

(19) 日本国特許庁(JP)

(12) 特 許 公 報(B2)

(11) 特許番号

特許第3760904号
(P3760904)

(45) 発行日 平成18年3月29日(2006.3.29)

(24) 登録日 平成18年1月20日(2006.1.20)

(51) Int. Cl.

F I

G O 6 F 9/44 (2006.01)

G O 6 F 9/06 6 2 O C

G O 6 F 3/048 (2006.01)

G O 6 F 3/00 6 5 8 A

請求項の数 5 (全 23 頁)

(21) 出願番号 特願2002-261354 (P2002-261354)
 (22) 出願日 平成14年9月6日(2002.9.6)
 (65) 公開番号 特開2004-102498 (P2004-102498A)
 (43) 公開日 平成16年4月2日(2004.4.2)
 審査請求日 平成16年4月21日(2004.4.21)

(73) 特許権者 000002185
 ソニー株式会社
 東京都品川区北品川6丁目7番35号
 (74) 代理人 100093241
 弁理士 宮田 正昭
 (74) 代理人 100101801
 弁理士 山田 英治
 (74) 代理人 100086531
 弁理士 澤田 俊夫
 (72) 発明者 鈴木 宏紀
 東京都品川区北品川6丁目7番35号 ソ
 ニー株式会社内
 (72) 発明者 森脇 研一
 東京都品川区北品川6丁目7番35号 ソ
 ニー株式会社内

最終頁に続く

(54) 【発明の名称】 G U I アプリケーション開発支援装置、G U I 表示装置、および方法、並びにコンピュータ・プログラム

(57) 【特許請求の範囲】

【請求項1】

グラフィカル・ユーザインタフェース(G U I)を構成するコンポーネントの機能および表示態様の設定情報を持つG U I アプリケーションの開発支援を行なうG U I アプリケーション開発支援装置であり、

コンポーネントのグラフィックス表現に関するプログラムを格納したグラフィックス表現部品を記憶したグラフィックス表現部品記憶部と、

前記グラフィックス表現部品において規定されるメソッドに従って読み出し可能なアニメーション・イメージデータを格納したイメージ記憶部と、

前記コンポーネントの状態に応じて読み出すイメージデータの対応付け処理を実行する編集処理部とを有し、

前記グラフィックス表現部品記憶部は、複数の異なるグラフィックス表現処理実行プログラムを格納した複数のグラフィックス表現部品記憶部からなり、

各グラフィックス表現部品記憶部に記憶されたグラフィックス表現部品は、それぞれ個別の読み出しイメージファイルの対応付けがなされた構成であり、

前記編集処理部は、複数の異なるコンポーネントの状態各々に対応付けて、異なる読み出しイメージデータを設定可能な構成を有することを特徴とするG U I アプリケーション開発支援装置。

【請求項2】

前記G U I アプリケーション開発支援装置は、さらに、

10

20

コンポーネントの機能に関するプログラムを格納した論理部品を記憶した論理部品記憶部を有し、

前記編集処理部は、

前記論理部品記憶部から抽出した論理部品と、前記グラフィックス表現部品記憶部から抽出したグラフィックス表現部品とを組み合わせ、GUIコンポーネントを生成する構成であることを特徴とする請求項1に記載のGUIアプリケーション開発支援装置。

【請求項3】

グラフィカル・ユーザインタフェース（GUI）を構成するコンポーネントの機能および表示態様の設定情報を持つGUIアプリケーションをGUIアプリケーション開発支援装置において生成するGUIアプリケーションプログラム生成方法であり、

10

前記GUIアプリケーション開発支援装置における編集処理部の実行ステップであり、コンポーネントのグラフィックス表現に関するプログラムを格納し、それぞれ個別の読み出しイメージファイルの対応付けがなされたグラフィックス表現部品を記憶したグラフィックス表現部品記憶部からグラフィックス表現部品を選択抽出するグラフィックス表現部品選択ステップと、

前記GUIアプリケーション開発支援装置における編集処理部の実行ステップであり、前記グラフィックス表現部品選択ステップにおいて選択したグラフィックス表現部品に対応するコンポーネントの状態に応じて異なる読み出しイメージデータの対応付け処理を実行する編集処理ステップであり、複数の異なるコンポーネントの状態各々に対応付けて、前記グラフィックス表現部品において規定されるメソッドに従って読み出すアニメーション・イメージデータを、個別に設定する編集処理ステップと、

20

を有することを特徴とするGUIアプリケーションプログラム生成方法。

【請求項4】

前記GUIアプリケーションプログラム生成方法は、さらに、

前記GUIアプリケーション開発支援装置における編集処理部の実行ステップとして、コンポーネントの機能に関するプログラムを格納した論理部品を記憶した論理部品記憶部から論理部品を読み出すステップを有し、

前記編集処理部の実行する前記編集処理ステップは、

前記論理部品記憶部から抽出した論理部品と、前記グラフィックス表現部品記憶部から抽出したグラフィックス表現部品とを組み合わせ、GUIコンポーネントを生成するステップを含むことを特徴とする請求項3に記載のGUIアプリケーションプログラム生成方法。

30

【請求項5】

グラフィカル・ユーザインタフェース（GUI）を構成するコンポーネントの機能および表示態様の設定情報を持つGUIアプリケーションプログラム生成処理をGUIアプリケーション開発支援装置において実行させるコンピュータ・プログラムであって、

前記GUIアプリケーション開発支援装置における編集処理部において、コンポーネントのグラフィックス表現に関するプログラムを格納し、それぞれ個別の読み出しイメージファイルの対応付けがなされたグラフィックス表現部品を記憶したグラフィックス表現部品記憶部からグラフィックス表現部品を選択抽出するグラフィックス表現部品選択ステップと、

40

前記編集処理部において、前記グラフィックス表現部品選択ステップにおいて選択したグラフィックス表現部品に対応するコンポーネントの状態に応じて異なる読み出しイメージデータの対応付け処理を実行する編集処理ステップであり、複数の異なるコンポーネントの状態各々に対応付けて、前記グラフィックス表現部品において規定されるメソッドに従って読み出すアニメーション・イメージデータを、個別に設定する編集処理ステップと、

を実行させることを特徴とするコンピュータ・プログラム。

【発明の詳細な説明】

【0001】

50

【発明の属する技術分野】

本発明は、GUIアプリケーション開発支援装置、GUI表示装置、および方法、並びにコンピュータ・プログラムに関する。さらに詳細には、グラフィカル・ユーザインタフェース（GUI）に設定されるボタン、リスト、コンボボックス等、各コンポーネントを視覚効果の高いアニメーション表現の実行コンポーネントとして構成することを可能としたGUIアプリケーション開発支援装置、GUI表示装置、および方法、並びにコンピュータ・プログラムに関する。

【0002】**【従来の技術】**

PC、通信端末、あるいは様々な情報家電等においては、データ表示あるいはデータ入出力用ユーザインタフェースとしてCRT、LCD等のディスプレイが使用されている。このようなディスプレイには、例えば情報家電の利用形態に応じた機能を設定した画面イメージがグラフィカル・ユーザインタフェース（GUI）として表示される。GUIには、例えば入力ボタンの配置、データ入力欄の設定、あるいはボタン押下時の処理等、様々な機能が設定され、また様々な表示画面のイメージ設定が行なわれる。このようなユーザインタフェース画面のデザイン、機能を設定するためのアプリケーションプログラムを開発するためのプログラム言語として、Sun Microsystems, Inc. により配布されているJavaが多く利用されている。

【0003】

Java言語で書かれたプログラムは、コンピュータ、プラットフォームに依存しないJava実行ファイル形式としてのバイトコード（Byte Code）に変換され、Java仮想マシン（Java VM）により読み取られて機械語変換（インタプリタ）がなされ様々な装置上において実行される。

【0004】

Javaにおいては、アプリケーションソフトウェアを容易に短時間で製作するために、再コンパイルの不要なソフトウェア部品（コンポーネント）をあらかじめ用意しておき、その部品を組み合わせることでプログラムを作成する。Javaにおいては、これらの部品の再利用化技術（Java Beans）が構築されている。

【0005】

部品は、例えば、ディスプレイに表示するウィンドウ、ボタン、リスト、コンボボックス、テキスト入力フィールド等の様々なコンポーネントを単位として設定され、これら部品（コンポーネント）は、例えばボタン形状、配置情報等の属性情報としての「プロパティ」、部品に対するアクションの結果としての処理としての「メソッド」、さらに、部品に対するアクション、例えばデータ到着、割り込み発生、プロパティ変更、メソッド呼び出し等が発生した際に、別の部品に事象を連絡する機能としての「イベント」が規定される。これら、「プロパティ」、「メソッド」、「イベント」は、Java beansにおける部品の構成要件とされる。

【0006】

Javaの実行最小プログラム単位は、「クラス（class）」と呼ばれ、1以上のクラスによってJavaプログラムが生成される。クラスはデータとしての変数（Variable）と、動作としてのメソッド（Method）を持つ。このクラスとして設定されるプログラムコンポーネントを、例えばインターネット、LANのようなネットワークを介してPC、情報家電に転送し、格納することが可能であり、クラスファイルを格納した装置側では、プラットフォームに依存しないJava仮想マシンが、クラスファイル内に格納されたプログラムを実行することができる。

