

(19) 日本国特許庁(JP)

## (12) 特許公報(B2)

(11) 特許番号

特許第3964988号  
(P3964988)

(45) 発行日 平成19年8月22日(2007.8.22)

(24) 登録日 平成19年6月1日(2007.6.1)

(51) Int.C1.

F 1

GO 6 F 3/048 (2006.01)

GO 6 F 3/048 653 A

請求項の数 4 (全 16 頁)

(21) 出願番号 特願平10-209367  
 (22) 出願日 平成10年7月24日(1998.7.24)  
 (65) 公開番号 特開平11-110098  
 (43) 公開日 平成11年4月23日(1999.4.23)  
 審査請求日 平成11年7月12日(1999.7.12)  
 審査番号 不服2004-7620(P2004-7620/J1)  
 審査請求日 平成16年4月15日(2004.4.15)  
 (31) 優先権主張番号 9718582.1  
 (32) 優先日 平成9年9月3日(1997.9.3)  
 (33) 優先権主張国 英国(GB)

(73) 特許権者 390009531  
 インターナショナル・ビジネス・マシーンズ・コーポレーション  
 INTERNATIONAL BUSINESS MACHINES CORPORATION  
 アメリカ合衆国10504 ニューヨーク  
 州 アーモンク ニューオーチャード  
 ロード  
 (74) 代理人 100086243  
 弁理士 坂口 博  
 (74) 代理人 100091568  
 弁理士 市位 嘉宏

最終頁に続く

(54) 【発明の名称】ヘルプ情報の表示方法および記録媒体

## (57) 【特許請求の範囲】

## 【請求項 1】

U I エンティティとのユーザ対話に応答して、前記 U I エンティティに関するコンピュータ・システム・ユーザ・インターフェースによるヘルプ情報を表示する方法であって、それぞれが複数の事前定義表示スタイルのそれぞれ異なる1つに応じてヘルプ情報を表示するための複数の選択可能表示コンポーネントを含み、表示情報と関連のインターフェース領域を定義するデータを返すための要求をインターフェースに追加される U I エンティティに出すように構成されている表示マネージャを使用してコンピュータによって実現され、

ユーザが、コンテナとして機能するアプリケーションによって U I コンポーネントを前記インターフェースに挿入することにより、前記 U I エンティティを前記インターフェースに追加したことに応答して、前記表示情報と前記インターフェース領域を定義するデータとをカプセル化する U I エンティティ定義コンポーネントのインスタンスを前記表示マネージャを使用して作成するステップと、

前記表示スタイルの1つに応じて前記ヘルプ情報を表示するために、複数の前記選択可能表示コンポーネントのリストを表示し、ユーザに選択させることにより、作成した前記 U I エンティティ定義コンポーネントに関連付ける前記表示コンポーネントを選択するステップと、

ユーザ入力装置ポインタが移動して前記インターフェース領域との事前定義接近関係になったときに、ユーザ対話に対する応答として前記ヘルプ情報を表示するために選択済み前記表示コンポーネントを始動するステップと、

10

20

前記 U I エンティティ定義コンポーネントのインスタンスのそれぞれが、一度に表示される前記ヘルプ情報の組を識別するためのグループ I D と事前設定される優先順位値とを有し、複数の前記データにより定義される複数の定義済みインタフェース領域が重なる場合、同じ前記グループ I D を有する複数の前記 U I エンティティ定義コンポーネントのインスタンスを識別し、識別した前記インスタンスの前記優先順位値を比較し、前記優先順位値が最も高いインスタンスに関するヘルプ情報を表示するステップとを含む、方法。

【請求項 2】

前記始動するステップは、前記インタフェース領域との前記ユーザ対話に対する応答として前記 U I エンティティ定義コンポーネントが通知を受け取り、前記通知の受け取りに応答して、前記選択済み表示コンポーネントに通知を送ることを含む、請求項 1 に記載の方法。

10

【請求項 3】

U I エンティティとのユーザ対話に応答して、コンピュータ・システム・ユーザ・インタフェースによる前記 U I エンティティに関するヘルプ情報の表示をサポートするためのコンピュータ可読なプログラムが記録された記録媒体であって、該プログラムは、それぞれが複数の事前定義表示スタイルのそれぞれ異なる 1 つに応じてヘルプ情報を表示するための複数の選択可能表示コンポーネントを含み、表示情報と関連のインタフェース領域を定義するデータを返すための要求をインタフェースに追加される U I エンティティに出すように構成されている表示マネージャを含み、

20

ユーザが、コンテナとして機能するアプリケーションによって U I コンポーネントを前記インタフェースに挿入することにより、前記 U I エンティティを前記インタフェースに追加したことに応答して、前記表示情報と前記インタフェース領域を定義するデータとをカプセル化する U I エンティティ定義コンポーネントのインスタンスを作成するステップと、

前記表示スタイルの 1 つに応じて前記ヘルプ情報を表示するために、複数の前記選択可能表示コンポーネントのリストを表示し、ユーザに選択されることにより、作成した前記 U I エンティティ定義コンポーネントに関連付ける前記表示コンポーネントを選択するステップと、

ユーザ入力装置ポインタが移動して前記インタフェース領域との事前定義接近関係になったときに、ユーザ対話に対する応答として前記ヘルプ情報を表示するために選択済み前記表示コンポーネントを始動するステップと、

30

前記 U I エンティティ定義コンポーネントのインスタンスのそれぞれが、一度に表示される前記ヘルプ情報の組を識別するためのグループ I D と事前設定される優先順位値とを有し、複数の前記データにより定義される複数の定義済みインタフェース領域が重なる場合、同じ前記グループ I D を有する複数の前記 U I エンティティ定義コンポーネントのインスタンスを識別し、識別した前記インスタンスの前記優先順位値を比較し、前記優先順位値が最も高いインスタンスに関するヘルプ情報を表示するステップとを前記コンピュータに実行させる、

記録媒体。

40

【請求項 4】

前記始動するステップは、前記インタフェース領域との前記ユーザ対話に対する応答として前記 U I エンティティ定義コンポーネントが通知を受け取り、前記通知の受け取りに応答して、前記選択済み表示コンポーネントに通知を送ることを含む、請求項 3 に記載の記録媒体。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】

