在影響を与えていなければ、デバイスユーザはこのデバイスを返還することを忘れるかもしれないのでこの処理は非常に非効率的である。デバイスの全てがアップグレードされる訳ではないので、幾つかのデバイスは、最良の性能を与えるように動作しないであろうし、ネットワークの全体動作が低下するかもしれない。

【0005】

従って、必要なことは、コンポーネントの選択されたバージョンがインストールされ、動作されるようにデバイス上のコンポーネントを自動的に処理するためのシステムである。このシステムは、比較的短い時間内で、多数のデバイスにおけるコンポーネントを処理できるように十分に柔軟であるべきである。これによって、ネットワーク上で動作する全てのデバイスに対して高速なアップグレードを提供し、もって、最良のデバイス性能と、増大したネットワーク効率との結果が得られる。

【発明の開示】

【0006】

本出願は、2002年12月20日に出願され"AUTO-INSTALL"と題された係属中の米国仮出願60/435,486号の優先権の利益を主張する。この出願は、全ての目的のためにその全体を参照することにより組み込まれる。

【0007】

本出願はまた、2002年12月20日に出願され"REGISTRY-BASED AUTO INSTALL COMPONENT HANDLING"と題された係属中の米国仮出願60/435,828号の優先権の利益を主張する。この出願は、全ての目的のためにその全体を参照することにより組み込まれ

10

20

30

40

50

る。

【0008】

一つ又は複数の実施例では、デバイス上のコンポーネントを自動的に処理するためにシステムが提供される。例えば、一つの実施例では、このシステムによって、デバイスは、ダウンロードサーバ上で利用可能なバージョン化されたアクションリストを用いて、デバイスにおけるコンポーネントのインストール、更新、削除、起動、ディセーブル、リコール、あるいはこのコンポーネントの状態の変更が可能となる。

【0009】

処理されることが可能なコンポーネントは、例えばアプリケーション、実行可能な設定情報、ユーザインタフェース設定、ランダムデータ、又はその他の種類の情報など、任意の種類でありうる。

10

【0010】

このシステムの動作中、デバイスは、ダウンロードサーバ上のアクションリストのバージョンを、デバイスによって処理された最後のアクションリストに関連した格納されたバージョンと比較する。もしも2つのバージョンが異なれば、デバイスは、サーバから新たなアクションリストをダウンロードし、このアクションリスト内の各項目を解析し、デバイス上のコンポーネントを処理する。一つの実施例では、このアクションリスト内の各項目は、コンポーネント識別子をアクション識別子に関連付けているコンポーネント/アクションペアを備えている。

【0011】

このコンポーネント識別子は、コンポーネントの種類と、その現在のバージョンとを識別する。このアクション識別子は、識別されたコンポーネントに関連するデバイスによって実行されるアクションを識別する。

20

【0012】

一つの実施例では、デバイスは、アクションリスト内のコンポーネント/アクションペアのそれぞれを解析し、識別されたコンポーネントのために実行されるアクションが必要あるかを判定する。

【0013】

このデバイスは、アクションリスト内のコンポーネントのバージョンを、デバイスに格納されたコンポーネントバージョンと比較する。もしも2つのコンポーネントバージョンが同一であれば、デバイスは、コンポーネント/アクションペアに関して何のアクションもとらない。もしもこれらバージョンが異なるのであれば、デバイスは、アクションリスト内のコンポーネントに関連付けられたアクションを実行する。例えば、もしもこのアクションが、識別されたコンポーネントをインストールすることであれば、デバイスはこのコンポーネント、及びその他の必要な情報をダウンロードサーバからダウンロードし、このコンポーネントをデバイス上にインストールする。従って、このデバイスは、新たなコンポーネントバージョンにおける指定された動作のみを実行するアクションリストを介して進む。

30

【0014】

特定のコンポーネント/アクションペアを処理した後、このデバイスに格納されたコンポーネントバージョンは、アクションリスト内に与えられた新たなコンポーネントバージョンを用いて更新される。アクションリスト全体が処理されたとき、現在のアクションリストの将来における処理を再び行うことを回避するために、デバイスは、アクションリストのバージョン識別子を記録する。従って、ダウンロードサーバ上で利用可能なアクションリストのバージョンが、デバイス上に格納されたバージョンと異なるまで、このデバイスは、別のアクションリストを処理しないであろう。

40

【0015】

一つの実施例では、デバイスがダウンロードサーバと通信するときにその都度、デバイスが、アクションリストのバージョンをチェックする。例えば、デバイスの電源がオンにされた後、又は定期的な間隔において、デバイスがダウンロードサーバと通信する。この