【0007】

前述したように、GUIを構成するコンポーネント、例えば、ボタン、リスト、コンボボックス、テキスト入力フィールド等の様々なコンポーネントは、プロパティとしてコンポーネントの形状、配置情報等の属性情報を有し、さらにメソッド、イベント等の機能情報が定義された部品として設定される。

10

20

30

40

50

【 0 0 0 8 】

すなわち、グラフィカル・ユーザインタフェース上に設定されるボタン等のコンポーネント（GUI部品）の機能、例えばオンされたときの処理やオフされたときの処理等の機能を定義している論理部分と、コンポーネントの表示イメージ、すなわちグラフィックス表現部分とは一体化され、GUIを設計するプログラミング処理においては、論理部分と、グラフィックス表現部分とが一体化されたコンポーネントを選択して設定することによる様々なアプリケーションの設計が行なわれている。

【 0 0 0 9 】

従って、例えば、コンポーネントとしてのボタンをオンとしたとき、あるいはオフとしたときの表示処理としてアニメーションを設定する場合には、基本的に固定的なアニメーションの設定となり、これらを容易に変更することは困難であった。

10

【 0 0 1 0 】

しかし、GUIを搭載する機器によって最適なグラフィックイメージは異なり、機器によって設定したいアニメーションも異なる場合がある。例えば子供等が操作する機器においてはアニメーションを多用し、ビジネス向け機器においては、シンプルとするなどである。また、機器自体のメモリ、CPU等の性能によっても最適なグラフィックス表現が異なる場合がある。

【 0 0 1 1 】

【 発明が解決しようとする課題 】

本発明は、上記問題点に鑑みてなされたものであり、グラフィカル・ユーザインタフェース（GUI）に設定される各コンポーネントに対して、様々なアニメーションを自在に設定することを可能とし、機器に応じて異なるグラフィックス表現としての異なるアニメーションを設定することを可能としたGUIアプリケーション開発支援装置、GUI表示装置、および方法、並びにコンピュータ・プログラムを提供することを目的とする。

20

【 0 0 1 2 】

【 課題を解決するための手段 】

本発明の第1の側面は、

グラフィカル・ユーザインタフェース（GUI）を構成するコンポーネントの機能および表示態様の設定情報を持つGUIアプリケーションの開発支援を行なうGUIアプリケーション開発支援装置であり、

30

コンポーネントのグラフィックス表現に関するプログラムを格納したグラフィックス表現部品を記憶したグラフィックス表現部品記憶部と、

前記グラフィックス表現部品において規定されるメソッドに従って読み出し可能なアニメーション・イメージデータを格納したイメージ記憶部と、

前記コンポーネントの状態に応じて読み出すイメージデータの対応付け処理を実行する編集処理部とを有し、

前記グラフィックス表現部品記憶部は、複数の異なるグラフィックス表現処理実行プログラムを格納した複数のグラフィックス表現部品記憶部からなり、

各グラフィックス表現部品記憶部に記憶されたグラフィックス表現部品は、それぞれ個別の読み出しイメージファイルの対応付けがなされた構成であり、

40

前記編集処理部は、複数の異なるコンポーネントの状態各々に対応付けて、異なる読み出しイメージデータを設定可能な構成を有することを特徴とするGUIアプリケーション開発支援装置にある。

【 0 0 1 3 】

さらに、本発明のGUIアプリケーション開発支援装置の一実施態様において、前記GUIアプリケーション開発支援装置は、さらに、コンポーネントの機能に関するプログラムを格納した論理部品を記憶した論理部品記憶部を有し、前記編集処理部は、前記論理部品記憶部から抽出した論理部品と、前記グラフィックス表現部品記憶部から抽出したグラフィックス表現部品とを組み合わせ、GUIコンポーネントを生成する構成であることを特徴とする。

50

【 0 0 1 4 】

さらに、本発明の第 2 の側面は、

グラフィカル・ユーザインタフェース (G U I) を構成するコンポーネントの機能および表示態様の設定情報を持つ G U I アプリケーションを G U I アプリケーション開発支援装置において生成する G U I アプリケーションプログラム生成方法であり、

前記 G U I アプリケーション開発支援装置における編集処理部の実行ステップであり、コンポーネントのグラフィックス表現に関するプログラムを格納し、それぞれ個別の読み出しイメージファイルの対応付けがなされたグラフィックス表現部品を記憶したグラフィックス表現部品記憶部からグラフィックス表現部品を選択抽出するグラフィックス表現部品選択ステップと、

前記 G U I アプリケーション開発支援装置における編集処理部の実行ステップであり、前記グラフィックス表現部品選択ステップにおいて選択したグラフィックス表現部品に対応するコンポーネントの状態に応じて異なる読み出しイメージデータの対応付け処理を実行する編集処理ステップであり、複数の異なるコンポーネントの状態各々に対応付けて、前記グラフィックス表現部品において規定されるメソッドに従って読み出すアニメーション・イメージデータを、個別に設定する編集処理ステップと、

を有することを特徴とする G U I アプリケーションプログラム生成方法にある。

【 0 0 1 5 】

さらに、本発明の G U I アプリケーションプログラム生成方法の一実施態様において、前記 G U I アプリケーションプログラム生成方法は、さらに、前記 G U I アプリケーション開発支援装置における編集処理部の実行ステップとして、コンポーネントの機能に関するプログラムを格納した論理部品を記憶した論理部品記憶部から論理部品を読み出すステップを有し、前記編集処理部の実行する前記編集処理ステップは、前記論理部品記憶部から抽出した論理部品と、前記グラフィックス表現部品記憶部から抽出したグラフィックス表現部品とを組み合わせ、G U I コンポーネントを生成するステップを含むことを特徴とする。

【 0 0 1 6 】

さらに、本発明の第 3 の側面は、

グラフィカル・ユーザインタフェース (G U I) を構成するコンポーネントの機能および表示態様の設定情報を持つ G U I アプリケーションプログラム生成処理を G U I アプリケーション開発支援装置において実行させるコンピュータ・プログラムであって、

前記 G U I アプリケーション開発支援装置における編集処理部において、コンポーネントのグラフィックス表現に関するプログラムを格納し、それぞれ個別の読み出しイメージファイルの対応付けがなされたグラフィックス表現部品を記憶したグラフィックス表現部品記憶部からグラフィックス表現部品を選択抽出するグラフィックス表現部品選択ステップと、

前記編集処理部において、前記グラフィックス表現部品選択ステップにおいて選択したグラフィックス表現部品に対応するコンポーネントの状態に応じて異なる読み出しイメージデータの対応付け処理を実行する編集処理ステップであり、複数の異なるコンポーネントの状態各々に対応付けて、前記グラフィックス表現部品において規定されるメソッドに従って読み出すアニメーション・イメージデータを、個別に設定する編集処理ステップと、

を実行させることを特徴とするコンピュータ・プログラムにある。

【 0 0 2 5 】

【 作用 】

本発明の構成によれば、グラフィカル・ユーザインタフェース (G U I) に設定される各コンポーネントのアニメーションを自在に設定することが可能となり、G U I 搭載機器に応じた最適なコンポーネント・アニメーションの設定を容易に行なうことが可能となる。すなわち、コンポーネントに含まれるグラフィックス表現部品のグラフィックス表現クラスにより実行されるアニメーションインデックス・メソッドに従って、イメージファイル

10

20

30

40

50

から、予め設定したイメージデータを読み込んで表示する構成としたので、コンポーネントの状態に応じて設定されたイメージによってアニメーションが実行可能となり、GUIコンポーネントの状態に応じた視覚的効果の高いアニメーション表現が可能となる。

【0026】

さらに、本発明では、コンポーネントの論理部分と、グラフィックス表現部分とを分離して構成し、論理部分と、グラフィックス表現部分とに基づいて、GUIコンポーネントを生成する構成とし、グラフィックス表現部分に、さらに、コンポーネント状態に応じたイメージファイルに基づくアニメーション実行機能を持たせたので、論理部分とは独立した処理により、アニメーションの設定態様を容易に変更することが可能となる。

【0027】

なお、本発明のコンピュータ・プログラムは、例えば、様々なプログラム・コードを実行可能な汎用コンピュータ・システムに対して、コンピュータ可読な形式で提供する記憶媒体、通信媒体、例えば、CDやFD、MOなどの記憶媒体、あるいは、ネットワークなどの通信媒体によって提供可能なコンピュータ・プログラムである。このようなプログラムをコンピュータ可読な形式で提供することにより、コンピュータ・システム上でプログラムに応じた処理が実現される。

【0028】

本発明のさらに他の目的、特徴や利点は、後述する本発明の実施例や添付する図面に基づく、より詳細な説明によって明らかになるであろう。なお、本明細書においてシステムとは、複数の装置の論理的集合構成であり、各構成の装置が同一筐体内にあるものには限らない。

【0029】

【発明の実施の形態】

以下、本発明のGUIアプリケーション開発支援装置、GUI表示装置、および方法、並びにコンピュータ・プログラムについて、図面を参照しながら詳細に説明する。

【0030】

〔1. ハードウェア構成〕

まず、本発明のGUIアプリケーション開発支援装置のハードウェア構成例について、図1を参照して説明する。図1を参照しながら、GUIアプリケーション開発支援装置100内の各構成要素について説明する。GUIアプリケーション開発支援装置100のメイン・コントローラであるCPU (Central Processing Unit) 101は、オペレーティング・システム(OS)の制御下で、各種のGUIアプリケーション開発プログラムを実行する。CPU 101は、例えば、Java言語に基づくプログラム作成処理の実行、具体的には、ボタン、コンボボックス等、GUI画面上に設定する各コンポーネントの機能の設定や、表示イメージの設定処理等を行なう。図示の通り、CPU 101は、バス107によって他構成部と相互接続されている。