本発明は、入力装置によるユーザ対話に対する応答としてコンピュータ・システム・ユーザ・インタフェースによる情報の表示に関する。

50

## 【0002】

## 【従来の技術】

コンピュータ・システム・ユーザ・インターフェース(ＵＩ)の一般的な要件は、ユーザ入力装置制御のポインタ(たとえば、マウス・ポインタ)がインターフェースの一部の上を通過するか、インターフェースの一部に接近するか、またはインターフェースの一部の上で停滞するときに、何らかの情報をユーザに表示する機能である。このような情報は、インターフェースのその部分ならびにユーザがそれと対話した結果に関する追加説明を提供することができ、それが使用する画面領域を低減するためにその情報の一部をインターフェース自体が省略する場合に特に有用になりうる。

## 【0003】

10

たとえば、多くのＵＩでは、それが実行するタスクを表す小さいアイコン・グラフィックを表示するプッシュ・ボタンを取り入れている。このようなボタンはときには「ツール・バー」と呼ばれる行または列に集めることができ、それらが占有する画面領域は、各プッシュ・ボタンにその機能の完全なテキスト説明を付ける場合に必要な領域よりかなり小さくなる。しかし、小さいアイコン・グラフィックは、特にインターフェースの新しいユーザにいつでも明確に理解されるわけではなく、マウスがそのボタンの上に停滞している場合に示される各ボタンのテキスト説明はインターフェースの使いやすさを大幅に高めることができる。

## 【0004】

20

情報は、多種多様な方法で表示することができる。例としては、テキストを含むポップアップ・ボックスまたはパブルがあり、それにより適当なグラフィックを備えた既存のインターフェースの上に重なってそれを周囲から線引きし、おそらくそれが関連するインターフェースの一部を補強する。あるいは、ステータス・バーまたはメッセージ・ウィンドウにテキストを示すことができ、または適当なオーディオ・インターフェースによりテキストを朗読することも可能である。ときには、情報が単純なテキスト表示の形式を取らない場合もあるが、アウトライン・カラーの変更などのインターフェースに加えられた他の変更や、グラフィックのアニメーション、音響効果の開始など、当然のことながら、他にも多くの可能性がある。

## 【0005】

30

適度な複雑さのインターフェースでも、このような情報を適用できる部分は多種多様に存在する可能性があり、このような個々の情報の集合を管理することは、当業者にとっても相当な難題を提示する可能性がある。この問題は、動的に変化するインターフェースの場合、特にインターフェース設計者が直接制御できないプログラム・コードによってインターフェースの各部分が追加され管理される場合に、増大する。たとえば、所与のアプリケーションが、様々なコンポーネント(アプレットまたはＵＩコントロール)を挿入できるコンテナとして機能し、おそらく異なるコンポーネントが異なる人によって異なる時点に作成されることは、当技術分野では既知のことである。システム・エンド・ユーザまたはインターフェース設計者によってこのようなコンポーネントを挿入することによるインターフェースへの追加は、容易にコンテナの開発者の特定の制御下に置くことができない。

## 【0006】

40

Java(サン・マイクロシステムズ社の商標)などのプラットフォーム非依存コンポーネント・プログラミング環境では、個々のコンポーネント自体がこの表示のメソッドを含まない限り、アプリケーション・コンポーネント用のヘルプ情報は一般にまったく表示されない。というのは、ヘルプ情報を表示するために適用可能な汎用メカニズムが現在、まったくないからである。これは、特定のオペレーティング・システム用に作成されたアプリケーションが「ヘルプ」情報を表示するために標準のオペレーティング・システム機能への呼出しを行うようにしばしばコーディングされるような、あまり動的ではない環境との違いである。このような標準のオペレーティング・システム機能は、通常、インターフェースを動的に変更するためにヘルプ表示を処理するためのメカニズムを提供しない。

## 【0007】

50

それは特定のオペレーティング・システム機能への依存を回避するが、コンポーネントがそのヘルプ情報の表示を完全に担当しなければならぬので、コンポーネント開発の労力が大幅に増大する。特に、感知可能画面領域用の状態モデルを定義すること（ヘルプ情報の除去などが必要なときに、どのようなポインタ移動によってヘルプ情報の表示をトリガすべきかを決定すること）は複雑かつ時間を要する作業である。

#### 【0008】

また、エンド・ユーザの観点から見ると、各コンポーネントがヘルプ情報の表示を担当する場合、インターフェースは、異なるコンポーネントとの同様の対話のために一貫した情報表示を有する場合が多くなる（たとえば、異なるスタイルのヘルプ・バブルが表示される可能性がある）。これは、ユーザを非常に混乱させる可能性がある。さらに、オペレーティング・システムによって提供されるかまたはコンポーネントに組み込まれる情報表示機能は、ユーザが本当に必要とするタイプの情報表示効果を提供できないこともある（たとえば、ユーザがオーディオ・ヘルプ情報を必要としているのに、テキストベースの「ホバー・ヘルプ」だけがサポートされている場合）。

10

#### 【0009】

上記のように、情報の表示時にユーザが行う要求によって、さらに問題点が発生する。それを必要とするユーザに対して適切な時期に情報が表示されるが、それを必要としないユーザの活動を妨げないようにするには、単純なインプリメンテーションが達成するものより高度な挙動を用意しなければならない。これが特に当てはまるのは、ある時点では援助を必要としないユーザでも他の時点で、または同じ時点でもインターフェースの不慣れな部分を使用する場合には同じ援助を必要とする場合があるからである。ユーザが必要とし期待する挙動が用意されてない場合、ユーザはいろいろして挫折する可能性があり、インターフェースの使いやすさは増大するのではなく、むしろ低下する可能性がある。

20

#### 【0010】

##### 【発明が解決しようとする課題】

本発明の目的は、情報が関連付けられているコンピュータ・システム・ユーザ・インターフェース（UI）エンティティとのユーザ対話に対する応答としてUIによりヘルプ情報を表示するための汎用メカニズムを提供することである。

#### 【0011】

##### 【課題を解決するための手段】

30

この汎用表示メカニズムは、UIコンポーネントから、ヘルプ情報の表示を管理する作業の多くを除去し、複数の異なるスタイル（テキスト、オーディオ、グラフィックのアニメーションなど）での情報の表示をサポートする。本発明は、プラットフォーム非依存コンピュータ・プログラミング環境向けのヘルプ情報表示マネージャ・フレームワークで実現されることが好ましい。

#### 【0012】

このコンテキストでのUI「エンティティ」は、ユーザ・インターフェース内の1つの表現を有するかまたはそれに関連するインターフェースの1つの領域を有するソフトウェア・コンポーネント（以下、UIコンポーネント）であり、UIコンポーネントに関連付けられたユーザ・インターフェースの領域である。したがって、「UIエンティティ」は、アイコンによって表されたアプリケーション・プログラムまたはアプレットなどのUIコンポーネント、ボタン、スクロールバー、入力フィールドなどのUIコントロール、ツールバーやメニュー・バーなどの項目リスト、ウィンドウ、図およびチャートの一部、イメージ・マップの領域を含む。