50

システムは、何のユーザ相互作用もなく、あるいは制限されたユーザ相互作用もなく、コンポーネントがデバイス上で処理されるようになるメカニズムを提供する。例えば、一つの実施例では、電源投入時にコンポーネントがデバイスにプッシュされ、「サイレントインストール」を効果的に提供する。別の実施例では、ユーザのオプション時にコンポーネント処理となるようにユーザインタフェースが提供される。従って、このシステムは、コンポーネント処理が強制されたり、促進されたり、要求されたり、オプションとされるようにする。

【0016】

このシステムは、単一デバイス上、又は多数のデバイス上のコンポーネントを処理するために使用される。例えば、多数のデバイスが、アクションリストサーバにアクセスすることができるデータネットワークでは、単一のアクションリストが全てのデバイスに配信され、各デバイスは、どのコンポーネントがそのデバイスのために処理されるのかを判定することができる。別の実施例では、サーバは、異なるデバイス種類のために使用することが可能な複数のアクションリストを提供する。例えば、異なる種類のデバイスは、異なるアクションリストにアクセスし、そのデバイス種類に対するコンポーネントを処理する。従って、このシステムが多数のデバイスに対してグローバルな更新を与えることが可能である。ここでは、各デバイスがアクションリストサーバにコンタクトするので、時間又は日の期間にわたって更新が実行される。

10

【0017】

更に、サーバとデバイスとの間の通信は、例えば暗号化や任意の種類 of 符号化のような任意の種類 of 安全な通信技術を使って実行される。これによって、デバイスが認証され、情報の任意の送信が安全な方式でなされるようになる。

20

【0018】

一つの実施例では、デバイス上のコンポーネントを自動的に処理するための方法が提供される。この方法は、アクションリストに関連したバージョン識別子を受信することと、前記バージョン識別子が、格納されたバージョン識別子と等価ではないことを判定することと、前記アクションリストを受信することと、前記アクションリストを解析し、コンポーネント識別子と、関連するアクションとを取得することと、前記コンポーネント識別子によって識別されたコンポーネントについて前記アクションを実行することとを備える。

30

【0019】

一つの実施例では、デバイス上のコンポーネントを自動的に処理する装置が提供される。この装置は、アクションリストに関連したバージョン識別子を受信するロジックと、前記バージョン識別子を、格納されたバージョン識別子と比較するロジックとを備える。またこの装置は、もしもバージョン識別子が、前記格納されたバージョン識別子と等価ではないのであれば、アクションリストを受信するロジックと、前記アクションリストを解析し、コンポーネント識別子と、関連するアクションとを取得するロジックとを備える。またこの装置は、前記コンポーネント識別子によって識別されるコンポーネントにおいてアクションを実行するロジックを備えている。

【0020】

一つの実施例では、デバイス上のコンポーネントを自動的に処理するための装置が提供される。この装置は、アクションリストに関連付けられたバージョン識別子を受信する手段と、前記バージョン識別子が、格納されたバージョン識別子と等価ではないことを判定する手段とを備える。またこの装置は、前記アクションリストを受信する手段と、前記アクションリストを解析し、コンポーネント識別子と、関連するアクションとを取得する手段とを備える。またこの装置は、コンポーネント識別子によって識別されたコンポーネントにおいて前記アクションを実行する手段を備える。

40

【0021】

一つの実施例では、コンピュータ読取可能媒体が提供される。この媒体は、デバイス内の処理ロジックによって実行されたときに、デバイス上のコンポーネントを自動的に処理するように動作する命令を備えている。このコンピュータ読取可能媒体は、アクションリ

50

ストに関連付けられたバージョン識別子を受信する命令と、前記バージョン識別子が、格納されたバージョン識別子と等価ではないことを判定する命令とを備えている。またこのコンピュータ読取可能媒体は、前記アクションリストを受信する命令と、前記アクションリストを解析し、コンポーネント識別子と、関連するアクションとを取得する命令とを備える。またこのコンピュータ読取可能媒体は、前記コンポーネント識別子によって識別されたコンポーネントにおいて前記アクションを実行する命令を備えている。