【0031】

メモリ102は、CPU 101において実行されるプログラム・コードを格納したり、実行中の作業データを一時保管するために使用される記憶装置である。同図に示すメモリ102は、ROMなどの不揮発性メモリ及びDRAMなどの揮発性メモリの双方を含む。

【0032】

グラフィックス処理チップ150は、各種GUIコンポーネントに対する配色処理を実行する専用の処理回路が組み込まれたチップである。

【0033】

表示制御部としてのディスプレイ・コントローラ103は、CPU 101が発行する描画命令を実際に処理するための専用コントローラである。ディスプレイ・コントローラ103において処理された描画データは、例えばフレーム・バッファ(図示しない)に一旦書き込まれた後、ディスプレイ108によって画面出力される。例えば、HDD(111)から再生された画像や、CPU 101により処理された画像は、ディスプレイ108で画面表示されて、ユーザはその提示画面を見ることが出来る。

10

20

30

40

50

【0034】

入力機器インターフェース104は、キーボード109やマウス110などのユーザ入力機器をGUIアプリケーション開発支援装置100に接続するための装置である。ユーザは、キーボード109やマウス110を介して、画像を処理するためや表示するためのコマンドなどを入力することができる。

【0035】

ネットワーク・インターフェース105は、Ethernetなどの所定の通信プロトコルに従って、GUIアプリケーション開発支援装置100をLAN(Local Area Network)などの局所的ネットワーク、さらにはインターネットのような広域ネットワークに接続することができる。

10

【0036】

ネットワーク上では、複数のホスト端末やサーバー(図示しない)がトランスペアレントな状態で接続され、分散コンピューティング環境が構築されている。ネットワーク上では、ソフトウェア・プログラムやデータ・コンテンツなどの配信サービスを行うことができる。例えば、動画像、静止画像等が保存されている他のサーバーから画像データを、ネットワーク経由でHDD(111)へダウンロードすることができる。

【0037】

外部機器インターフェース106は、デジタルカメラや、ハード・ディスク・ドライブ(HDD)111やメディア・ドライブ112などの外部装置をGUIアプリケーション開発支援装置100に接続するための装置である。

20

【0038】

HDD111は、記憶媒体としての磁気ディスクを固定的に搭載した外部記憶装置であり、記憶容量やデータ転送速度などの点で優れており、ランダムアクセスも可能である。例えばソフトウェア・プログラムを実行可能な状態でHDD111上に置くプログラム・インストールが可能である。HDD111には、CPU101が実行すべきオペレーティング・システムのプログラム・コードや、アプリケーション・プログラム、デバイス・ドライバなどが不揮発的に格納されている。

【0039】

メディア・ドライブ112は、CD(Compact Disc)やMO(Magneto-Optical disc)、DVD(Digital Versatile Disc)などの可搬型メディア160を装填して、そのデータ記録面にアクセスするための装置である。

30

【0040】

可搬型メディア160は、主として、ソフトウェア・プログラムやデータ・ファイルなどをコンピュータ可読形式のデータとしてバックアップすることや、これらをシステム間で移動(すなわち販売・流通・配布を含む)する目的で使用される。各種の処理を行うためのアプリケーション・プログラムを、これら可搬型メディアを利用して複数の機器間で物理的に流通・配布することができる。

【0041】

なお、図1に示すような装置は、米IBM社のパーソナル・コンピュータ"PC/AT(Personal Computer/Advanced Technology)"の互換機又は後継機として実現可能である。勿論、他のアーキテクチャを備えたコンピュータを適用することも可能である。

40

【0042】

[2. 機能構成]

次に、グラフィカル・ユーザインタフェース(GUI)に設定される各コンポーネントの論理部分と、グラフィックス表現部分とを分離して構成して、GUIアプリケーションを開発する本発明のGUIアプリケーション開発支援装置の機能構成について図2を参照して説明する。

【0043】

図2に示すように、本発明のGUIアプリケーション開発支援装置は、ユーザインタフェースとしての画面上に設定するボタン、コンボボックス、リスト、ウィンドウ、テキスト

50

ボックス等、様々な部品（コンポーネント）について記憶した部品記憶部 200 を有する。部品記憶部 200 は論理部品記憶部 201 と、グラフィックス表現部品記憶部（シンプル）202 と、グラフィックス表現部品記憶部（リッチ）203 を有する。

【0044】

論理部品記憶部 201 は、ユーザインタフェースとしての画面上に設定するボタン、コンボボックス、リスト、ウィンドウ、テキストボックス等、様々な部品（コンポーネント）についての論理情報を格納した部品である。例えばボタンであれば、ボタンの機能としてのオンされた際の処理やオフされたときの処理等の機能を定義したソフトウェアプログラム部品として構成される。

【0045】

一方、グラフィックス表現部品記憶部（シンプル）202 と、グラフィックス表現部品記憶部（リッチ）203 は、コンポーネントの表示イメージ情報を格納している、すなわちグラフィックス表現プログラムを格納したソフトウェアプログラム部品として構成される。グラフィックス表現部品記憶部（シンプル）202 と、グラフィックス表現部品記憶部（リッチ）203 とは、それぞれ異なるグラフィックス表現を実行するプログラムを格納した部品である。

【0046】

イメージデータ記憶部 205 は、例えば `jpeg`、`gif` 等のイメージデータを格納している。これらのイメージデータを、各コンポーネントのグラフィックス表現部品に対応付けることで、コンポーネントの様々な動作に応じたイメージデータに基づくアニメーション表示が可能となる。これらのアニメーション設定処理については後述する。

【0047】

GUIアプリケーションプログラムの開発者としてのオペレータは、データ入出力部 220 を介して編集処理部 210 に対して、各種のデータ入力を実行し、ボタン、コンボボックス等の各コンポーネントの機能、グラフィックイメージを設定してGUIコンポーネントを構築する。

【0048】

コンポーネントを作成処理実行プログラムは、例えば `Java` プログラムの 1 つのクラス（ファクトリークラス）である。編集処理部 210 はファクトリークラスを実行して各種のコンポーネントを作成する。なお、編集処理部 210 における処理は、図 1 のハードウェア構成図においては、CPU 101 の制御の下に実行される処理である。

【0049】

部品（コンポーネント）は、例えばボタン形状、配置情報等の属性情報としての「プロパティ」、部品に対するアクションの結果としての処理としての「メソッド」、さらに、部品に対するアクション、例えばデータ到着、割り込み発生、プロパティ変更、メソッド呼び出し等が発生した際に、別の部品に事象を連絡する機能としての「イベント」が規定されたGUIコンポーネントとして編集処理部 210 において編集される。

【0050】

編集処理部 210 は、部品（コンポーネント）に設定されるプロパティを編集するプロパティ編集部 211、メソッドを編集するメソッド編集部 212、イベントを編集するイベント編集部 213 を有する。

【0051】

オペレータは、生成するGUIコンポーネントの見た目、すなわちグラフィックス表現を設定する際に、ファクトリークラスを実行する編集処理部 210 に対して、`Look & Feel` パッケージ指定データを入力する。すなわち、「リッチ」あるいは「シンプル」のいずれかを指定する。

【0052】

この指定データに従って、編集処理部 210 においてファクトリークラスが実行され、指定データに応じて、グラフィックス表現部品記憶部（シンプル）202、あるいは、グラフィックス表現部品記憶部（リッチ）203 から、いずれかの対応プログラム部品を読み

10

20

30

40

50

出すとともに、論理部品記憶部 201 の読み出し部品とともにコンポーネントを作成する。

【0053】

例えばボタンについての GUI コンポーネントを生成する際には、部品（コンポーネント）についての論理情報を格納した部品を記憶した論理部品記憶部 201 からボタン部品を読み出し、さらに、オペレータからの Look & Feel パッケージ指定データ、すなわち、「リッチ」あるいは「シンプル」のいずれかの指定データに応じて、グラフィックス表現部品記憶部（シンプル）202、あるいは、グラフィックス表現部品記憶部（リッチ）203 から、ボタンのグラフィックス表現プログラム格納部品を読み出して、論理部品と、「リッチ」あるいは「シンプル」のいずれかのグラフィックス表現部品とを組み合わせ、1つのボタンコンポーネントを生成する。

10

【0054】

ボタン以外にも、コンボボックス、リスト、ウィンドウ、テキストボックス等、様々な部品（コンポーネント）について、同様の編集処理が実行され、編集処理の後、設定された GUI アプリケーションプログラムがプログラム記憶部 221 に格納され、プログラム実行部 222 において実行され、生成された GUI が表示部 223 に表示される。

【0055】

なお、上述の実施例ではグラフィックス表現プログラムを格納した部品を「シンプル」、「リッチ」の2種類のみを設定した例を説明したが、これらの種類は3以上の任意の数として設定することも可能であり、オペレータは、設定されたグラフィックス表現部品のいずれかの指定データを入力し、指定データに従ってグラフィックス表現部品が選択されて、各種の GUI コンポーネントが作成される。

20

【0056】

[3. 処理シーケンス]

