

(19) 日本国特許庁(JP)

(12) 特 許 公 報(B2)

(11) 特許番号

特許第3755500号  
(P3755500)

(45) 発行日 平成18年3月15日(2006.3.15)

(24) 登録日 平成18年1月6日(2006.1.6)

(51) Int. Cl.

F I

G 0 6 F 9/44 (2006.01)

G 0 6 F 9/06 6 2 0 C

G 0 6 F 3/048 (2006.01)

G 0 6 F 3/00 6 5 1 E

請求項の数 5 (全 17 頁)

(21) 出願番号	特願2002-261363 (P2002-261363)	(73) 特許権者	000002185
(22) 出願日	平成14年9月6日(2002.9.6)		ソニー株式会社
(65) 公開番号	特開2004-102500 (P2004-102500A)		東京都品川区北品川6丁目7番35号
(43) 公開日	平成16年4月2日(2004.4.2)	(74) 代理人	100093241
審査請求日	平成16年4月21日(2004.4.21)		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、

データ入力部と、

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と、

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、

少なくとも 1 以上のグラフィックス表現部品記憶部の格納するグラフィックス表現部品にはペイント処理実行プログラムが格納されており、

さらに、少なくとも 1 以上のグラフィックス表現部品記憶部の格納するグラフィックス表現部品には、アニメーション実行プログラムが格納されており、

前記編集処理部は、前記データ入力部に入力された入力情報に従って、前記複数のグラ

10

20

フィックス表現部品記憶部から1つのグラフィックス表現部品記憶部を選択し、選択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抽出する処理を実行する構成であることを特徴とするGUIアプリケーション開発支援装置。

【請求項2】

前記編集処理部は、GUIコンポーネントの生成において、各コンポーネントに対応するプロパティ、メソッド、および、イベントの編集処理を実行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のデータ入力部において入力データを受領するステップと、

前記GUIアプリケーション開発支援装置の編集処理部における実行ステップであり、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選択抽出する論理部品選択ステップと、

前記編集処理部における実行ステップであり、前記入力データ受領ステップにおいて受領した情報に基づいて、少なくともペイント処理実行プログラムおよびアニメーション実行プログラムを含むコンポーネントのグラフィックス表現に関する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、1つのグラフィックス表現部品記憶部を選択し、選択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の抽出処理を実行するグラフィックス表現部品選択ステップと

20

、前記編集処理部における実行ステップであり、前記入力データ受領ステップにおいて受領した情報に基づいて、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と、

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】

前記編集処理部における実行ステップである前記編集処理ステップは、GUIコンポーネントの生成において、各コンポーネントに対応するプロパティ、メソッド、および、イベントの編集処理を実行する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アプリケーションをGUIアプリケーション開発支援装置において生成させるGUIアプリケーションプログラム生成処理を実行するコンピュータ・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のデータ入力部において、入力データを受領するステップと、

40

前記GUIアプリケーション開発支援装置の編集処理部において、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選択抽出する論理部品選択ステップと、

前記編集処理部において、前記入力データ受領ステップにおいて受領した情報に基づいて、少なくともペイント処理実行プログラムおよびアニメーション実行プログラムを含むコンポーネントのグラフィックス表現に関する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、1つのグラフィックス表現部品記憶部を選択し、選択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の抽出処理を実行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、前記入力データ受領ステップにおいて受領した情報に基づい

50

て、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】

【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、論理部分を様々な機器において共通に使用することを可能とし、一方、グラフィックス表現部分を各機器に応じて選択可能とすることにより、様々な機器に適応したユーザインタフェースを容易に構築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および方法、並びにコンピュータ・プログラムに関する。

10

【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。

20

【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。

30

【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。

40

【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

50

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。

【０００８】

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。

【０００９】

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る。例えば全体として柔らかい穏やかなイメージとしたり、あるいは余分なグラフィックスを排除し、機能重視型のイメージとする等である。このように異なるイメージを持つＧＵＩを作成する場合、上述の論理部分と、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利用する構成では、各イメージに応じた別コンポーネントを生成することが必要となり、コンポーネント作成処理の負荷が増大し、またコンポーネント数の増加にもつながることとなる。