【0022】

本発明のその他の局面、利点、及び特徴は、以下の図面の簡単な説明、本発明の詳細な説明、及び特許請求の範囲の説明のレビューの後に明らかになるであろう。

【0023】

ここに記載された実施例の前述した局面及び付随する利点は、添付図面と連携してなされた以下に示す詳細記述を参照することによって容易に明らかになるであろう。

【発明を実施するための最良の形態】

【0024】

以下に示す詳細記述は、デバイス上のコンポーネントを自動的に処理するシステムの実施例を示す。このシステムは、限定される訳ではないが、通信ネットワーク、例えばインターネットのような公衆ネットワーク、例えば仮想プライベートネットワークのようなプライベートネットワーク、ローカルエリアネットワーク、ワイドエリアネットワーク、長距離ネットワーク、又はその他任意の種類 of データネットワークを含む有線又は無線の任意のネットワークにおいて使用されるのに適している。またこのシステムは、コンポーネントをダウンロード及びインストールすることが可能な任意の種類 of デバイスと共に使用するのに適している。例えば、このシステムは、オフィスコンピュータ、ノートブックコンピュータ、及び例えばポータブル電話、PDA、又はコンポーネントを受信及びインストールすることが可能なその他任意の種類 of デバイスのようなハンドヘルドデバイスとともに使用することに適している。

【0025】

一つ又は複数の実施例では、このシステムは、例えばデバイス特有のリソースのために一般化されたコールを提供することによって、デバイスの動作を簡単にするために使用されるデバイス上で実行しているランタイム環境と相互作用する。そのようなランタイム環境は、カリフォルニア州、サンディエゴのQUALCOMM社によって開発されたBinary Runtime Environment for Wireless (BREW (登録商標)) ソフトウェアプラットフォームである。以下の記述において、このデバイスは、例えばBREWソフトウェアプラットフォームのようなランタイム環境を実行していると仮定される。しかしながら、このシステムの一つ又は複数の実施例は、種々の有線及び無線デバイス上のコンポーネントを自動的に処理する他の種類のランタイム環境とともに使用されるのに適している。

【0026】

図1は、デバイス上のコンポーネントを自動的に処理するシステム100の一つの実施例を示す。このシステム100は、サーバ102、データネットワーク104、及びデバイス106を備えている。データネットワーク104は、サーバ102とデバイス106との間で情報が通信されるようにする任意の種類 of 有線又は無線ネットワークでありうる。

【0027】

例えば、ネットワーク104は、通信ネットワーク、ワイドエリアネットワーク、仮想プライベートネットワーク、又は例えばインターネットのような公衆ネットワークでありうる。

【0028】

一つ又は複数の実施例では、このシステムは、デバイス106上のコンポーネントを処理するために動作する。

【0029】

例えば、サーバ102は、コンポーネント112、アクションリスト110、及びアク

10

20

30

40

50

ションリストバージョン識別子108を含む。このバージョン識別子108は、アクションリスト110の現在のバージョンを識別する。アクションリスト110は、コンポーネント/アクションペアを備えている。ここで、各ペアは、コンポーネント識別子とアクション識別子とを備える。このコンポーネント識別子は、特定のコンポーネントを識別するユニークな項目IDである。コンポーネント識別子の一部は、コンポーネントのバージョンを識別するために使用される。例えば、コンポーネント識別子は、ビットのシリーズである。このビットの一部は、コンポーネントのバージョンを識別するために使用される。アクション識別子は、例えばインストール、更新、削除、リコール、ディセーブル、又はデバイス106がこの識別されたコンポーネントに関連して実行するその他のアクションのような特定のアクションを識別する。

10

【0030】

動作の間、サーバ102は、アクションリストバージョン識別子108を、ネットワーク104を経由してデバイス106に送信する。例えば、デバイス106は、電源投入シーケンスの間、サーバ102とコンタクトし、サーバ102は、アクションリストバージョン識別子108をデバイス106に送信することによって応答する。デバイス106は、バージョン識別子108を、デバイス106によって処理される最後のアクションリストのバージョンを表す格納されたバージョン識別子114と比較する。もしもダウンロードされたバージョン108と、格納されたバージョン114とが等しければ、デバイスは、デバイス上の既存のコンポーネント116に対して如何なるインストールもまた変更も行う必要がない。しかしながら、もしもこのダウンロードされたバージョン108が、格納されたバージョン114と異なるのであれば、デバイス106は、追加のコンポーネント処理が必要であることを知る。例えば、デバイス106上にインストールされる必要がある新たなコンポーネントがサーバ102に位置しているかもしれない。

20

【0031】

一旦デバイス106がアクションリストの新たなバージョンが存在していることを検出すると、新たなアクションリスト110を取得するためにサーバ102に要求を送信する。サーバ102は、アクションリスト110をデバイス106に送信することによって応答する。そしてデバイス106は、アクションリスト110を処理し、このアクションリスト110内で識別されたコンポーネントのインストール、更新、削除、あるいは処理を行う。