次に、本発明の GUI アプリケーション開発支援装置を適用して、GUI コンポーネントを生成する処理の手順について、図3および図4を参照して説明する。

【0057】

図3は、グラフィックス表現部品の「シンプル」を選択して、GUI コンポーネントを生成する処理シーケンスを示した図である。図3において、左から、オペレータ、GUI 生成アプリケーション、コンポーネントファクトリ、グラフィックス表現部分、論理部分の各処理を示している。なお、グラフィックス表現部分、論理部分の処理とは、図2に示すグラフィックス表現部品記憶部（シンプル）202、論理部品記憶部 201 の読み出し部品に格納されたプログラムの実行に相当し、これらの処理、およびコンポーネントファクトリの処理は、図2の編集処理部 210 において実行される。

30

【0058】

まず、ステップ S101 において、オペレータは、GUI コンポーネントの生成コマンド（Create）を GUI 生成アプリケーションに対して入力する。この GUI コンポーネントの生成コマンド（Create）には、コンポーネントの種類、例えば「ボタン」、「リスト」、「コンボボックス」、「ラジオボタン」、「ウィンドウ」などのコンポーネント種類を指定したデータとともに入力される。

40

【0059】

さらに、ステップ S102 において、オペレータからグラフィックス表現を指定するための「Look & Feel」指定がアプリケーションを介してコンポーネントファクトリに対して出力される。ここでは、「Look & Feel」指定として「シンプル」を指定したものとする。

【0060】

次にステップ S103 において、アプリケーションからコンポーネントファクトリに対してコンポーネント生成コマンドが出力される。このコマンドには、先のステップ S101 において指定されたコンポーネント種類を示す識別データが含まれる。

【0061】

50

ステップS 1 0 4では、コンポーネントファクトリからグラフィックス表現部分（シンプル）および論理部分に対してコンポーネント生成命令（C r e a t e）が出力され、論理部分、グラフィックス表現部分の格納プログラムに従って、コンポーネント生成が実行される。

【0062】

ステップS 1 0 5以下の処理は、アプリケーションからコンポーネントの処理、例えばコンポーネントとしての「ウィンドウ」のオープン処理命令を行なった場合の処理シーケンスを示す。アプリケーションからの、コンポーネントのオープン処理命令は、論理部分に渡され、論理部分のプログラムの実行に基づいて、ペイント（P a i n t）命令がグラフィックス表現部分（シンプル）に渡される。グラフィックス表現部分（シンプル）は、論理部分からの入力命令に従って、コンポーネントのペイント処理を実行する。

10

【0063】

ここでは、コンポーネント生成に適用されているグラフィックス表現部分が「シンプル」であるため、ディスプレイに表示される生成コンポーネント、すなわちボタン等は、P a i n tによるカラー設定のなされたのみのシンプル構成となる。

【0064】

次に、グラフィックス表現部品の「リッチ」を選択して、G U Iコンポーネントを生成する処理シーケンスについて図4を参照して説明する。図3と同様、左から、オペレータ、G U I生成アプリケーション、コンポーネントファクトリ、グラフィックス表現部分、論理部分の各処理を示している。図3と同様、グラフィックス表現部分、論理部分の処理とは、図2に示すグラフィックス表現部品記憶部（シンプル）2 0 2、論理部品記憶部2 0 1の読み出し部品に格納されたプログラムの実行に相当し、これらの処理、およびコンポーネントファクトリの処理は、図2の編集処理部2 1 0において実行される。

20

【0065】

まず、ステップS 2 0 1において、オペレータは、G U Iコンポーネントの生成コマンド（C r e a t e）をG U I生成アプリケーションに対して入力する。このG U Iコンポーネントの生成コマンド（C r e a t e）には、コンポーネントの種類、例えば「ボタン」、「リスト」、「コンボボックス」、「ラジオボタン」、「ウィンドウ」などのコンポーネント種類を指定したデータとともに入力される。

【0066】

さらに、ステップS 2 0 2において、オペレータからグラフィックス表現を指定するための「L o o k & F e e l」指定がアプリケーションを介してコンポーネントファクトリに対して出力される。ここでは、「L o o k & F e e l」指定として「リッチ」を指定したものとす。

30

【0067】

次にステップS 2 0 3において、アプリケーションからコンポーネントファクトリに対してコンポーネント生成コマンドが出力される。このコマンドには、先のステップS 2 0 1において指定されたコンポーネント種類を示す識別データが含まれる。

【0068】

ステップS 2 0 4では、コンポーネントファクトリからグラフィックス表現部分（シンプル）および論理部分に対してコンポーネント生成命令（C r e a t e）が出力され、論理部分、グラフィックス表現部分（リッチ）の格納プログラムに従って、コンポーネント生成が実行される。

40

【0069】

ステップS 2 0 5以下の処理は、アプリケーションからコンポーネントの処理、例えばコンポーネントとしての「ウィンドウ」のオープン処理命令を行なった場合の処理シーケンスを示す。アプリケーションからの、コンポーネントのオープン処理命令は、グラフィックス表現部分に渡され、グラフィックス表現部分に格納されたアニメーションプログラムの実行に基づいて、アニメーションが実行（S 2 0 6）される。

【0070】

50

さらに、ステップ S 2 0 7 において、オープン処理命令が論理部分に渡され、論理部分のプログラムの実行に基づいて、ペイント (P a i n t) 命令がグラフィックス表現部分 (シンプル) に渡されて、グラフィックス表現部分 (リッチ) は、論理部分からの入力命令に従って、コンポーネントのペイント処理を実行する。

【 0 0 7 1 】

この図 4 に示すシーケンス図では、コンポーネント生成に適用されているグラフィックス表現部分が「リッチ」であるため、ディスプレイに表示される生成コンポーネント、すなわちボタン等は、P a i n t によるカラー設定がなされるのみならず、アニメーション表現が実行される構成となる。

【 0 0 7 2 】

10

[4 . ファイル構成例]

上述したような、コンポーネント表示を実行するためのファイル構成例について説明する。

【 0 0 7 3 】

例えば G U I コンポーネントの例として、ボタン機能を持つアイコンとして、ボタコン [B u t t c o n] コンポーネントを G U I コンポーネントとして構成する場合のファイルとしては、例えば次の各ファイルが必要となる。

【 0 0 7 4 】

C B u t t c o n . j a v a : ボタコンの論理部分を実装するクラス、
B u t t c o n . j a v a : ボタコンのグラフィックス表現部としての L o o k & F e e 20
l 部分を実装するクラス、
b u t t c o n _ n o r m a l _ o f f . g i f : ボタコンが「オフ」のときの画像、
b u t t c o n _ n o r m a l _ o n . g i f : ボタコンが「オン」のときの画像、
b u t t c o n _ p u s h e d . g i f : ボタコンが押されているときの画像、
C C o m p o n e n t F a c t o r y . j a v a : アプリケーションがボタコンを作成する
ために使用するファクトリークラス、
A p p l i c a t i o n . j a v a : 作成したボタコンを表示するためのアプリケーション。
なお、クラスは、J a v a の実行単位プログラムを示す。

【 0 0 7 5 】

30

上記各ファイルのディレクトリ構成を図 5 に示す。ある G U I アプリケーションファイル [a p p l i c a t i o n] の下位にツールキットとしての [c u s t o m t o o l k i t] ファイルがあり、当ファイルに C B u t t c o n . j a v a : ボタコンの論理部分を実装するクラスと、C C o m p o n e n t F a c t o r y . j a v a : アプリケーションがボタコンを作成するために使用するファクトリークラスが含まれ、さらに、グラフィックス表現情報としての L o o k & F e e e l ファイル [l a f] ファイルが含まれる。

【 0 0 7 6 】

L o o k & F e e e l ファイル [l a f] には、B u t t c o n . j a v a : ボタコンのグラフィックス表現部としての L o o k & F e e l 部分を実装するクラス、すなわち、上述したグラフィックス表現部品に対応するクラスが含まれ、さらに、イメージデータファ 40
イル [i m a g e] として、b u t t c o n _ n o r m a l _ o f f . g i f : ボタコンが「オフ」のときの画像、b u t t c o n _ n o r m a l _ o n . g i f : ボタコンが「オン」のときの画像、b u t t c o n _ p u s h e d . g i f : ボタコンが押されているときの画像、の各々が含まれる。

【 0 0 7 7 】

G U I アプリケーションプログラムは、ユーザの入力情報に従って、これらの各ファイルに含まれる処理プログラムおよびイメージデータを適用した処理を実行する。具体的には、論理部品クラスのプログラムに従った各種の機能実行処理と、グラフィックス表現部品クラスのプログラムに従った各種のグラフィックス表現処理の両者が実行される。

【 0 0 7 8 】

50

〔 5 . 表示例 〕

次に、図 6、および図 7 を参照して、各コンポーネントの論理部分と、グラフィックス表現部分とを分離した構成に基づいて生成した G U I コンポーネントの表示処理例について説明する。

【 0 0 7 9 〕

図 6 は、グラフィックス表現部分（シンプル）を適用して各コンポーネントを生成して表示した例を示す。G U I コンポーネントとしてボタン 3 1 1、コンボボックス 3 1 2 を例にして説明する。

【 0 0 8 0 〕

ボタン 3 1 1 の操作、例えばボタンのオン、オフ、あるいはフォーカス設定、すなわちカーソル設定を行なうと、ボタンコンポーネントの論理部分の処理により、ボタンのオン、オフ動作に基づく処理が実行される。さらに、論理部分からの例えばペイント命令により、グラフィックス表現部分（シンプル）が実行されて、ペイント処理、すなわち、ボタンの表示カラーを変更する処理が行なわれる。