40

#### 【0013】

このコンテキストでの「ヘルプ」情報は、インターフェースの一部またはそれと対話した結果についてユーザに通知する、テキスト、イメージ、サウンド、その他の媒体の形式の情報である。このような情報はUIコンポーネントの本質的な内部機能ではないので、UIコンポーネントの状態モデルから個別に実現することができるかまたは異なる動作状態を有していないUIエンティティ（チャートの一部分など）に関連付けることができる。す

50

なわち、ヘルプ情報の表示は、UIコンポーネントの状態も変更する対話イベントによって始動できるが、ヘルプ情報を表示するアクションは、UIコンポーネントの状態遷移エンジンとは無関係にトリガすることができる。

#### 【0014】

本発明は、「第1のレベルのヘルプ」の表示に特に適している。第1のレベルのヘルプは、ユーザ・インターフェースの定義済み領域に関連するかまたはそこからアクセス可能なヘルプ情報の単一レベル(すなわち、非階層配置)の項目を含む。本発明は、先行技術の第1のレベルのヘルプ機能の能力を相當に上回るものである。

#### 【0015】

第1の態様では、本発明は、コンピュータ・システム・ユーザ・インターフェースによるUIエンティティに関するヘルプ情報の表示をサポートするための表示マネージャ・フレームワークを提供し、この表示マネージャ・フレームワークは、UIエンティティに関するインターフェース領域定義と関連のヘルプ情報をカプセル化するためのUIエンティティ定義コンポーネントと、それぞれの表示スタイルに応じて複数の表示コンポーネントによって実現するための1つまたは複数の汎用ヘルプ情報表示メソッドを有するセットを提供する表示インターフェースとを含む。

#### 【0016】

このフレームワークを実現する表示マネージャは、複数の選択可能表示コンポーネントを含み、それぞれが複数の事前定義表示スタイルのそれぞれ異なる1つに応じてヘルプ情報を表示するために表示インターフェースを実現することが好ましい。また、この表示マネージャは、前記UIエンティティの1つに関する情報を表示するために所望の表示スタイルに応じて前記表示コンポーネントの1つを選択する手段と、前記UIエンティティとのユーザ対話に対する応答として選択済み表示コンポーネントを始動する手段も含むことが好ましい。

#### 【0017】

第2の態様では、本発明は、コンピュータ・システムUIによる複数のUIエンティティに関するヘルプ情報の表示を処理するための表示マネージャを提供し、この表示マネージャは、それぞれが複数の事前定義表示スタイルのそれぞれ異なる1つに応じてヘルプ情報を表示するための複数の表示コンポーネントと、前記UIエンティティの1つに関連する情報を表示するために前記表示コンポーネントの1つを選択する手段と、前記UIエンティティとのユーザ対話に対する応答として前記選択済み表示コンポーネントを始動する手段とを含む。

#### 【0018】

本発明による表示マネージャは、以下に示す複数の「表示スタイル」に関する表示コンポーネントを提供することが好ましく、すなわち、表示スタイルは以下の通りである。

- ポップアップ・バブル内のテキスト表示(この場合の「情報」はテキストである)
- メッセージ行またはステータス行でのテキスト表示(「情報」はテキストである)
- オーディオ・クリップの再生(「情報」がオーディオの場合)
- グラフィックのアニメーション(「情報」がアニメーションによって達成される結果であり、表示メソッドによって操作されるデータが1組のイメージと各イメージ変化の間の時間などのアニメーション特性を含む場合)。

#### 【0019】

所定の範囲の各種ヘルプ・バブル・スタイル(明確な吹出しの外観、あまり明確ではない矩形の外観など)など、追加の表示スタイルもサポートすることができる。

#### 【0020】

第1レベルのヘルプを表示するための先行技術の解決策では、現在、1組の選択可能表示コンポーネントを使用して複数の異なる方法でUIエンティティ関連情報を表示するための汎用メカニズムを表示マネージャに提供していない。異なる表示コンポーネントを処理するための汎用メカニズムのこのようなフレキシビリティは、本発明によるヘルプ情報表示の重要な新規態様の1つである。

10

20

30

40

50

**【 0 0 2 1 】**

好みしい実施例による表示マネージャは、それぞれが情報表示のメソッドを実現する1つまたは複数の選択可能表示コンポーネントと、1つまたは複数のU Iエンティティ定義コンポーネントとを含む。U Iエンティティ定義コンポーネントのインスタンスは、U Iエンティティ用のインターフェース領域定義と、コンピュータ・システムU Iにより表示されるその領域に関連する情報をカプセル化する。この情報は、イメージ、カラー、サウンド、ヘルプ・テキストなどにすることができる。1つまたは複数のU Iエンティティ定義コンポーネントは、U Iの特定の領域へのポインティング・デバイスの移動や、一定の時間のその領域上でのポインタの停滞など、関連ユーザ対話の通知を受け取り、保持する情報を表示すべきかまたは除去すべきかを決定したときに表示コンポーネントに通知を出すように適合されている。

10

**【 0 0 2 2 】**

U Iエンティティに関するU Iエンティティ定義コンポーネントが作成されると、表示マネージャは、関連マウス移動情報を表示マネージャに通知すること以外に関連U Iコンポーネントからの介入なしに、そのU Iエンティティに関するヘルプ情報の表示を処理することができる。

**【 0 0 2 3 】**

各U Iエンティティ定義コンポーネントは1つまたは複数の表示コンポーネントに関連付けることができ、そのいずれかを選択して表示を処理することができる。選択済み表示コンポーネントは、情報を表示または除去するときに通知を受け取り、適切なアクションを行う。

20

**【 0 0 2 4 】**

本発明による表示コンポーネントの選択は、1組の可能な表示コンポーネントのいずれを使用すべきかというユーザ選択のサポートを含むことが好みしい。これにより、ユーザは、その優先ヘルプ表示スタイルを使用するためにユーザ・インターフェースをカスタマイズすることができ、そのヘルプ情報を表示するためにダウンロードしたU Iコンポーネントがそれ専用のカスタム作成メソッドを使用した場合に発生する不整合という問題を回避することができる。