30

【0032】

このデバイス106は、各コンポーネント/アクションペアを通り、各コンポーネントに関してアクションを行うか否かを判定することによってこのアクションリスト110を処理するように動作する。例えば、もしもコンポーネントがインストール又は更新される必要があるとアクションリストが特定すれば、デバイス106は、コンポーネント112及びその他任意の必要なファイルをサーバ102からダウンロードし、それをデバイス106にインストールする。例えば、サーバ102からデバイス106へ送信された情報は、一般に120で示される。

【0033】

もしもコンポーネントが削除される必要があるとアクションリスト110が特定すれば、デバイス106はこのコンポーネントを削除する。従って、アクションリスト110内の各コンポーネント/アクションペアは、例えばこの識別されたコンポーネントを自動的にインストール、更新、削除するためにデバイス106によって処理される。アクションリストは任意の長さであり、このアクションリストを処理した後、デバイス106は、格納されたアクションリストバージョン識別子114を更新し、最も最近に処理されたアクションリストのバージョン(すなわちバージョン108)を格納する。

40

【0034】

一つの実施例では、デバイス106が、アクションリスト110内のコンポーネント/アクションペアを解析し、特定のコンポーネントに関連するアクションを講じる必要があるか否かを判定する。例えば、デバイス106は、インストールされたコンポーネントの

50

最も新しいバージョンを既に持っているかもしれない。もしそうであれば、このデバイスは、そのコンポーネントを再インストールする必要はない。一つの実施例では、アクションリスト 110 内のコンポーネント識別子は、コンポーネントのバージョンを判定するための情報を含んでいる。例えば、このバージョン情報は、コンポーネント識別子の終わりに追加することができる。デバイス 106 は、格納されたコンポーネントバージョン 118 を、アクションリスト 110 内にダウンロードされたコンポーネントバージョンと比較する。もしも、格納されたコンポーネントバージョンと、ダウンロードされたコンポーネントバージョンとが同一であれば、デバイスは、そのコンポーネントに関していかなるアクションをとる必要もない。しかしながら、もしも、格納されたコンポーネントバージョンと、ダウンロードされたコンポーネントバージョンとが同一ではないのであれば、デバイス 106 は、アクションリスト内のそのコンポーネントに関連するアクションを実行する。従って、デバイス 106 は、新たなバージョンのコンポーネントを単に処理するだけなので、システムは非常に効率的である。

10

【0035】

図 2 は、デバイス 200 の機能図を示す。これは、デバイス 200 上のコンポーネントを自動的に処理するシステムの一つの実施例を含む。このデバイスは、処理ロジック 208 と、比較ロジック 206 と、機能ロジック 214 と、バージョン更新ロジック 224 と、格納されたアクションリストバージョン識別子 204 と、コンポーネントバージョン識別子 212 と、インストールされたコンポーネント 226 とを備える。デバイス 200 によって提供されたロジック及び機能は、ハードウェア、ソフトウェア、又はハードウェアとソフトウェアとの組み合わせによって実現することができる。例えば、一つ又は複数の実施例では、処理ロジック 208 は、CPU と、プロセッサと、ゲートアレイと、ハードウェアロジックと、メモリコンポーネントと、仮想マシンと、ソフトウェアと、あるいはハードウェア及びソフトウェアの任意の組み合わせとを備える。

20

【0036】

従って、処理ロジック 208 は一般に、機械読取可能命令を実行し、ここに記載した機能を実現するロジックを備えている。デバイス 200 は、単に一つの実施例を示しており、デバイス要素の変更、追加、又は再配置は、本発明の範囲から逸脱することなくなされることが留意されるべきである。

【0037】

図 3 は、例えばデバイス 200 のようなデバイスを動作させ、デバイス 200 上のコンポーネントを自動的に処理するためのシステムを提供する方法 300 の一つの実施例を示す。

30

【0038】

明確さの目的のために、この方法 300 は、図 2 に示すデバイス 200 に関連して記載される。デバイス 200 が、図 1 に示すように、データネットワークを経由してダウンロードサーバと安全に通信していることが更に仮定される。

【0039】

ブロック 302 では、デバイスが、ダウンロードサーバからアクションリストバージョン識別子を取得する。例えば、デバイスが、データネットワークを経由してダウンロードサーバと通信し、202 に示すように、ダウンロードサーバが、アクションリストバージョン識別子をデバイスに送信する。