【００１０】

【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、論理部分を様々な機器において共通に使用することを可能とし、一方、グラフィックス表現部分を各機器に応じて選択可能とすることにより、様々な機器に適応したユーザインタフェースを容易に構築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。

【００１１】

【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、

データ入力部と、

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と、

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、

少なくとも１以上のグラフィックス表現部品記憶部の格納するグラフィックス表現部品にはペイント処理実行プログラムが格納されており、

さらに、少なくとも１以上のグラフィックス表現部品記憶部の格納するグラフィックス表現部品には、アニメーション実行プログラムが格納されており、

10

20

30

40

50

前記編集処理部は、前記データ入力部に入力された入力情報に従って、前記複数のグラフィックス表現部品記憶部から1つのグラフィックス表現部品記憶部を選択し、選択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抽出する処理を実行する構成であ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。

【0012】

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、前記編集処理部は、GUIコンポーネントの生成において、各コンポーネントに対応するプロパティ、メソッド、および、イベントの編集処理を実行する構成であることを特徴とする。

【0013】

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（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、

前記GUIアプリケーション開発支援装置のデータ入力部において入力データを受領するステップと、

前記GUIアプリケーション開発支援装置の編集処理部における実行ステップであり、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選択抽出する論理部品選択ステップと、

前記編集処理部における実行ステップであり、前記入力データ受領ステップにおいて受領した情報に基づいて、少なくともペイント処理実行プログラムおよびアニメーション実行プログラムを含むコンポーネントのグラフィックス表現に関する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、1つのグラフィックス表現部品記憶部を選択し、選択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の抽出処理を実行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ける実行ステップであり、前記入力データ受領ステップにおいて受領した情報に基づいて、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と、

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にある。

【0014】

さらに、本発明のGUI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、前記編集処理部における実行ステップである前記編集処理ステップは、GUIコンポーネントの生成において、各コンポーネントに対応するプロパティ、メソッド、および、イベントの編集処理を実行することを特徴とする。

【0015】

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（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させるGUIアプリケーションプログラム生成処理を実行するコンピュータ・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のデータ入力部において、入力データを受領するステップと、

前記GUIアプリケーション開発支援装置の編集処理部において、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選択抽出する論理部品選択ステップと、

前記編集処理部において、前記入力データ受領ステップにおいて受領した情報に基づいて、少なくともペイント処理実行プログラムおよびアニメーション実行プログラムを含むコンポーネントのグラフィックス表現に関する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、1つのグラフィックス

10

20

30

40

50

表現部品記憶部を選択し、選択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の抽出処理を実行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、前記入力データ受領ステップにおいて受領した情報に基づいて、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。

【0019】

【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（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、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たので、論理部分を様々な機器において共通に使用することを可能とし、一方、様々な表現態様を設定した異なるグラフィックス表現部分を各機器に応じて選択可能とすることにより、様々な機器に適応したユーザインタフェースを容易に構築することが可能となる。

10

【0020】

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。

20

【0021】

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。

【0022】

【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。

30

【0023】

[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。CPU101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、CPU101は、バス107によって他構成部と相互接続されている。

40

【0024】

メモリ102は、CPU101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。

【0025】

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。

【0026】

50

表示制御部としてのディスプレイ・コントローラ103は、CPU101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、CPU101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。

【0027】

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。

10

【0028】

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。

【0029】

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。

20

【0030】

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。

【0031】

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。

30

【0032】

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。

【0033】

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。

40

【0034】

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。

【0035】

[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

50

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。

【0036】

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。

【0037】

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。

【0038】

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。

【0039】

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。

【0040】

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、CPU101の制御の下に実行される処理である。

【0041】

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。

【0042】

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。

【0043】

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。

【0044】

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

10

20

30

40

50



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。

【0045】

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

【0046】

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。

【0047】

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

【0048】

[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。

【0049】

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

【0050】

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

【0051】

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。

【0052】

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。

50

## 【 0 0 5 3 】

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。

## 【 0 0 5 4 】

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

## 【 0 0 5 5 】

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。

## 【 0 0 5 6 】

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

## 【 0 0 5 7 】

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。

30

## 【 0 0 5 8 】

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る。

## 【 0 0 5 9 】

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。

## 【 0 0 6 0 】

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

## 【 0 0 6 1 】

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。

50

## 【 0 0 6 2 】

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 6 3 】

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。

10

## 【 0 0 6 4 】

## [ 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 6 5 】

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 6 6 】

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 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。