【 0 0 8 1 〕

コンボボックス 3 1 2 の操作、例えばコンボボックス 3 1 2 のオープン、クローズ等を行なうと、コンボボックスコンポーネントの論理部分の処理により、コンボボックスのオープン、クローズ動作に基づく処理が実行される。さらに、論理部分からの例えばペイント命令により、コンボボックスコンポーネントのグラフィックス表現部分（シンプル）が実行されて、ペイント処理、すなわち、コンボボックスの表示カラーを変更する等の処理が行なわれる。

【 0 0 8 2 〕

なお、オペレータからの処理によってどのようなグラフィックス表現を実行するかは、各 G U I コンポーネントに応じて異なる設定とすることが可能である。すなわち、グラフィックス表現部分（シンプル）の処理プログラムを変更することによって、各コンポーネントに対応したグラフィックス表現処理が可能となる。

【 0 0 8 3 〕

図 7 は、グラフィックス表現部分（リッチ）を適用して各コンポーネントを生成して表示した例を示す。G U I コンポーネントとしてボタン 3 2 0、コンボボックス 3 3 0 を例にして説明する。

【 0 0 8 4 〕

ボタン 3 2 0 の操作、例えばボタンのオン、オフ、あるいはフォーカス設定、すなわちカーソル設定を行なうと、前述のペイント処理に加えて、ボタンコンポーネントのグラフィックス表現部分（リッチ）の処理として、アニメーション 3 2 1、3 2 2 の実行がなされる。図では分かり難いが、（ b ）、（ c ）においてボタンの周囲に波紋が広がるようなアニメーションが実行される。なお、論理部分の処理により、ボタンのオン、オフ動作に基づく処理が実行される。

【 0 0 8 5 〕

さらに、コンボボックス 3 3 0 の操作、例えばコンボボックス 3 3 0 のオープン、クローズ等を行なうと、コンボボックスコンポーネントの論理部分の処理により、コンボボックスのオープン、クローズ動作に基づく処理が実行されるとともに、コンボボックスコンポーネントのグラフィックス表現部分（リッチ）の処理として、アニメーション 3 3 1、3 3 2 の実行がなされる。図では分かり難いが、（ b ）、（ c ）においてコンボボックスの下部に下側に広がるように設定されたアニメーションが実行される。

【 0 0 8 6 〕

〔 6 . アニメーション設定および実行処理 〕

次に、ウィンドウ、ボタン、コンボボックス、リスト等の各種の G U I コンポーネントにアニメーションを設定し、実行する処理について説明する。

【 0 0 8 7 〕

図 8 は、コンポーネントに対してアニメーションを実行させるためのイメージデータを設

10

20

30

40

50

定するための処理画面を示した図である。コンポーネント指定欄 351 にコンポーネント [ボタン (Button)] が設定され、下部にイメージデータ設定欄 352 が構成され、コンポーネントの状態、すなわち、通常時 (Normal)、オン (On)、オフ (Off)、フォーカス (Focus) 設定時等の各コンポーネント状態における呼び出しイメージ (Jpeg) が、個別に設定可能な構成を持つ。なお、フォーカス (Focus) 設定時とは、そのコンポーネントにカーソルが設定され、オン、オフ等の動作可能な状態にある状態を示す。

【 0088 】

例えば、オン (On) の処理が実行される際には、グラフィックス表現部品に規定されたプログラムに従ってアニメーションインデックス・メソッドが呼ばれて実行される。アニメーションインデックス・メソッドの実行により、イメージファイル [Image] から Jpeg イメージ 11 ~ 21 が読み込まれて、これらの複数イメージに基づいて GUI 表示されたボタンのイメージがアニメーション表示される。オフ (Off) の処理が実行される際には、イメージファイル [Image] から Jpeg イメージ 31 ~ 38 が読み込まれて、これらの複数イメージに基づいて GUI 表示されたボタンのイメージがアニメーション表示される。

【 0089 】

なお、図 8 に示す例では、コンポーネント状態として通常時 (Normal)、オン (On)、オフ (Off)、フォーカス (Focus) の 4 状態を持つ例を示してあるが、状態定義は、これに限らず、例えば以下に示すような定義とし、それぞれの状態に応じてイメージデータを設定する構成としてもよい。

NORMAL : 通常状態

NORMAL DEACTIVATED : アクティブでない通常状態

FOCUSED : フォーカス状態

FOCUSED DEACTIVATED : アクティブでない通常状態

PUSHED : プッシュ状態

PUSHED DEACTIVATED : アクティブでないプッシュ状態

DISABLED : 操作できない状態

DISABLED DEACTIVATED : アクティブでない操作できない状態。

上記 8 つの状態に応じて、それぞれ異なるアニメーション構成イメージデータを対応付けて設定することで、ユーザはアニメーションに応じてコンポーネントの状態を把握することが可能となる。なお、アクティブでない状態とは、操作対象にないウィンドウに表示されている状態にある場合などを示す。

【 0090 】

このように、コンポーネント毎に、処理に対応して、個別のイメージファイルの設定が可能であり、様々な処理に応じた最適なアニメーション表現を自在に設定することが可能となる。図には、ボタンに対するアニメーションを設定する例を示しているが、その他のコンポーネントについても同様に、各種の処理に対応したアニメーション設定が可能である。

【 0091 】

例えばコンボボックスにおけるオープン時のアニメーション、クローズ時のアニメーション等がイメージファイルからイメージをそれぞれ個別に選択設定することで、それぞれのアニメーションが設定される。

【 0092 】

上述した各コンポーネントの状態に対応するイメージデータの設定によるアニメーション表示設定処理を実行するための GUI アプリケーション開発支援装置は、先に、図 1、図 2 を参照して説明した構成により実現されるが、さらに、各コンポーネント状態に応じたイメージファイルの設定処理を実行するための機能に限定した処理構成を図 9 に示すブロック図を参照して説明する。

【 0093 】

10

20

30

40

50

図 9 に示すように、G U I アプリケーション開発支援装置は、ユーザインタフェースとしての画面上に設定するボタン、コンボボックス、リスト、ウィンドウ、テキストボックス等、様々なコンポーネントについて記憶した部品記憶部 4 0 0 を有し、部品記憶部 4 0 0 は、コンポーネントの処理機能を定義したソフトウェアプログラム部品としての論理部品記憶部 4 0 1 と、コンポーネントのグラフィックス表現を定義したソフトウェアプログラム部品としてのグラフィックス表現部品記憶部 4 0 2 を持つ。なお、グラフィックス表現部品は、前述したようにシンプル、リッチ等、複数種類のグラフィックス表現を実行する複数の異なる部品として構成可能である。

【 0 0 9 4 】

イメージデータ記憶部 4 0 3 は、グラフィックス表現部品記憶部 4 0 2 に記憶されたグラフィックス表現部品に格納されたプログラムとしてのグラフィックス表現クラスの実行により呼び出されるアニメーションインデックス・メソッドに従って読み出されるイメージデータを格納している。イメージデータは、例えば j p e g 、 g i f 等のデータファイルであり、グラフィックス表現部品記憶部 4 0 2 に記憶されたグラフィックス表現部品各々に対応する。

10

【 0 0 9 5 】

G U I アプリケーションプログラムの開発者としてのオペレータは、データ入出力部 4 2 0 を介して編集処理部 4 1 0 に対して、各種のデータ入力を実行し、ボタン、コンボボックス等の各コンポーネントの機能、グラフィックイメージを設定して G U I コンポーネントを構築する。

20

【 0 0 9 6 】

コンポーネント作成処理実行プログラムは、例えば J a v a プログラムの 1 つのクラス (ファクトリークラス) である。編集処理部 4 1 0 はファクトリークラスを実行して各種のコンポーネントを作成する。なお、編集処理部 4 1 0 における処理は、図 1 のハードウェア構成図においては、C P U 1 0 1 の制御の下に実行される処理である。

【 0 0 9 7 】

アニメーションの設定処理においては、図 8 を参照して説明したように、編集処理部 4 1 0 において、コンポーネントの状態に応じて読み出すイメージデータの対応付け処理を実行する。この処理は、コンポーネントの「プロパティ」、「メソッド」、「イベント」の編集処理の一部として実行される。

30

【 0 0 9 8 】

オペレータは、生成する G U I コンポーネントの見た目、すなわちグラフィックス表現を設定する際に、ファクトリークラスを実行する編集処理部 4 1 0 に対して、L o o k & F e e l パッケージ指定データを入力する。すなわち、「リッチ」あるいは「シンプル」のいずれかを指定し、さらに、図 8 を参照して説明したように、コンポーネントの状態に応じて読み出すイメージデータを設定する。

【 0 0 9 9 】

これらの設定情報に従って、各種のコンポーネントを有する G U I アプリケーションプログラムが生成され、生成された G U I アプリケーションプログラムがプログラム記憶部 4 2 1 に格納され、プログラム実行部 4 2 2 において実行され、生成された G U I が表示部 4 2 3 に表示される。

40

【 0 1 0 0 】

次に、アニメーション実行時の処理、すなわち G U I 表示装置における処理について、図 1 0 を参照して説明する。ディスプレイ 5 1 3 に表示された G U I コンポーネントとしてのボタンのアニメーション 5 1 4 の表示処理について説明する。