40

【0040】

ブロック 304 では、テストが行われ、ダウンロードされたアクションリストバージョン識別子が、デバイスによって処理される最後のアクションリストに関連している格納されたバージョン識別子と異なっているかが判定される。例えば、このダウンロードされたバージョン識別子 202 と、格納されたバージョン識別子 204 とが比較ロジック 206 に入力される。比較ロジック 206 は、これら 2 つの識別子を比較し、これらが等価であるかを判定する。もしもこれら 2 つの識別子が等価であれば、本方法はブロック 302 に戻り、別の機会に、新たなバージョンのアクションリストを取得する。もしもこれら 2 つ

50

のバージョン識別子が異なれば (D i f f .)、本方法はブロック 3 0 6 に進む。

【 0 0 4 1 】

ブロック 3 0 6 では、デバイスが、ダウンロードサーバからアクションリストを検索する。例えば、アクションリスト 2 1 0 は、サーバからデータネットワークを経由して処理ロジック 2 0 8 にダウンロードされる。

【 0 0 4 2 】

ブロック 3 0 8 では、アクションリスト内の第 1 のコンポーネント / アクションペアを解析することによってデバイスが、このアクションリストの処理を開始する。例えば、処理ロジック 2 0 8 は、ダウンロードされたアクションリスト 2 1 0 を処理してコンポーネント / アクションペアを解析するように動作する。

10

【 0 0 4 3 】

ブロック 3 1 0 では、テストが行われ、必要であれば、現在処理されているコンポーネント / アクションペアのために、何のアクションが必要であるかが判定される。一つの実施例では、デバイスは、ブロック 3 1 2 に進むことによって、このアクションを自動的に実行するように動作する。

【 0 0 4 4 】

しかしながら、これは、デバイス上に再インストールされている既存のコンポーネントという結果になるかもしれない。別の実施例では、コンポーネントのバージョンがチェックされ、アクションが必要であるかが判定される。例えば、このアクションが、コンポーネントを「インストール」することであれば、このコンポーネントのバージョンがチェックされ、このデバイスが、現在インストールされているコンポーネントのバージョンを持っているかが確認される。従って、本方法は、デバイス上に既にインストールされたコンポーネントを再インストールすることを回避するよう動作する。例えば、処理ロジック 2 0 8 は、格納されたコンポーネントバージョン識別子 2 1 2 を検索し、それを、アクションリスト内で識別されたコンポーネントのバージョンと比較する。一つの実施例では、コンポーネントのバージョンが、アクションリスト内に提供されたコンポーネント識別子に組み込まれる。もしもこれら 2 つのコンポーネントバージョンが同一であれば、そのコンポーネントに関して更なるアクションは必要とされず、本方法はブロック 4 1 6 に進む。もしもこれら 2 つのコンポーネントバージョンが異なれば、処理ロジック 2 0 8 は、コンポーネントに関連したアクションを実施するように動作し、本方法はブロック 3 1 2 に進む。

20

30

【 0 0 4 5 】

ブロック 3 1 2 では、現在のコンポーネント / アクションペア内のコンポーネントに関連したアクションが実行され、識別されたコンポーネントの状態が変更される。例えば、処理ロジック 2 0 8 は、アクションロジック 2 1 4 を制御し、識別されたコンポーネントのインストール、更新、削除、起動、ディセーブル、リコール、又はこのコンポーネントの状態の変更のアクションを実行するように動作する。例えば、ソフトリコールは、コンポーネントがデバイスから削除されたが、関連するデータ及び / 又はライセンス情報が除去されない場合に実行されうる。例えば、もしもこのアクションが、コンポーネントをインストール又は更新するものであれば、処理ロジック 2 0 8 が、データネットワークを経由してダウンロードサーバからコンポーネント 2 2 2 をダウンロード (又は更新) するように動作する。このダウンロードされたコンポーネントは、インストールされたコンポーネント 2 2 6 としてインストールされる。処理ロジック 2 0 8 は、あらゆる種類のインストール又は更新処理を実施し、ダウンロードされたコンポーネント 2 2 2 を、インストールされたコンポーネント 2 2 6 としてインストール又は更新する。もしもこのアクションが、コンポーネントを削除することであれば、処理ロジック 2 0 8 は、削除ロジック 2 2 0 を制御して、インストールされたコンポーネント 2 2 6 から、識別されたコンポーネントを削除する。

40

【 0 0 4 6 】

図 2 及び図 3 には示されていないが、例えばインストール、更新、削除、リコール、起

50

動、及び非起動など事実上いかなる種類のアクションであっても、コンポーネントに関して実行される。

【0047】

ブロック314では、コンポーネントバージョンリストが更新され、コンポーネントの新たなバージョンがインストール又は更新されたこと、あるいは、コンポーネントが削除されたことが反映される。