20

なお、クラスは、J a v a の実行単位プログラムを示す。

30

## 【 0 0 6 7 】

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 l ファイル [ l a f ] ファイルが含まれる。

## 【 0 0 6 8 】

L o o k & F 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。

40

## 【 0 0 6 9 】

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。

50

## 【 0 0 7 0 】

## [ 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 1 】

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 7 2 】

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 7 3 】

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 7 4 】

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 7 5 】

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 7 6 】

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 7 7 】

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 7 8 】

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（G U I）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、論理部分と、グラフィックス表現部分とに基づいて、G U I コンポーネントを生成する構成としたので、論理部分を様々な機器において共通に使用することを可能とし、一方、

10

20

30

40

50

様々な表現態様を設定した異なるグラフィックス表現部分を各機器に応じて選択可能とすることにより、様々な機器に適応したユーザインタフェースを容易に構築することが可能となる。

#### 【0079】

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照すべきである。

#### 【0080】

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。

#### 【0081】

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。

#### 【0082】

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。

#### 【0083】

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。

#### 【0084】

#### 【発明の効果】

以上説明してきたように、本発明のアプリケーション開発支援装置および方法によれば、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たので、論理部分を様々な機器において共通に使用することを可能とし、一方、様々な表現態様を設定した異なるグラフィックス表現部分を各機器に応じて選択可能とすることにより、様々な機器に適応したユーザインタフェースを容易に構築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。

【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

10

20

30

40

50

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。

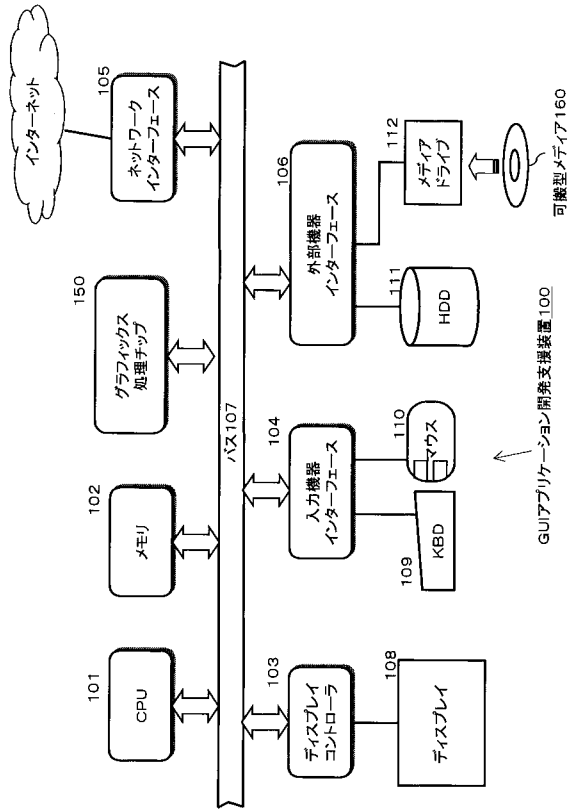
【図6】本発明のGUIアプリケーション開発支援装置において生成したGUIコンポーネントの表示例を説明する図である。