**【 0 0 2 5 】**

第3の態様による本発明は、複数のU Iエンティティに関するコンピュータ・システム・ユーザ・インターフェースによるヘルプ情報表示の方法を提供し、この方法は、それぞれが複数の事前定義表示スタイルのそれぞれ異なる1つに応じてヘルプ情報を表示するための複数の選択可能表示コンポーネントを含む表示マネージャによって実現され、この方法は、U Iに追加されるU Iエンティティに対する応答として、ヘルプ情報と前記U Iエンティティによって提供されるU I領域定義とをカプセル化するU Iエンティティ定義コンポーネントのインスタンスを作成するステップと、前記表示スタイルの1つに応じて前記ヘルプ情報を表示するために前記表示マネージャの表示コンポーネントを選択するステップと、ユーザ入力装置ポインタが移動して前記U I領域との事前定義接近関係になったときにユーザ対話に対する応答として前記ヘルプ情報を表示するために前記選択済み表示コンポーネントを始動するステップとを含む。

30

**【 0 0 2 6 】**

選択済み表示コンポーネントは以下のように始動されることが好みしい。事前定義ポインタ接近関係が達成されたときに、そのU Iエンティティに関するU Iエンティティ定義コンポーネントに第1の通知が送られる。次に、U Iエンティティ定義コンポーネントは、選択済み表示コンポーネントを始動するために通知を送る。

40

**【 0 0 2 7 】**

第4の態様では、本発明は、コンピュータ・システム・ユーザU Iによる複数のU Iエンティティに関するヘルプ情報の表示をサポートするためのプラットフォーム非依存表示マネージャ・フレームワークを提供し、この表示マネージャ・フレームワークは、複数のU Iエンティティのそれぞれに関するU I領域定義と関連のヘルプ情報をカプセル化するた

50

めのU I エンティティ定義クラスと、ヘルプ情報を表示するためにU I エンティティ定義クラスのインスタンスによって呼び出されるように適合された1つまたは複数の表示クラス内で実現するための1つまたは複数の汎用表示メソッドを提供する表示インターフェースとを含む。

#### 【0028】

ヘルプ情報を表示するための先行技術の解決策では、Java プログラミング言語で作成されたU I コンポーネントに代わるヘルプ情報の表示など、プラットフォーム非依存プログラミング環境向けに適合させたメカニズムを提供していない。

#### 【0029】

表示インターフェースの汎用表示メソッドは、表示インターフェースを実現する表示クラスを定義する際に多大なフレキシビリティが得られるように、非常に一般化されている。したがって、フレームワークを使用して、テキスト表示またはサウンド・クリップなどの異なるヘルプ情報表示スタイルを実現することができる。

#### 【0030】

本発明による表示マネージャ・フレームワークおよび方法は、コンピュータ可読記録媒体に格納されたコンピュータ可読プログラム・コードを含むコンピュータプログラム製品内で実現することができる。

#### 【0031】

【発明の実施の形態】  
好ましい実施例による本発明は、そこで表示マネージャ・ソフトウェアが実行されるコンピュータ・システムの動作を制御するための表示マネージャ・ソフトウェアとして実現される。

#### 【0032】

図1は、本発明が実現可能な典型的なコンピュータ・システムの主要機能要素を示している。コンピュータ・システム1は、中央演算処理装置2と、メモリ3と、システムの一部として組み込まれているかまたは遠隔からアクセスすることができる記憶装置4と、キーボード8、マウス9、またはマイクロフォン10からのユーザ入力を受け取るための入力手段7と、プロセッサ2から受け取ったデータを接続済み表示装置6に転送するためのディスプレイ・コントローラ5と、プロセッサから1つまたは複数のスピーカ13にオーディオ信号を渡すためのオーディオ出力コントローラ14とを含む。

#### 【0033】

システム・メモリ3は、オペレーティング・システム・ソフトウェア11と、現在システム上で実行されているアプリケーション・プログラム12と、本発明による表示マネージャ・ソフトウェア15とを格納する。表示装置6上に表示されるG U Iは、ユーザが作業するためのインターフェース・コントロールを含む所与のウィンドウおよびアイコンの視覚表現をユーザに提示する。オーディオ情報は、スピーカ13により再生することができる。

#### 【0034】

ディスプレイ・コントローラ5は、表示装置6にデータを渡すために使用するソフトウェアおよびハードウェア要素を含む。IBMのOS/2オペレーティング・システムが実行されるコンピュータ・システムでは、オペレーティング・システムの1つの層がデータをディスプレイ・コントローラに渡す。しかし、本発明の好ましい実施例では、このような根元的なオペレーティング・システム表示機能へのマッピングは不要である。

#### 【0035】

好ましい実施例による表示マネージャのコンピュータ・プログラム・コードはJava プログラミング言語で作成されている。当技術分野で既知の通り、標準のJava クラス・ライブラリは、ディスプレイ・コントローラに表示データを渡すために使用する、java.awt.Componentクラス内の表示更新コンポーネントを含む。しかし、この既知の表示更新コンポーネントは、現在、ヘルプ情報を表示するための機能を含んでいない。したがって、Java 環境のU I コンポーネントは、通常、ヘルプ情報の表示をサポートせず、ヘルプ

10

20

30

40

50

情報の表示をサポートしないU Iコンポーネントは、このサポート自体を完全に実現しなければならない。

【0036】

本発明の好ましい実施例は、ユーザ入力装置ポインタに対して敏感なユーザ・インターフェース領域の定義を、表示すべきヘルプ情報の詳細に容易に関連付け、1つまたは複数の適当な表示コンポーネントに関連付けることができるJava環境向けの汎用表示マネージャ・フレームワークを提供し、適切な時点での第1レベルのヘルプ情報の表示は、U Iコンポーネントおよびそれに関連するインターフェース領域に代わって表示マネージャ・フレームワークおよび表示コンポーネントによって自動的に管理される。

【0037】

本発明によるヘルプ情報表示は「モデル/ビュー分離」として当技術分野で既知の設計原理を応用し、感知可能インターフェース領域に関連するデータと表示すべき情報とを定義し管理する機能は情報の表示を果たす機能とは別に実現される。

【0038】

ヘルプ情報の表示では、2つの主要表示管理コンポーネントを使用する。そのうちの第1のものは、表示マネージャ・フレームワークの一部を含む汎用U Iエンティティ定義コンポーネント20である。このU Iエンティティ定義コンポーネントの各インスタンスは特定のU Iエンティティに固有のものであり、インターフェースの感知可能領域とその領域に関連する情報の詳細をカプセル化する。このU Iエンティティ定義コンポーネントは、ポインティング・デバイスの動きなど、関連ユーザ対話の通知を受け取り、それが保持する情報を表示、変更、移動、または除去すべきであると決定したときに通知を出す。