【0048】

例えば、処理ロジック208が、バージョン更新ロジック224を制御して、格納されたコンポーネントバージョン識別子212を、現在処理されているコンポーネントに関する新たな情報を用いて更新する。

【0049】

ブロック316では、テストが行われ、アクションリスト内のコンポーネント/アクションペアの全てが処理されたかが判定される。もしも全てのペアが処理されたならば、本方法はブロック320に進む。もしも全てのペアが処理されたわけではないのであれば、本方法はブロック318に進み、次のペアがブロック310における処理のためにアクセスされる。アクションリスト210は、任意の長さでありうるので、処理されるコンポーネント/アクションペアは任意数存在する。

【0050】

ブロック320では、このデバイスで格納されたアクションリストバージョン識別子が更新される。

【0051】

例えば、処理ロジック208が、更新バージョンロジック224を制御して、格納されたアクションリストバージョン識別子204を、最も最近処理されたアクションリストに関連する識別子を用いて更新する。従って、新たなバージョンのアクションリストが利用可能になるまで、このシステムは、別のアクションリストを処理するように動作することはない。

【0052】

一つの実施例では、デバイス上のコンポーネントを自動的に処理するシステムは、処理ロジック208によって実行されたときに、ここに記載された機能を提供するコンピュータ読取可能媒体に格納されたプログラム命令を備える。

【0053】

例えば、命令は、例えばフロッピー（登録商標）ディスク、CDROM、メモリカード、フラッシュメモリデバイス、RAM、ROM、又はデバイス200にインタフェースする任意の種類メモリデバイスやコンピュータ読取可能媒体のようなコンピュータ読取可能媒体からデバイス200へとロードされる。別の実施例では、この命令は、データネットワークを經由してデバイス200にインタフェースするネットワークリソースからデバイス200へとロードされうる。この命令は、処理ロジック208によって実行されたとき、ここに記載したようなデバイス上のコンポーネントを自動的に処理するシステムの一つ又は複数の実施例を提供する。

【0054】

本方法300は、ほんの一つの実施例を示しており、本発明要素の変形、追加、再配置は、本発明の範囲を逸脱することなくなされることに留意されたい。

【0055】

図4は、トランザクション400を示す。このトランザクションは、デバイス上のコンポーネントを自動的に処理するシステムの一つの実施例の動作中、デバイスとダウンロードサーバとの間で起こる。例えば、トランザクション400は、図1に示すサーバ102とデバイス106との間で起こる。

【0056】

この自動処理の開始において、402に示すように、デバイス106が、サーバ102から、アクションリストの最新のバージョン識別子を要求する。このアクションリストは

10

20

30

40

50

、識別された各コンポーネントに関してデバイスが実行すべきアクションを記述するコンポーネント/アクションペアを含む。このアクションリストは、定期的に変更又は更新され、アクションリストバージョン識別子は、このアクションリストの現在のバージョンを識別する。

【0057】

サーバ102は、404に示すように、現在のアクションリストのバージョン識別子を送信することによって、デバイス102からの要求に対して応答する。アクションリストバージョン識別子を受信した後に、デバイスは、この識別子を、格納されたアクションリストバージョン識別子と比較する。もしもこれら2つのバージョン識別子が等価であれば、デバイスは更なるアクションを講じない。もしもこれら2つのバージョン識別子が異なっていれば、406に示すように、デバイス102は、サーバ102からの新たなアクションリストを要求する。

10

【0058】

サーバ102は、408に示すように、新たなアクションリストを送信することによって、デバイス106からの要求に対して応答する。デバイス106は、このアクションリスト内の各コンポーネント/アクションペアを処理し、特定のコンポーネントのインストール、更新、又は削除をするかしないかを判定する。もしも特定のコンポーネントのインストール又は更新が必要であるとデバイス106が判定したのであれば、410に示すように、デバイス106が、サーバ102からコンポーネントを要求(または更新)する。

【0059】

サーバ102は、要求されたコンポーネントをデバイス106に送信することによってこの要求に応答する。デバイス106は、このコンポーネントを受信し、必要に応じてインストール又は更新を行う。コンポーネントは、デバイスがローカルに格納するコンポーネントバージョン識別子を持っている。デバイス106がアクションリスト全体を解析し、必要に応じてサーバ102から必要な全てのコンポーネントを検索した後に、デバイス106は、丁度処理されたアクションリストのバージョンを用いて、ローカルに格納されたアクションリストを更新する。従って、サーバ102からダウンロードされたアクションリストバージョン識別子が、格納された識別子と異なるまで、デバイス106は、サーバ102からの別のアクションリストを処理するであろう。