【 0 1 0 1 】

アニメーション表示処理は、ボタンのコンポーネントに含まれるグラフィックス表現部品のグラフィックス表現クラス 5 1 1 が実行する。グラフィックス表現クラス 5 1 1 に規定されたプログラムに従ってアニメーションインデックス・メソッドが呼ばれ、イメージファイル [I m a g e] 5 1 2 から、先に図 8 を参照して説明した設定処理において設定さ

50

れたJpegイメージ11~21が読込まれて、これらの複数イメージに基づいてGUI表示されたボタンのイメージがアニメーション514として表示される。

【0102】

ここでは、ボタン回りに波紋が広がるアニメーションが表示される例を示している。なお、イメージファイル[Image]512は、図1のハードウェア構成のメモリ102、HDD111、その他の記憶手段の一部領域を確保してイメージファイル記憶部として設定し、格納することが可能である。

【0103】

上述した処理をディスプレイに表示するGUIコンポーネントのアニメーション表示処理方法シーケンスとしてまとめると、

(1)コンポーネントに対応付けられたグラフィックス表現部品のグラフィックス表現クラスを実行するクラス実行ステップと、

(2)クラス実行ステップにおいて呼び出されるアニメーションインデックス・メソッドに従って、イメージファイルから、予め設定したイメージデータを読み出すイメージ読み出しステップと、

(3)イメージ読み出しステップにおいて読み出したイメージデータをディスプレイに表示するステップと、

による実行ステップとしてまとめられる。これらの処理ステップは、GUIアプリケーションの実行プログラムとして記録され、ユーザによるGUI操作に従って実行、すなわち、アニメーション表示がなされることになる。

【0104】

なお、ボタン等のコンポーネントのアニメーションは、呼び出しイメージデータを変更することで、様々なアニメーションが容易に設定できる。例えば、オン、オフ等の動作として、タンが凹凸形状に歪みを発生させたり、色を変化させたりなど、様々なアニメーション設定が可能である。また、オン、オフ等の動作時のみではなく、カーソルが設定されたフォーカス状態にあるとき、継続してアニメーションを実行させる構成とすることも可能である。これは、カーソル位置がボタンコンポーネント位置に重なったことを条件として、アニメーションインデックス・メソッドを呼び込み、フォーカスイメージファイルをイメージデータから読込んで実行させることで実現される。

【0105】

なお、図2を参照して説明したように、グラフィックス表現部品記憶部は、シンプル、あるいはリッチ等、同じコンポーネント、例えばボタンでも、異なるグラフィックス表現処理実行プログラムを格納した複数のグラフィックス表現部品が用意されている。図2の例では、シンプルと、リッチの2つの種類のみを示してあるが、さらにリッチ1、リッチ2、...等、複数のグラフィックス表現部品の設定が可能である。

【0106】

これら各グラフィックス表現部品には、それぞれ異なる個別の読み出しイメージファイルの対応付けを行なうことが可能であり、同じ機能を持つボタンであっても、最初のLook & Feelパッケージ指定により適用するグラフィックス表現部品が決定され、その部品に対応付けて設定されたアニメーションが表示されることになる。従って、Look & Feelパッケージ指定に応じて様々な異なるアニメーション表示を行なうことが可能となる。

【0107】

上述したように、本発明の構成では、グラフィカル・ユーザインタフェース(GUI)に設定される各コンポーネントのアニメーションを自在に設定することが可能となり、GUI搭載機器に応じた最適なコンポーネント・アニメーションの設定を容易に行なうことが可能となる。

【0108】

本発明では、コンポーネントに含まれるグラフィックス表現部品のグラフィックス表現クラスにより実行されるアニメーションインデックス・メソッドに従って、イメージファイ

10

20

30

40

50

ルから、予め設定したイメージデータを読み込んで表示する構成であり、読み出しイメージは、コンポーネントの状態（オン、オフ、フォーカス等）に応じて異なる組み合わせに設定可能であるので、ユーザに対して、GUIコンポーネントの状態をより視覚的效果を高めて認識させることが可能となる。

【0109】

本発明では、コンポーネントの論理部分と、グラフィックス表現部分とを分離して構成し、論理部分と、グラフィックス表現部分とに基づいて、GUIコンポーネントを生成する構成とし、グラフィックス表現部分に、さらに、コンポーネント状態に応じたイメージファイルに基づくアニメーション実行機能を持たせたので、論理部分とは独立した処理により、アニメーションの設定態様を容易に変更することが可能となる。

10

【0110】

[7. GUI表示装置構成例]

次に、情報家電等、上述した説明に従った様々な表示処理を実行するグラフィカル・ユーザインタフェース（GUI）を有する装置構成例について、図11を参照して説明する。

【0111】

GUI搭載機器としては、様々な家電機器、通信機器が想定される。図11に示す装置構成例は、その1つの例を示し、動画像をネットワークを介して受信、あるいは記憶媒体から読み出して再生可能な構成を持つGUI表示装置構成例を示している。

【0112】

図11に示すGUI表示装置800には、インターネット配信データ、あるいはブロードキャストデータ等を受信する通信インタフェース801を有し、データ配信サイトからHTML文書、動画コンテンツなどを受信する。また、記憶媒体インタフェース802は、DVD、フラッシュメモリ等、各種記憶媒体820との間のデータ転送用インタフェースであり、画像データなど、各種コンテンツを記憶媒体820とデータ処理装置800間で入出力するインタフェースである。

20

【0113】

画像配信サイトから送信されるコンテンツ、あるいは記憶媒体820に格納されるコンテンツが符号化されている場合には、デコード/エンコード処理部803においてデコード処理が実行された後、表示制御部808の制御のもとに、表示装置810において表示する。

30

【0114】

一方、ユーザによる入力データ、各種コマンドは、コントローラ811から入力され、受信部807が受信して、制御部（CPU）804に出力する。制御部804は、前述した各コンポーネントに対応する論理部品、グラフィックス表現部品等に従って、各種の処理、例えばコンポーネントの状態に応じたイメージデータを取得し、取得したイメージデータを出力することでアニメーション表示処理を実行する。

【0115】

GUI表示装置800における記憶部は、受信データの記憶、各種処理プログラムの記憶、プリセット表示画像プログラムの記憶、あるいはCPU804におけるワークエリアとしての記憶領域等に用いられる記憶部であり、RAM、ROM等によって構成される。タイマ806は、GUI処理における各種プログラムの実行時間の計測に使用される。

40

【0116】

以上、特定の実施例を参照しながら、本発明について詳解してきた。しかしながら、本発明の要旨を逸脱しない範囲で当業者が該実施例の修正や代用を成し得ることは自明である。すなわち、例示という形態で本発明を開示してきたのであり、限定的に解釈されるべきではない。本発明の要旨を判断するためには、冒頭に記載した特許請求の範囲の欄を参酌すべきである。

【0117】

なお、明細書中において説明した一連の処理はハードウェア、またはソフトウェア、あるいは両者の複合構成によって実行することが可能である。ソフトウェアによる処理を実行

50

する場合は、処理シーケンスを記録したプログラムを、専用のハードウェアに組み込まれたコンピュータ内のメモリにインストールして実行させるか、あるいは、各種処理が実行可能な汎用コンピュータにプログラムをインストールして実行させることが可能である。

【0118】

例えば、プログラムは記録媒体としてのハードディスクやROM (Read Only Memory) に予め記録しておくことができる。あるいは、プログラムはフレキシブルディスク、CD-ROM (Compact Disc Read Only Memory)、MO (Magneto optical) ディスク、DVD (Digital Versatile Disc)、磁気ディスク、半導体メモリなどのリムーバブル記録媒体に、一時的あるいは永続的に格納（記録）しておくことができる。このようなリムーバブル記録媒体は、いわゆるパッケージソフトウェアとして提供することができる。

10

【0119】

なお、プログラムは、上述したようなリムーバブル記録媒体からコンピュータにインストールする他、ダウンロードサイトから、コンピュータに無線転送したり、LAN (Local Area Network)、インターネットといったネットワークを介して、コンピュータに有線で転送し、コンピュータでは、そのようにして転送されてくるプログラムを受信し、内蔵するハードディスク等の記録媒体にインストールすることができる。

【0120】

なお、明細書に記載された各種の処理は、記載に従って時系列に実行されるのみならず、処理を実行する装置の処理能力あるいは必要に応じて並列的にあるいは個別に実行されてもよい。また、本明細書においてシステムとは、複数の装置の論理的集合構成であり、各構成の装置が同一筐体内にあるものには限らない。

20

【0121】

【発明の効果】

以上説明してきたように、本発明によれば、グラフィカル・ユーザインタフェース (GUI) に設定される各コンポーネントのアニメーションを自在に設定することが可能となり、GUI搭載機器に応じた最適なコンポーネント・アニメーションの設定を容易に行なうことが可能となる。すなわち、コンポーネントに含まれるグラフィックス表現部品のグラフィックス表現クラスにより実行されるアニメーションインデックス・メソッドに従って、イメージファイルから、予め設定したイメージデータを読み込んで表示する構成としたので、コンポーネントの状態に応じて設定されたイメージによってアニメーションが実行可能となり、GUIコンポーネントの状態に応じた視覚的効果の高いアニメーション表現が可能となる。

30

【0122】

さらに、本発明では、コンポーネントの論理部分と、グラフィックス表現部分とを分離して構成し、論理部分と、グラフィックス表現部分とに基づいて、GUIコンポーネントを生成する構成とし、グラフィックス表現部分に、さらに、コンポーネント状態に応じたイメージファイルに基づくアニメーション実行機能を持たせたので、論理部分とは独立した処理により、アニメーションの設定態様を容易に変更することが可能となる。