【0039】

このようなコンポーネントの第2のものは、汎用表示メソッドを定義する表示マネージャ・フレームワークのインターフェースを実現するように適合させた表示コンポーネント30である。この表示コンポーネントは、汎用メソッドを使用する情報表示の特定のメソッドの詳細をカプセル化する。このような表示コンポーネントの1つまたは複数は、以下に説明するように、各U Iエンティティ定義コンポーネントに関連付けることができる。選択済み表示コンポーネントは、情報を表示、変更、移動、または除去すべきときに通知を受け取り、適切なアクションを行う。異なる表示コンポーネント・インプリメンテーションは、異なる方法で情報を表示するために異なるアクションを行うことができる。

【0040】

それぞれの明確な感知可能インターフェース領域と個々の表示情報ごとに、U Iエンティティ定義コンポーネントのインスタンスが少なくとも1つずつ存在する。しかし、表示コンポーネントの单一インスタンスはU Iエンティティ定義コンポーネントの多くのインスタンスを処理することができるので、必要なインスタンスは比較的少なくなる可能性がある。実際には、すべての情報が单一様式（ポップアップ・バブル内に表示されたテキストなど）で表示される場合、すべてのU Iエンティティ定義コンポーネントについて单一インスタンスで十分であると思われる。

【0041】

次に、1組のデータ・クラスと、U Iエンティティ定義コンポーネント20を提供するクラスと表示コンポーネント30によって実現される表示インターフェースとの間の定義済み対話とを含む、表示マネージャ・フレームワークの設計について詳細に説明する。

【0042】

InfoTrigger

このクラス20は、画面の感知可能領域とその領域に関連する情報の詳細をカプセル化するためU Iエンティティ定義コンポーネントを実現する。このようなデータは、java.awt.ComponentクラスのU Iコンポーネント40（このU Iコンポーネントは、イベントのターゲットとして、ならびにディスプレイに情報を表示するための媒体として、Java Abstract Windowing Toolkitのjava.awt.Componentクラスを実現するボタン、イメージ・マップ、リスト・ボックスなどにすることができる）、そのU Iコンポーネント上のインタ

10

20

30

40

50

フェース領域、表示すべき情報を含むテキスト・ストリング（またはその他の媒体）への参照として格納される。このフレームワークは、このようなプロパティを設定し照会するためのaccessorメソッドを提供する。

#### 【 0 0 4 3 】

このクラスによって表示されるその他のプロパティとしては、情報表示をトリガするためにマウス・ポインタが停滞（定義済みインタフェース領域内に残留）しなければならないミリ秒単位の間隔と、情報グループ（グループID）と優先順位値を識別する2つの整数とを含む。このような値は、複数の定義済みインタフェース領域が重なっているときにどの情報が表示されるかを決定するために使用する。各グループ内で表示される情報は多くても1組であり、同じグループIDを共用するInfoTriggerの複数のインスタンスがトリガされる場合、最も高い優先順位値を備えたインスタンスの情報だけが表示される。

10

#### 【 0 0 4 4 】

このクラスは、既知のjava.awt.event.MouseListenerおよびjava.awt.event.MouseMotionListenerインターフェースを実現し、すべての関連マウス対話の通知を受け取るためにそれ自身をUIコンポーネントに登録する。また、これは、InfoListenerインターフェース（以下を参照）を実現するクラスをリスナとしてそれに追加することもでき、次にそれらに適切な通知を出す。

#### 【 0 0 4 5 】

InfoTriggerクラスは、トリガされたInfoTriggerインスタンス（以下の参照）の单一静的リストを管理する。このリストは最初は空である。所与の時点では、トリガされたリスト内の正確に1つのInfoTriggerインスタンスが、それに関するInfoTriggerインスタンスがトリガされたリスト内に存在するそれぞれの明確なグループIDごとにそのヘルプ情報を表示させるはずである。この所望の状態は、manageDisplay()というInfoTrigger上の静的メソッドによって実施されるが、このメソッドは明確なグループIDのリストを作成し、見つかったIDごとにmanageDisplayFor()という静的メソッドを呼び出す。メソッドmanageDisplayFor()は、パラメータとしてグループIDを取り、その特定のグループについてトリガされたInfoTriggerインスタンスのうちの正確に1つについてヘルプ情報が表示されることを保証する。メソッドmanageDisplayFor()は、トリガされたInfoTriggerインスタンスのリストを走査して、グループIDを有するインスタンスを識別し、その優先順位を比較し、優先順位が最も高いものを除くすべてのインスタンスについてkill()メソッド（以下を参照）を呼び出す。優先順位が最も高いものの場合、これはshow()メソッドを呼び出す。InfoTriggerインスタンスがトリガされるかまたはトリガされなくなると、そのインスタンスはこのトリガ済みリストに追加され、リストから除去される。InfoTriggerクラスは、正しい表示を維持するために適切な時期にこのような静的メソッドが呼び出されることを保証する。優先順位が低いインスタンスがリストに追加された場合、リストの走査は、現在表示されているインスタンスの表示を続行しなければならないことを確認するだけに過ぎない。

20

#### 【 0 0 4 6 】

適切な挙動およびタイミングを実現するため、InfoTriggerクラスはバックグラウンド・タイマ・スレッドと単純状態モデルを使用し、状態遷移は関連マウス対話およびタイマの満了によってトリガされる。InfoTriggerインスタンスの内部状態は、以下の値の1つを取ることができる。

40

DEAD	情報は表示されない。
READYTOWAIT	できるだけ早くタイマを始動しなければならない。
WAITING	タイマが動作中である。
TRIGGERED	タイマが満了し、このInfoTriggerインスタンスが表示候補になっている。
SHOWING	このInfoTriggerインスタンスが表示中である。

#### 【 0 0 4 7 】

状態変更を引き起こすメソッドは以下の通りである。

50

`startTimer()` -- このメソッドは、InfoTriggerインスタンス用の感知可能領域内でマウスの移動が発生するたびに呼び出され、そのタスクはタイマを始動（または再始動）することである。このInfoTriggerインスタンスと同じグループIDを備えたInfoTriggerインスタンスがトリガ済みリスト内にまったくない場合、これは、現行状態をREADYTOWAITに設定し、タイマ・スレッドに通知する。しかし、このようなInfoTriggerインスタンスがトリガ済みリスト内にある場合、直ちにtrigger()が呼び出される。

`trigger()` -- このメソッドは、タイマが満了したときに以下に示すタイマ・スレッドによって呼び出され、同じグループ内の他のInfoTriggerインスタンスがすでにトリガされている場合にstartTimer()によって呼び出すこともでき、そのタスクはこのInfoTriggerインスタンスを表示候補にすることである。これは、現行状態をTRIGGEREDに設定し、このInfoTriggerインスタンスをトリガ済みリストに追加し、manageDisplayFor()を呼び出してこのInfoTriggerインスタンスのグループIDをそれに渡す。10

