(19)日本国特許庁(JP)　(12)公開特許公報(A)　(11)特許出願公開番号
特開2005-196669
(P2005-196669A)
(43)公開日　平成17年7月21日(2005.7.21)

(51)Int.Cl.[7]		F I			テーマコード（参考）
G06F	9/44	G06F	9/06	620C	5B069
G06F	3/00	G06F	3/00	651A	5B076
G06F	3/14	G06F	3/14	310A	5C053
H04N	5/91	H04N	5/91	N	5E501

審査請求 未請求　請求項の数 16　ＯＬ　（全 17 頁）

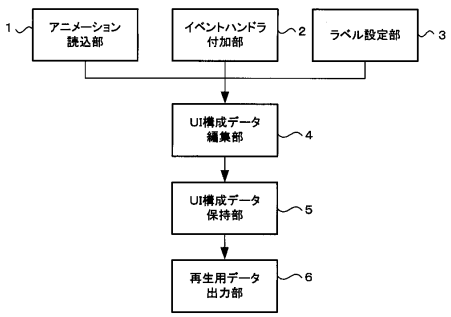
(21)出願番号　特願2004-4504(P2004-4504)
(22)出願日　平成16年1月9日(2004.1.9)

(71)出願人　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　100122884
弁理士　角田　芳末
(74)代理人　100113516
弁理士　磯山　弘信
(72)発明者　河野　宗一
東京都品川区北品川６丁目７番３５号　ソニー株式会社内
(72)発明者　松田　晃一
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】オーサリング方法、オーサリングシステム、データ再生方法及びデータ再生システム

(57)【要約】　（修正有）
【課題】表現能力の異なる機器に対するＧＵＩの移植性を向上させるオーサリングシステムを提案する。
【解決手段】動画及び／又は静止画を含むデジタル画像データを利用したＧＵＩのオーサリングシステムであって、既存のデジタル画像データを読み込む画像データ読込部と、読込んだデジタル画像データに対してイベント及び対応する処理を記述したイベントハンドラを設定するイベントハンドラ付加部と、読み込んだデジタル画像データ内の所定の時刻に対してラベル付けを行うラベル設定部と、イベントハンドラ及びラベルからイベント処理ルールを作成し、該イベント処理ルール及び読み込んだデジタル画像データより構成されるＵＩ構成データを保持するＵＩ構成データ保持部と、ＵＩ構成データ保持部のＵＩ構成データをデータ再生システムにて再生可能なフォーマットで出力する再生用データ出力部より構成する。
【選択図】図１



【特許請求の範囲】
【請求項１】
動画及び／又は静止画を含むデジタル画像データを利用したＧＵＩのオーサリング方法であって、
既存のデジタル画像データを読み込むステップと、
読み込んだデジタル画像データに対してイベント及び対応する処理を記述したイベントハンドラを設定するステップと、
前記読み込んだデジタル画像データ内の所定の時刻に対してラベル付けを行うステップと、
前記イベントハンドラ及び前記ラベルからイベント処理ルールを作成し、該イベント処理ルール及び前記読み込んだデジタル画像データよりＵＩ構成データを構成するステップとを有する
ことを特徴とするオーサリング方法。
【請求項２】
前記イベント処理ルール内の時間に関する記述について、前記ラベルにより表現する
ことを特徴とする請求項１記載のオーサリング方法。
【請求項３】
前記イベント処理ルールにおいて、前記読み込んだデジタル画像データの再生位置により前記イベント処理ルールのイベント処理方法が変更されるよう記述する
ことを特徴とする請求項１記載のオーサリング方法。
【請求項４】
動画及び／又は静止画を含むデジタル画像データを利用したＧＵＩをオーサリングするオーサリングシステムであって、
既存のデジタル画像データを読み込む画像データ読込部と、
読み込んだデジタル画像データに対してイベント及び対応する処理を記述したイベントハンドラを設定するイベントハンドラ付加部と、
前記読み込んだデジタル画像データ内の所定の時刻に対してラベル付けを行うラベル設定部と、
前記イベントハンドラ及び前記ラベルからイベント処理ルールを作成し、該イベント処理ルール及び前記読み込んだデジタル画像データより構成されるＵＩ構成データを保持するＵＩ構成データ保持部と、
前記ＵＩ構成データ保持部のＵＩ構成データを、出力先のデータ再生システムにて再生可能なフォーマットで出力する再生用データ出力部とを有する
ことを特徴とするオーサリングシステム。
【請求項５】
前記イベント処理ルール内の時間に関する記述について、前記ラベルにより表現する
ことを特徴とする請求項４記載のオーサリングシステム。
【請求項６】
前記イベント処理ルールにおいて、デジタル画像データの再生位置により前記イベント処理ルールのイベント処理方法が変更されるよう記述する
ことを特徴とする請求項４記載のオーサリングシステム。
【請求項７】
オーサリングシステムにより開発された動画及び／又は静止画を含むデジタル画像データを利用したＧＵＩの、デジタル画像データ及びイベント処理ルールから構成される再生用フォーマットのＵＩ構成データを再生するデータ再生方法であって、
イベントの発生を検出するステップと、
発生したイベントが終了イベントではなく、かつ前記イベント処理ルールに適合する場合に、前記発生イベントに対応した前記イベント処理ルールの処理内容に従って前記デジタル画像データの再生位置を変更する、及び／又は、外部プログラムを起動させる処理を行うステップと、