20

【0060】

図5は、デバイス上のコンポーネントを自動的に処理するシステムにおいて使用するためのアクションリスト500の一つの実施例を示す。アクションリスト500は、アクションリストバージョン識別子502を含む。アクションリストバージョン識別子502の後には、コンポーネント/アクションペア(504, 506)が続く。例えば、コンポーネント/アクションペア504は、コンポーネント識別子と、対応するアクションとを含む。一つの実施例では、コンポーネント識別子はまた、コンポーネントバージョン識別子をも含んでいる。これによって、コンポーネントのバージョンが、デバイス上に現在コンポーネントが存在しているか否かを判定するために使用されることが可能となる。このアクションは、「インストール」、「更新」、又は「削除」のうちの一つでありうるが、その他任意のアクションが特定されることも可能である。アクションリスト内の情報は、任意の適切なフォーマットを用いて符号化され、アクション識別子は、任意の種類でありうる。このように、デバイス上のコンポーネントを自動的に処理するシステムの一つ又は複数の実施例が説明され、記載されたが、これら精神又は本質的な特徴から逸脱することなく種々の変更が本実施例に対してなされるものと理解されよう。よって、この開示及び記述は、特許請求の範囲に記述された本発明の範囲を限定するものではなく、例示することを意図している。

30

40

【図面の簡単な説明】

【0061】

【図1】図1は、デバイス上のコンポーネントを自動的に処理するシステムの一つの実施例を示す。

50

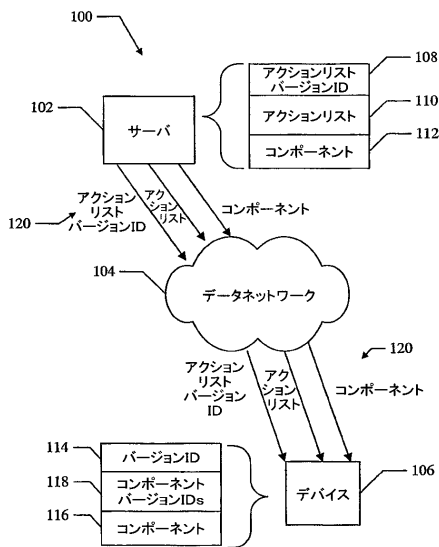
【図2】図2は、自動的にコンポーネントを処理するシステムの一つの実施例を含むデバイスの機能ブロック図である。

【図3】図3は、デバイス上のコンポーネントを自動的に処理するシステムを提供するデバイスを動作させる方法の一つの実施例を示す。

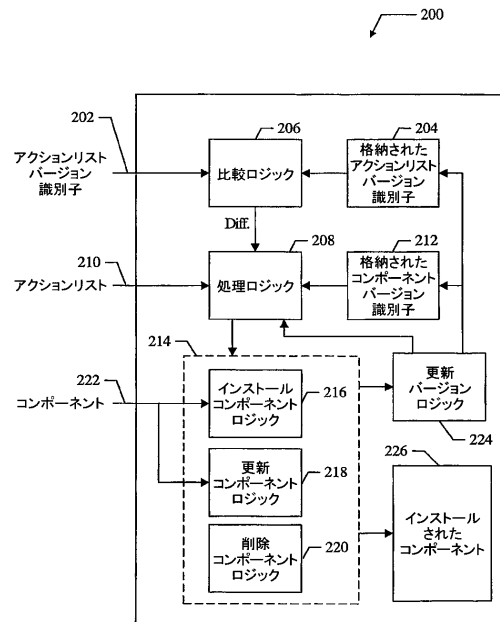
【図4】図4は、デバイス上のコンポーネントを自動的に処理するシステムの一つの実施例の動作の間に、ダウンロードサーバとデバイスとの間で起こるトランザクションを示す。