`kill()` -- このメソッドは、InfoTriggerインスタンス用の感知可能領域外でマウスの移動が発生するたび、ならびにマウスがUIコンポーネントのインターフェース領域外に完全に移動するたびに呼び出され、そのタスクは情報の表示を終了することである。これは、このInfoTriggerインスタンスをトリガ済みリストから除去し、`hide()`を呼び出す。

`show()` -- このメソッドは、状態をSHOWINGに設定し、このInfoTriggerインスタンスのすべての登録リスナ上で`showInfo()`（以下を参照）を呼び出す。

`hide()` -- このメソッドは、優先順位がより高いInfoTriggerインスタンスが優先されたためにこれがSHOWINGから呼び出された場合に状態をTRIGGEREDに設定するか、またはメソッド`kill()`によって呼び出された場合に状態をDEADに設定する。次にこれは、このInfoTriggerのすべての登録リスナ上で`hideInfo()`（以下を参照）を呼び出し、manageDisplayFor()を呼び出してこのInfoTriggerのグループIDをそれに渡す。20

#### 【0048】

他のメソッドとしては、InfoListenerを実現するクラスのインスタンスを追加し除去するためのメソッドと、プロパティ値を獲得し設定するためのaccessorメソッドを含む。

#### 【0049】

タイマ・スレッドは、イベントが発生するのを待ち、現行状態に応じてこのようなイベントに応答する。また、必要ではない場合にスレッドを終了することができ、再び必要になったときにスレッドを再始動するメカニズムも存在する。可能なそれぞれの状態ごとに行われるアクションは以下の通りである。30

DEAD           タイマ・スレッドは不要である。不定待機に入る（または終了するように手配する）。

READYTOWAIT   タイマを始動すべきである。タイマが始動されると、状態はWAITINGに設定され、このInfoTrigger用に設定されたタイムアウトによって決定される時間間隔の間、待機に入る。

WAITING       この状態からタイマが満了すると、trigger()が呼び出され、タイマ・スレッドが不定待機に入る。

TRIGGERED      タイマは不要である。不定待機に入る。

SHOWING        タイマは不要である。不定待機に入る。40

#### 【0050】

##### InfoListener

このインターフェースは、特定の変更が行われたときにInfoTriggerインスタンスによって呼び出されるメソッドを定義する。InfoListenerを実現するクラスのインスタンスは、InfoTrigger上の`addInfoListener()`メソッドに参照を渡すことによってInfoTriggerインスタンスに登録することができ、`removeInfoListener()`メソッドに参照を渡すことによって登録解除することができる。30

#### 【0051】

このインスタンスによって定義されるメソッドと、それが呼び出される状況をここに示す。各メソッドは、以下に記載するInfoEventのインスタンスをパラメータとして受け取る50

。

showInfo() -- このメソッドは、その情報を表示することをInfoTriggerインスタンスが決定したときに呼び出される。情報表示は直ちに行わなければならない。

hideInfo() -- このメソッドは、その情報をもはや表示しないことをInfoTriggerインスタンスが決定したときに呼び出される。情報表示は直ちに除去しなければならない。

changeInfo() -- このメソッドは、情報が表示されている間にInfoTriggerインスタンスについてテキスト・ストリングまたは他の媒体ディスクリプタの内容などの表示パラメータが変化した場合に呼び出される。ディスプレイに対する更新を行う必要がある可能性がある。

moveInfo() -- このメソッドは、情報が表示されている間にマウス・ポインタ位置が変化した場合に呼び出される。一部の表示スタイルでは、マウスによって情報表示を移動することを選ぶことができ、その場合、これは、これが行われることを保証する機会になる。 10

#### 【 0 0 5 2 】

##### InfoEvent

InfoEvent 5 0 のインスタンスは、InfoTriggerについて特定の変更が行われたときに上記の4つのメソッドに渡される。InfoEventクラスはjava.swt.evnet.Eventを拡張し、イベント・ソースはいつでもそれに関する変更が行われたInfoTriggerインスタンスになる。

さらに、このクラスはgetMousePosition()メソッドをサポートし、このメソッドは変更がトリガされたときにマウス・ポインタ位置を返す。 20

#### 【 0 0 5 3 】

##### InfoPopup

InfoPopupクラスは、InfoListenerインターフェースの1つの表示コンポーネント・インプリメンテーション例を表している。したがって、InfoPopupは、上記のInfoListenerインターフェースの一部であるshowInfo()、hideInfo()、changeInfo()、moveInfo()に関する抽象メソッド定義を実現する特定のメソッドを含む。 30

#### 【 0 0 5 4 】

InfoPopup上のshowInfo()メソッドは、InfoPopupクラスに固有であり、現行入力装置ポインタ位置を取り、そのポインタ位置に関連するヘルプ・テキストを表示するウィンドウをオープンするメソッドを呼び出す。この表示クラス固有メソッドに関するプロパティは、InfoPopupによって実現されるヘルプ表示スタイルが薄い黄色の背景に単純な矩形ポップアップ・ウィンドウを表示し、都合の良い割合になるように黒のテキストがワードラップし、その回りに細い黒の境界が付くような、ウィンドウ・サイズと形状、テキスト・フォントとカラーを含む。これは、外観上は、当技術分野で既知のポップアップまたはホバー・ヘルプの多くの一般的なインプリメンテーションと同様である。 30

#### 【 0 0 5 5 】

このInfoPopupクラスと、追加の事前定義表示クラスは、本発明の表示マネージャ・フレームワークとともにパッケージ化することができる。さらに、InfoPopupクラスは、InfoTriggerのインスタンスに容易に関連付けられるデフォルト表示コンポーネントになるよう定義される。事前定義表示コンポーネントと、デフォルト表示コンポーネントにアクセスするための単純なメカニズムをこのように用意することにより、特定のカスタム表示スタイルが必要でない限り、カスタマイズした表示コンポーネントを定義する必要なしに、本発明の表示マネージャ・フレームワークを使用することができる。 40

#### 【 0 0 5 6 】

デフォルト表示コンポーネントは、パラメータとして真を指定してInfoTrigger呼出しenableDefaultDisplay()上でメソッドを呼び出すことによってInfoTriggerの各インスタンスについて使用可能にし、パラメータとして偽を指定してそのメソッドを呼び出すことによって使用禁止にすることができる。このデフォルト・スタイルは、問題のInfoTriggerインスタンスのリスナとして明示的に登録されるInfoListenerのインプリメンテーションのインスタンスに追加して、またこのようなインスタンスとは無関係に適用される。