【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。

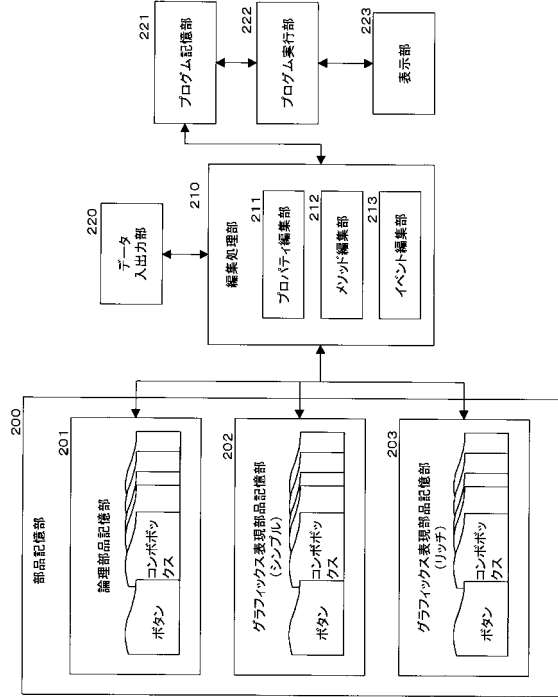
【符号の説明】

101	CPU	
102	メモリ	10
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	
110	マウス	
111	HDD	
112	メディアドライブ	20
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）	
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	30
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	
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	40