【図面の簡単な説明】

【図1】本発明のGUIアプリケーション開発支援装置のハードウェア構成例を示す図である。

40

【図2】本発明のGUIアプリケーション開発支援装置の機能構成を示すブロック図である。

【図3】本発明のGUIアプリケーション開発支援装置の処理シーケンスを説明するシーケンス図である。

【図4】本発明のGUIアプリケーション開発支援装置の処理シーケンスを説明するシーケンス図である。

【図5】本発明のGUIアプリケーション開発支援装置の処理を実行するために必要なファイル構成例を説明する図である。

【図6】本発明のGUIアプリケーション開発支援装置において生成したGUIコンポー

50

メントの表示例を説明する図である。

【図 7】本発明の GUI アプリケーション開発支援装置において生成した GUI コンポーネントの表示例を説明する図である。

【図 8】本発明の GUI アプリケーション開発支援装置において実行されるコンポーネントの各状態に対応したアニメーション設定処理例を説明する図である。

【図 9】本発明の GUI アプリケーション開発支援装置においてコンポーネントのアニメーション処理を設定するための機能構成を示すブロック図である。

【図 10】本発明の GUI アプリケーション開発支援装置において生成したコンポーネントの状態に対応するアニメーション実行処理例を説明する図である。

【図 11】GUI 表示装置の構成例を示すブロック図である。

10

【符号の説明】

101 CPU

102 メモリ

103 ディスプレイコントローラ

104 入力機器インターフェース

105 ネットワークインターフェース

106 外部機器インターフェース

107 バス

108 ディスプレイ

109 キーボード

20

110 マウス

111 HDD

112 メディアドライブ

150 グラフィックス処理チップ

160 過搬型メディア

200 部品記憶部

201 論理部品記憶部

202 グラフィックス表現部品記憶部 (シンプル)

203 グラフィックス表現部品記憶部 (リッチ)

205 イメージデータ記憶部

30

210 編集処理部

211 プロパティ編集部

212 メソッド編集部

213 イベント編集部

220 データ入出力部

221 プログラム記憶部

222 プログラム実行部

223 表示部

311 ボタン

312 コンボボックス

40

320 ボタン

330 コンボボックス

321, 322 アニメーション

331, 332 アニメーション

351 コンポーネント指定欄

352 イメージ設定欄

400 部品記憶部

401 論理部品記憶部

402 グラフィックス表現部品記憶部

403 イメージデータ記憶部

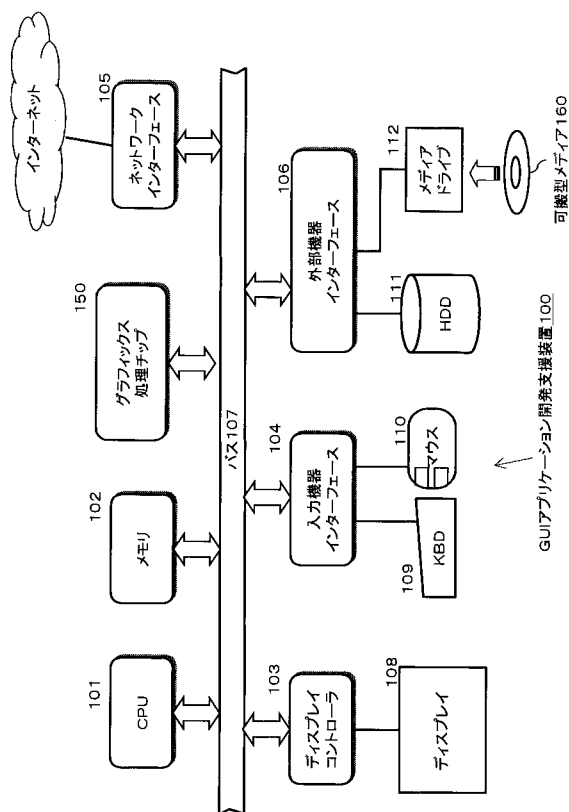
50

- 4 1 0 編集処理部
- 4 2 0 データ入出力部
- 4 2 1 プログラム記憶部
- 4 2 2 プログラム実行部
- 4 2 3 表示部
- 5 1 1 グラフィックス表現クラス
- 5 1 2 イメージファイル
- 5 1 3 ディスプレイ
- 5 1 4 アニメーション
- 8 0 0 G U I 表示装置
- 8 0 1 通信 I / F
- 8 0 2 記憶媒体 I / F
- 8 0 3 デコード / エンコード処理部
- 8 0 4 制御部 (C P U)
- 8 0 5 記憶部
- 8 0 6 タイマ
- 8 0 7 受信部
- 8 0 8 表示制御部
- 8 1 0 表示部
- 8 1 1 コントローラ