#### 【 0 0 5 7 】

50

次に、本発明によりヘルプ情報を表示するための一連の動作について説明する。

#### 【0058】

初期設定ステップは、U Iコンポーネントがユーザ・インターフェースに追加されたときにU Iコンポーネントに代わって情報表示を処理するように表示マネージャを準備する。本発明の好ましい実施例では、U Iコンポーネントはそれぞれ、そのヘルプ・データとそれに関連するインターフェース領域定義を表示マネージャに供給することを担当し、ユーザまたはアプリケーション・アセンブラーは特定の表示コンポーネントとの関連付けを定義する。また、各U Iコンポーネントは、U Iコンポーネントのリスナとしてこの表示コンポーネントを適用するコンテナ・オブジェクトへの参照も含む。

#### 【0059】

使用中に表示マネージャは、マウス・ポインタまたはその他のユーザ入力装置ポインタが所定の事前設定時間間隔の間、感知可能領域として定義されたインターフェースの領域内に残留するときに、情報を表示させる。各感知可能領域はグループIDを有する。マウス・ポインタが感知可能領域を出ると、インターフェースから情報が除去される。しかし、同じグループIDを共有する他の領域に入ることにより、マウス・ポインタがある感知可能領域を出る場合、新しい領域の情報が直ちに現れ、先行技術のヘルプ・システムでしばしば見られるような遅延は発生しない。

#### 【0060】

本発明の好ましい実施例では、アプリケーション・プログラマは、表示マネージャのどの表示コンポーネントを使用すべきかを決定し、その表示コンポーネントが使用するために適切な表示データを供給しなければならない。次に、必要な表示コンポーネントへの参照により、各U Iエンティティ定義コンポーネント・インスタンスが作成される。

20

#### 【0061】

本発明の代替実施例では、以下のように、U Iコンポーネントがインターフェースに追加されたときにビジュアル・ビルダ・ツールを使用する、表示コンポーネントのユーザ選択をサポートする。第1に、アプリケーション開発者は、複数の異なる表示コンポーネント用の適切な表示データを定義して使用しなければならない。U Iコンポーネントはその表示データを表示マネージャに供給し、次にビジュアル・ビルダ・ツールは選択可能表示コンポーネントのリストをユーザに提示する。このリストは、それに関して表示情報が定義された1組の表示コンポーネントを含むことが好ましいが、リストを生成する際に他のヒューリスティックを適用することができる（表示コンポーネント・タイプとU Iエンティティ・タイプとの事前定義関連付けに応じた表示コンポーネントのリスト表示など）。

30

#### 【0062】

好ましい表示コンポーネントを選択して所望の表示スタイルを実現するためにアプリケーション・アセンブラーまたはエンド・ユーザにこのような機構を提供すると、一貫したインターフェースの作成が可能になり、ユーザが本当に必要とする表示スタイルを実現するためのカストマイズが可能になる。

#### 【0063】

InfoTriggerクラスの説明で前述したように、感知可能領域が重なり、異なる優先順位を有するようにするためのメカニズムが含まれる。所与のグループIDを有する多くの1つの感知可能領域は一度にその情報を表示し、したがって、複数の情報表示が潜在的に混乱するのを回避する。その優先順位は、一連のウィンドウの以前の活動化の順序を識別するz値に基づくものではないが、事前設定優先順位値（U Iエンティティ・タイプに依存する可能性がある）に基づくものである。

40

#### 【0064】

上記の本発明の好ましい実施例では、U Iコンポーネントは表示マネージャ・フレームワークにそれ自身を登録するためのアクションを行う。他の代替実施例では、表示マネージャが登録プロセスを開始する。表示マネージャは、U Iコンポーネントがユーザ・インターフェースに追加されたときに呼び出され、新しいU Iコンポーネント用の画面領域の予約や、イベント経路指定用の規則の変更などの操作を実行する。U Iコンポーネントが追加

50

されたときに呼び出されると、表示マネージャは、それがその表示情報と関連のユーザ・インターフェース領域定義を定義するデータを返すための要求を新しいUIコンポーネントに出すように適合させている。本発明を使用するように適合されたUIコンポーネントは、表示情報項目（ヘルプ・テキスト、サウンド・クリップ、イメージのアニメーション・シーケンスなど）と、個々の表示情報に関連するインターフェース領域の定義のリストを表示マネージャに返すメソッドを実現する。次に表示マネージャは、各UIコンポーネントごとに、このような情報項目と領域定義をカプセル化する汎用UIエンティティ定義コンポーネントの新しいインスタンスを作成する。

#### 【0065】

まとめとして、本発明の構成に関して以下の事項を開示する。

10

#### 【0066】

(1) コンピュータ・システム・ユーザ・インターフェースによるUIエンティティに関するヘルプ情報の表示をサポートするための表示マネージャ・フレームワークであって、前記UIエンティティに関するインターフェース領域定義と関連のヘルプ情報をカプセル化するためのUIエンティティ定義コンポーネントと、それぞれの表示スタイルに応じて複数の表示コンポーネントによって実現するための1つまたは複数の汎用ヘルプ情報表示メソッドを有するセットを提供する表示インターフェースとを含む、表示マネージャ・フレームワーク。

(2) 複数の選択可能表示コンポーネントを含み、それぞれが複数の事前定義表示スタイルのそれぞれ異なる1つに応じてヘルプ情報を表示するために表示インターフェースを実現する、上記(1)に記載のフレームワークを実現する表示マネージャ。

20

(3) UIエンティティの1つに関する情報を表示するために所望の表示スタイルに応じて前記表示コンポーネントの1つを選択する手段と、

前記UIエンティティとのユーザ対話に対する応答として選択済み前記表示コンポーネントを始動する手段とを含む、上記(2)に記載の表示マネージャ。

(4) 前記UIエンティティ定義コンポーネントがオブジェクト・クラスを含み、そのインスタンスのそれぞれが前記UIエンティティに関するインターフェース領域定義と関連のヘルプ情報を含み、それぞれの定義済みインターフェース領域とのユーザ対話に対する応答として通知を受け取り、それに応答して、前記表示コンポーネントの選択済みの1つに通知を送り、それにより、前記選択済み表示コンポーネントを始動するように適合している、上記(3)に記載の表示マネージャ。

30

(5) 前記UIエンティティ定義コンポーネントのインスタンスのそれぞれがグループIDと優先順位値とを有し、任意の一時点では同じグループIDを有する複数の通知済みUIエンティティ定義コンポーネント・インスタンスのうちの優先順位が最も高いインスタンスのみに関するヘルプ情報が表示されることを保証する手段を前記フレームワークが含む、上記(4)に記載の表示マネージャ。