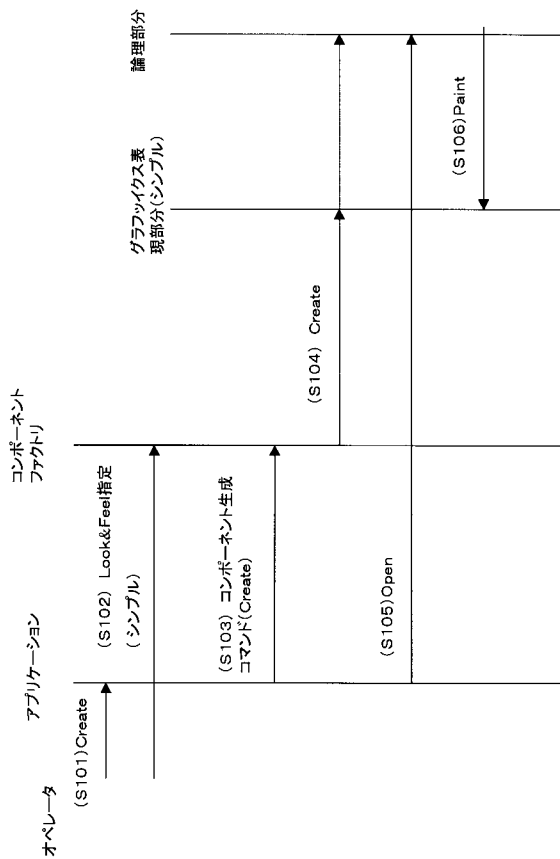
【図 1】



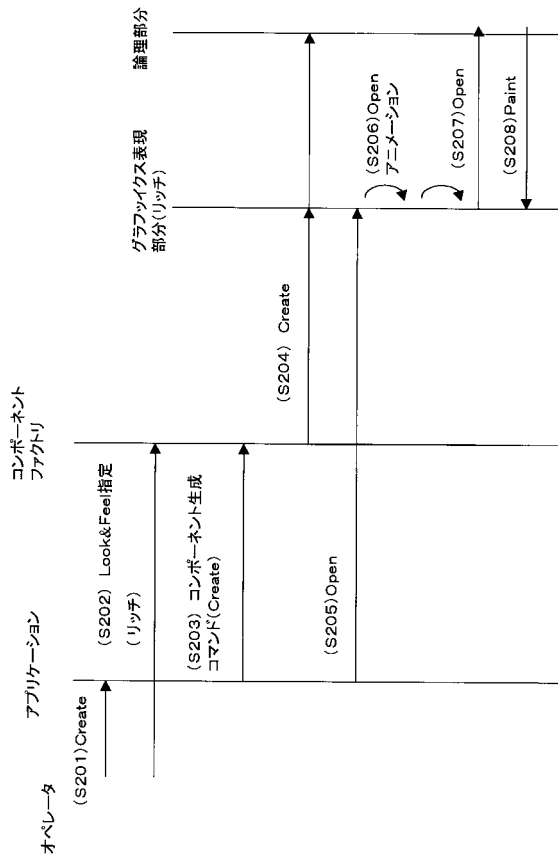
【図 2】



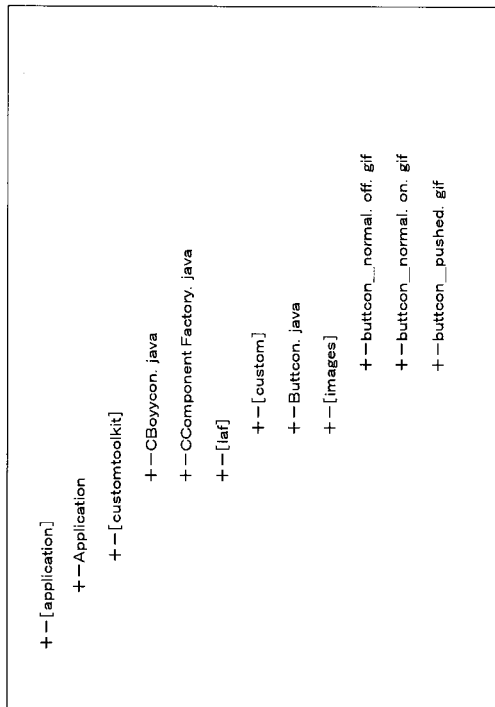
【図 3】



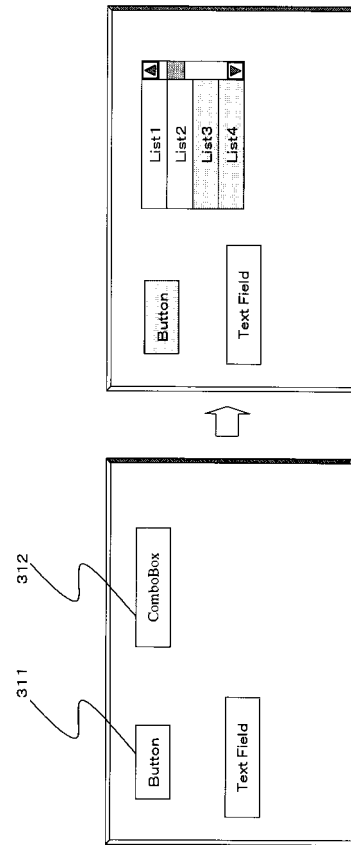
【図 4】



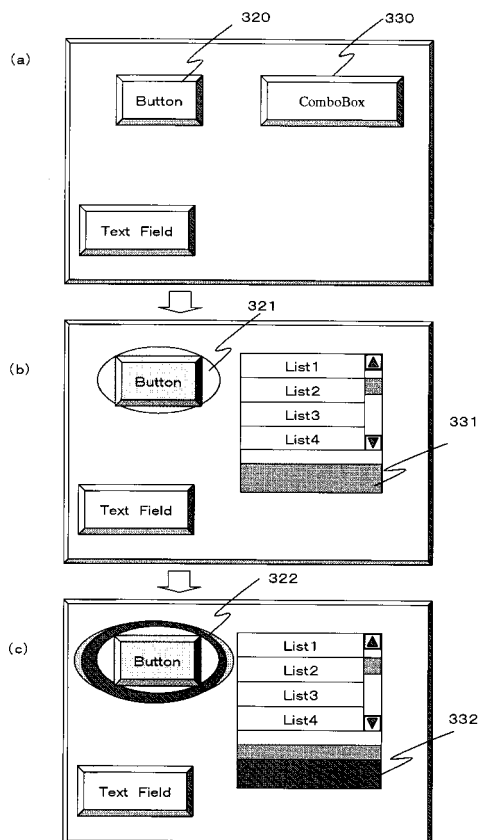
【 図 5 】



【 図 6 】



【 図 7 】





---

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2004-102497(JP,A)

特開2004-102498(JP,A)

特開2004-102499(JP,A)

登内敏夫(外1名), 組み合わせオブジェクトに基づくGUIライブラリ, コンピュータソフトウェア, 日本ソフトウェア科学会, 1995年 5月15日, 第12巻第3号, p.49~58

Brad A. Myers(外7名), Garnet: 対話性にすぐれ, 柔軟なGUIの作成を支援するツール, 日経エレクトロニクス, 日経BP社, 1991年 3月18日, 第522号, p.187~203

(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)