【図5】図5は、デバイス上のコンポーネントを自動的に処理するシステムで使用されるアクションリストの一つの実施例を示す。

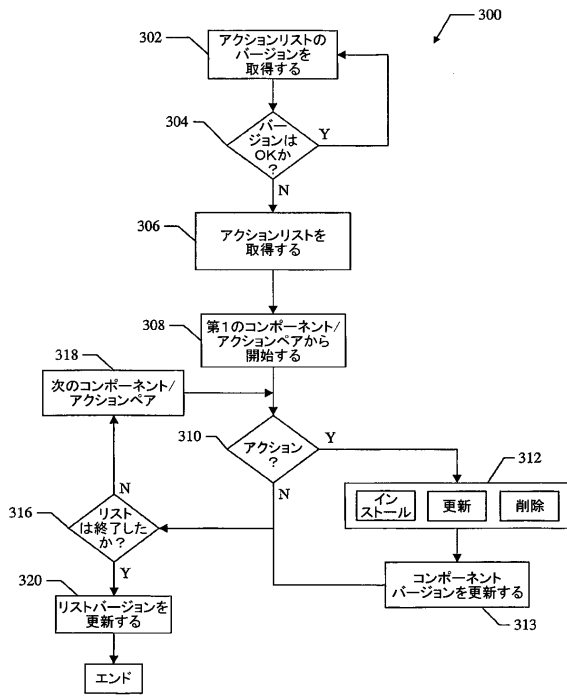
【図1】



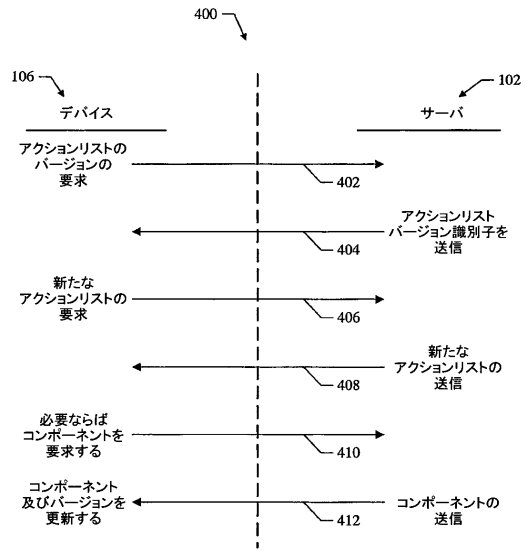
【図2】



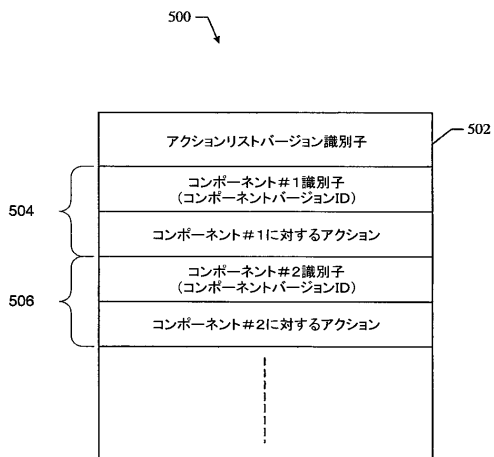
【 図 3 】



【 図 4 】



【 図 5 】



フロントページの続き

(81) 指定国 AP(BW, GH, GM, KE, LS, MW, MZ, SD, SL, SZ, TZ, UG, ZM, ZW), EA(AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), EP(AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HU, IE, IT, LU, MC, NL, PT, RO, SE, SI, SK, TR), OA(BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, ML, MR, NE, SN, TD, TG), AE, AG, AL, AM, AT, AU, AZ, BA, BB, BG, BR, BW, BY, BZ, CA, CH, CN, CO, CR, CU, CZ, DE, DK, DM, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, HR, HU, ID, IL, IN, IS, JP, KE, KG, KP, KR, KZ, LC, LK, LR, LS, LT, LU, LV, MA, MD, MG, MK, MN, MW, MX, MZ, NI, NO, NZ, OM, PG, PH, PL, PT, RO, RU, SC, SD, SE, SG, SK, SL, SY, TJ, TM, TN, TR, TT, TZ, UA, UG, UZ, VC, VN, YU, ZA, ZM, ZW

(74) 代理人 100075672

弁理士 峰 隆司

(74) 代理人 100109830

弁理士 福原 淑弘

(74) 代理人 100095441

弁理士 白根 俊郎

(74) 代理人 100084618

弁理士 村松 貞男

(74) 代理人 100103034

弁理士 野河 信久

(74) 代理人 100092196

弁理士 橋本 良郎

(74) 代理人 100100952

弁理士 風間 鉄也

(72) 発明者 スプリング、スティーブン・エー・

アメリカ合衆国、カリフォルニア州 9 2 0 6 4、ポウエイ、トラバーティン・コート 1 2 1 2
4

(72) 発明者 ミニア、ブライアン

アメリカ合衆国、カリフォルニア州 9 2 1 2 8、サン・ディエゴ、フォンタネル・プレイス 1
3 7 0 4

F ターム(参考) 5B076 AB10 AB20 AC01 AC03 BB06 EA02 EA03 EB02

5K067 AA34 BB04 BB21 DD27 DD51 EE02 EE10 EE16 FF02 KK13

KK15