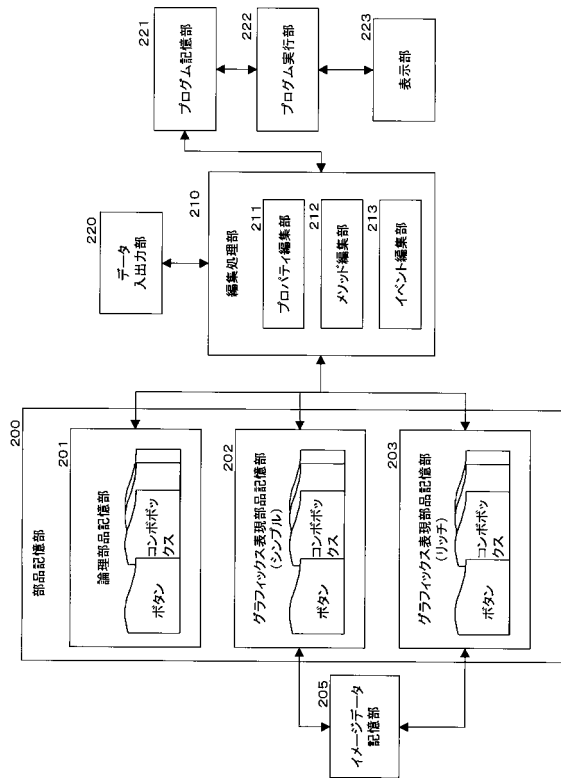
10

20

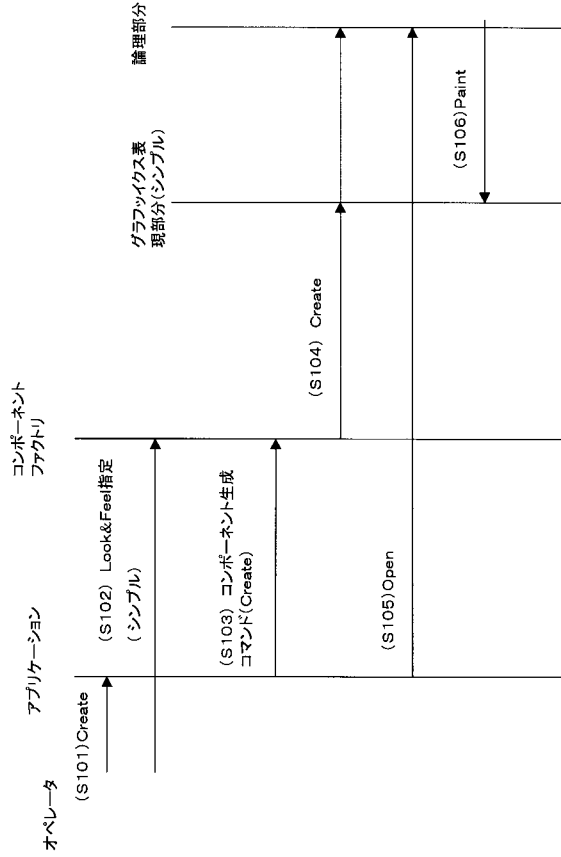
【図 1】



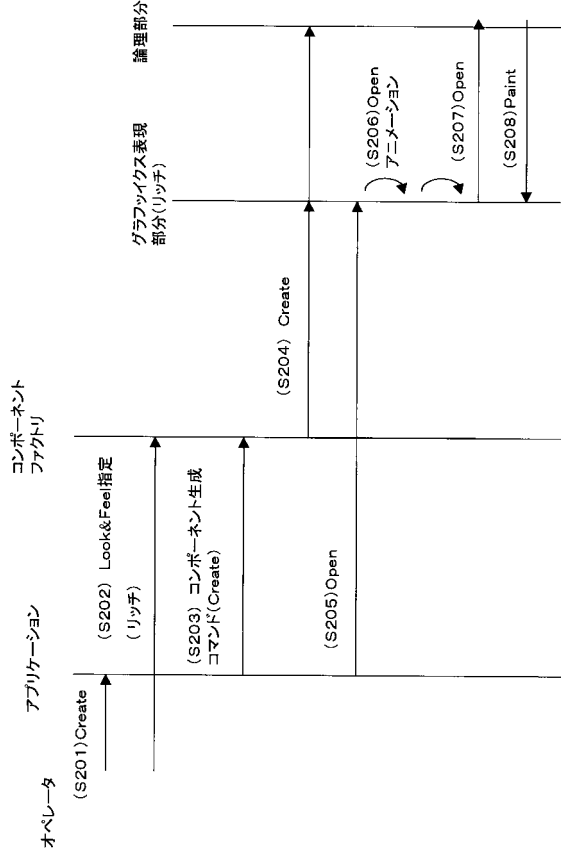
【図 2】



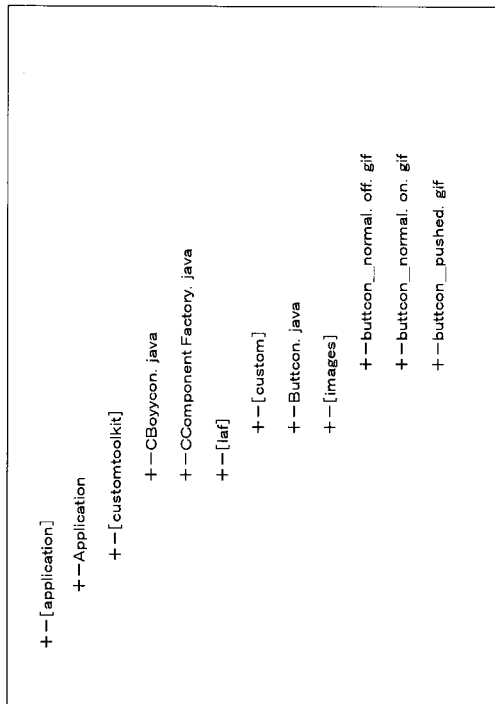
【図 3】



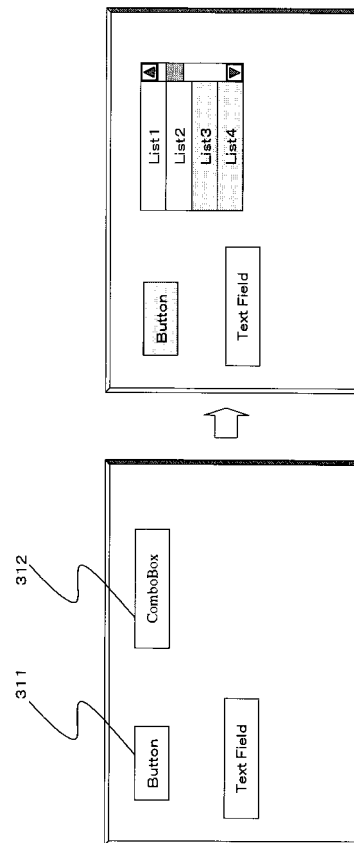
【図 4】



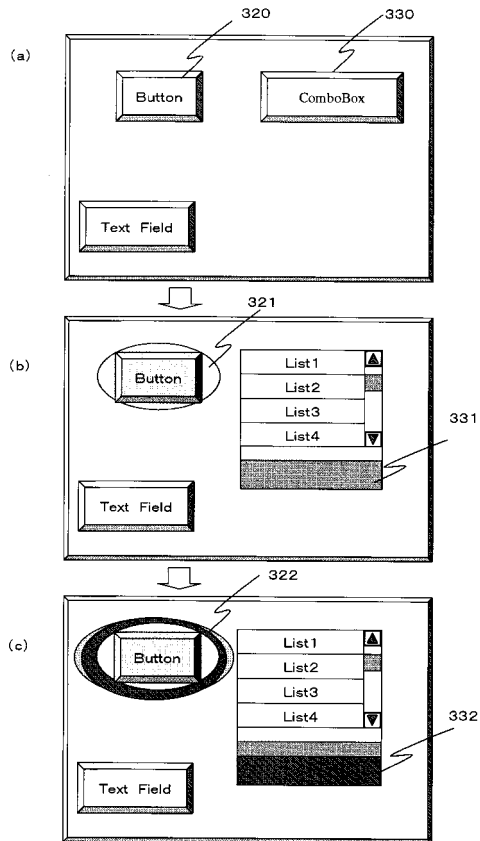
【図 5】



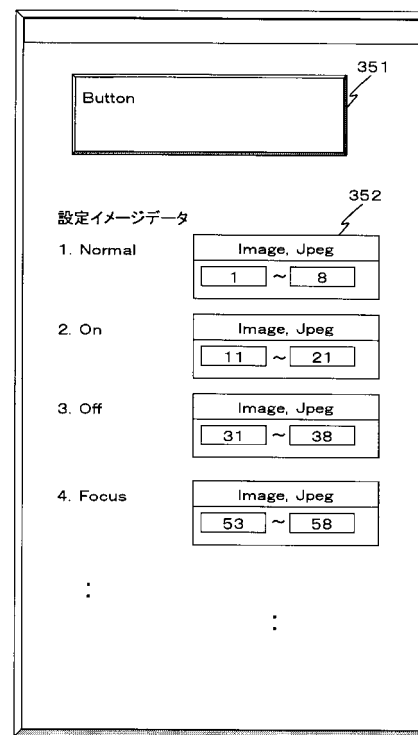
【図 6】



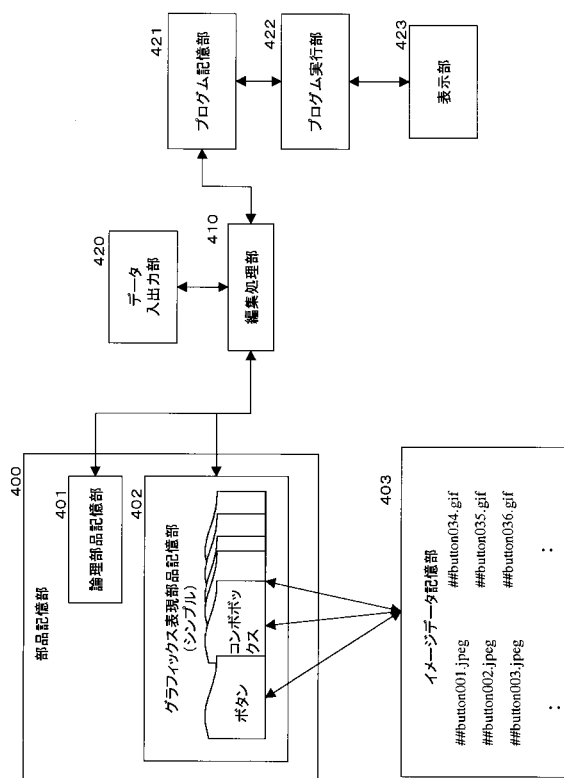
【図 7】



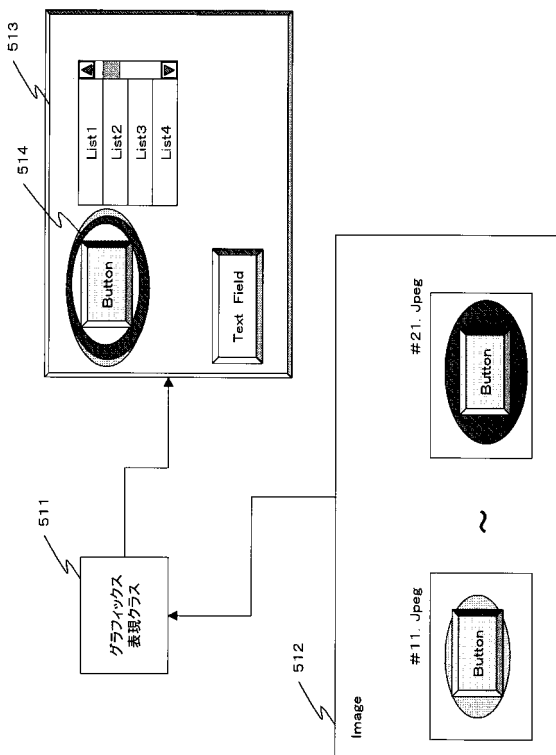
【図 8】



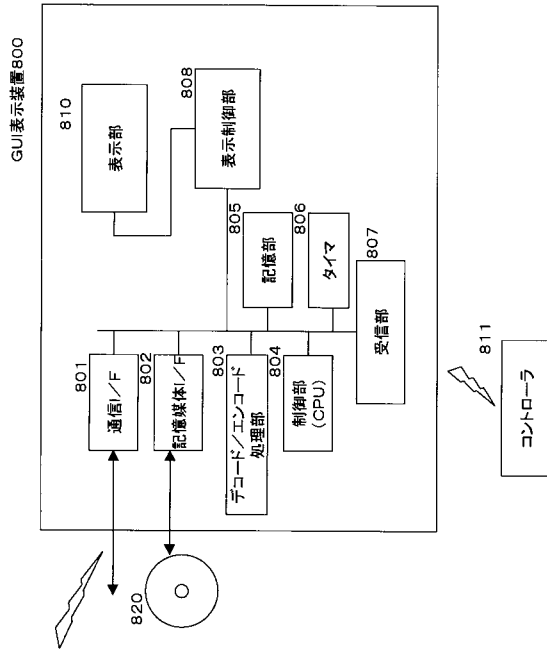
【図 9】



【図 10】



【図 11】



フロントページの続き

(72)発明者 岡本 直也
東京都品川区北品川6丁目7番35号 ソニー株式会社内

審査官 中野 裕二

(56)参考文献 特開平11-73259(JP,A)
特開2000-276339(JP,A)
特開2002-244848(JP,A)
特開2004-102497(JP,A)
特開2004-102499(JP,A)
特開2004-102500(JP,A)
米国特許第6222537(US,B1)

(58)調査した分野(Int.Cl., DB名)
G06F 9/44
G06F 9/06
G06F 3/00
G06F 3/14
JSTPLUS、JST7580ファイル(JOIS)