(6) コンピュータ・システム・ユーザ・インターフェースによるUIエンティティに関するヘルプ情報の表示を処理するための表示マネージャにあって、

それが複数の事前定義表示スタイルのそれぞれ異なる1つに応じてヘルプ情報を表示するための複数の表示コンポーネントと、

40

前記UIエンティティの1つに関連する情報を表示するために前記表示コンポーネントの1つを選択する手段と、

前記UIエンティティとのユーザ対話に対する応答として選択済み前記表示コンポーネントを始動する手段とを含む、表示マネージャ。

(7) 前記UIエンティティ用のインターフェース領域定義と、関連の表示情報をカプセル化するための1つまたは複数のUIエンティティ定義コンポーネントを含み、前記UIエンティティ定義コンポーネントが、ユーザ対話通知を受け取り、選択済み前記表示コンポーネントを始動するように通知を出すことにより前記通知に応答するように適合している、上記(6)に記載の表示マネージャ。

(8) UIエンティティに関するコンピュータ・システム・ユーザ・インターフェースによ

50

るヘルプ情報表示の方法であって、それぞれが複数の事前定義表示スタイルのそれぞれ異なる1つに応じてヘルプ情報を表示するための複数の選択可能表示コンポーネントを含む表示マネージャによって実現され、

前記U Iエンティティがインターフェースに追加されたことに対する応答として、表示情報と前記U Iエンティティによって提供されるインターフェース領域定義とをカプセル化するU Iエンティティ定義クラスのインスタンスを作成するステップと、

前記表示スタイルの1つに応じて前記情報を表示するために前記表示マネージャの表示コンポーネントを選択するステップと、

ユーザ入力装置ポインタが移動して前記インターフェース領域との事前定義接近関係になつたときに、ユーザ対話に対する応答として前記情報を表示するために選択済み前記表示コンポーネントを始動するステップとを含む方法。

(9) コンピュータ・システム・ユーザ・インターフェースによるU Iエンティティに関するヘルプ情報の表示をサポートするためのプラットフォーム非依存表示マネージャ・フレームワークであつて、

U Iエンティティに関するU I領域定義と関連のヘルプ情報をカプセル化するためのU Iエンティティ定義クラスと、

前記ヘルプ情報を表示するために前記U Iエンティティ定義クラスのインスタンスによって呼び出されるように適合された1つまたは複数の表示コンポーネント内で実現するための1つまたは複数の汎用表示メソッドを提供する表示インターフェースとを含む、表示マネージャ・フレームワーク。

(10) 複数の選択可能表示コンポーネントを含み、前記U Iエンティティ定義クラス・インスタンスによって呼び出されたときにそれぞれが複数の事前定義表示スタイルのそれぞれ異なる1つに応じてヘルプ情報を表示する、上記(9)に記載の表示マネージャ・フレームワーク。

#### 【図面の簡単な説明】

【図1】本発明を実現可能なコンピュータ・システムの概略表現である。

【図2】本発明の一実施例による表示マネージャ・フレームワークのクラス構造を示す図である。

#### 【符号の説明】

1 コンピュータ・システム

30

2 中央演算処理装置、プロセッサ

3 メモリ

4 記憶装置

5 ディスプレイ・コントローラ

6 表示装置

7 入力手段

8 キーボード

9 マウス

10 マイクロフォン

11 オペレーティング・システム・ソフトウェア

40

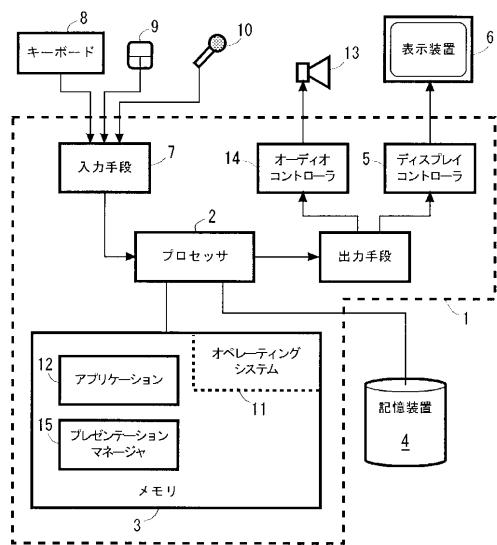
12 アプリケーション・プログラム

13 スピーカ

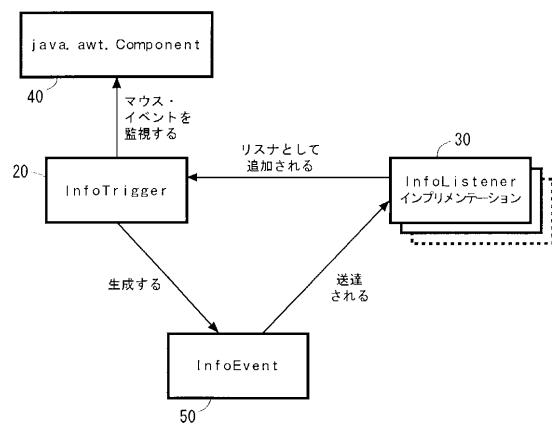
14 オーディオ出力コントローラ

15 表示マネージャ・ソフトウェア

【図1】



【図2】



---

フロントページの続き

- (72)発明者 デービッド・クラーク  
イギリス シー・ヴィー34 5ユー・アール ワーウィック ウッドローズ・パーク ボスウェル・グローブ 54
- (72)発明者 アダム・アレクサンダー・ドブソン  
イギリス エス・エス7 5ピー・エイ エセックス ベンフリート カンバーランド・アベニュー  
- 122
- (72)発明者 サイモン・ピーター・グッドチャイルド  
イギリス ジェイ・イー2 4キュー・ディー チャネル・アイランズ ジャージー セント・ヘリエ ドン・テラス 4 フラット 3
- (72)発明者 アンドリュー・ジョン・スミス  
イギリス ワーウィックシャー ワーウィック ウエルズボーン アンソン・クローズ 8
- (72)発明者 レイモンド・トレイナー  
イギリス ワーウィックシャー ワーウィック サマー・ウェイ 25

## 合議体

審判長 川寄 健

審判官 小林 正明

審判官 竹井 文雄

- (56)参考文献 特開平9-190287 (JP, A)  
特開平5-2476 (JP, A)  
特開平8-161102 (JP, A)

## (58)調査した分野(Int.Cl., DB名)

G06F3/00