前記処理が行われたデジタル画像データを再生するステップとを有する
ことを特徴とするデータ再生方法。
【請求項８】
発生したイベントが終了イベントではなく、かつ前記イベント処理ルールに適合する場合に、前記イベント処理ルールの処理内容に前記デジタル画像データの属性を変更する処理が含まれるときは、前記デジタル画像データの属性を変更するステップを有する
ことを特徴とする請求項７記載のデータ再生方法。
【請求項９】
前記読み込まれたデジタル画像データが複数の場合、前記デジタル画像データの処理内容に従って前記処理を行った後に、一つのデジタル画像データに合成するステップを有する
ことを特徴とする請求項７又は８記載のデータ再生方法。
【請求項１０】
前記イベント処理ルール内の時間に関する記述が前記ラベルにより表現されている
ことを特徴とする請求項７記載のデータ再生方法。
【請求項１１】
前記デジタル画像データの再生位置により前記イベント処理ルールのイベント処理方法が変更される
ことを特徴とする請求項７記載のデータ再生方法。
【請求項１２】
オーサリングシステムにより開発された動画及び／又は静止画を含むデジタル画像データを利用したＧＵＩの再生用フォーマットのＵＩ構成データを再生するデータ再生システムであって、
デジタル画像データ及びイベント処理ルールから構成される再生用のＵＩ構成データを読み込む再生用データ読込部と、
前記デジタル画像データを保持する画像データ保持部と、
前記イベント処理ルールを管理するイベント処理ルール保持部と、
イベント発生を通知するイベント検出部と、
前記イベント検出部より通知されるイベントに対応して前記イベント処理ルール保持部に記述された処理を行うイベント処理部と、
前記イベント処理部からの指示により前記デジタル画像データの再生位置を変更する画像データ再生位置変更部と、
読み込まれたデジタル画像データが複数の場合、一つのデジタル画像データに合成する画像データ合成部と、
前記画像データ合成部により作成されたデジタル画像データを再生する画像データ再生部と
から構成されることを特徴とするデータ再生システム。
【請求項１３】
前記イベント処理部は、前記イベント処理ルールに記述された処理内容に従い外部プログラムを起動させる
ことを特徴とする請求項１２記載のデータ再生システム。
【請求項１４】
前記イベント処理部からの指示により、前記デジタル画像データの属性を変更する画像データ変更部を有する
ことを特徴とする請求項１２記載のデータ再生システム。
【請求項１５】
前記イベント処理ルール内の時間に関する記述が前記ラベルにより表現されている
ことを特徴とする請求項１２記載のデータ再生システム。
【請求項１６】
前記画像データ再生部で再生される前記デジタル画像データの再生位置により前記イベ

ント処理ルールのイベント処理方法が変更される
ことを特徴とする請求項１２記載のデータ再生システム。
【発明の詳細な説明】
【技術分野】
【０００１】
本発明は、動画及び／又は静止画を含むデジタル画像データを利用したグラフィカル・ユーザ・インターフェイス（GUI；Graphical User Interface）のオーサリングに適用して好適なオーサリング方法及びオーサリングシステム、並びに、このオーサリング作業により開発されたＧＵＩを実行するのに好適なデータ再生方法及びデータ再生システムに関する。
【背景技術】
【０００２】
従来、テレビジョン受像機やパーソナルコンピュータ、携帯電話等に代表されるようなディスプレイ装置搭載機器において、ＧＵＩによる操作が一般的である。これらのＧＵＩでは、ユーザに対して操作方法を説明する場合や、操作結果をユーザに伝える場合など、ユーザビリティ向上のために、例えばアニメーションなどの画像を用いることが効果的である。
【０００３】
従来のアニメーション付きＧＵＩ開発環境では、デザイナーがＧＵＩのデザイン及びその挙動を考え、このデザイン及び挙動を実現するプログラムをプログラマが作成している。例えば、プログラム内容は、まずボタン等の外見を作り、ボタンをディスプレイ上のどこに配置し、ボタンが押された場合の処理を記述するというような方法である。このような従来の方法をＯＯＡ（Object Oriented Authoring）と呼ぶことにする。
【０００４】
このＯＯＡ方式のものとして、デジタル素材データを、内部にデータと関係式とを有する部品として、配置・編集してマルチメディアコンテンツを作成する部品編集機能を備えたオーサリングツールが提案されている。この部品は複数の親部品と子部品とを組み合わされた階層構造とされ、子部品が親部品の表示座標系を継承することにより親部品の変形動作に応じて子部品が変形動作するように構成されている（例えば、特許文献１参照。）。
【特許文献１】特開２００２－１５３３３号公報
【発明の開示】
【発明が解決しようとする課題】
【０００５】
デザイナーはＧＵＩのデザインイメージをプログラマに伝達する方法として、静止画や動画を用いている。プログラマはそれらの絵からボタンやキャラクタといったパーツのレイアウト情報や、パーツがどのように移動・出現・消滅するかといったアニメーションに関する情報を、これらの静止画や動画から読み取り、予想する必要がある。特にアニメーションは正確に数式化されるわけではなく、デザイナーのイメージをプログラマに正確に伝達することは困難であり、ＧＵＩ開発期間の長期化を引き起こす原因となっていた。
【０００６】
一方、ある機器用に作成したＧＵＩを、利用可能な色数や演算能力といった表現能力の異なる機器に移植しようとした場合、ＧＵＩのプログラムを書き直す必要があるという問題があった。
【０００７】
斯かる点に鑑み、本発明は、デザイナーが自分で考案した又は他機器で作成されたデザインを、直接動画及び／又は静止画を含むデジタル画像データを利用したＧＵＩのオーサリングに利用できるようにするとともに、表現能力の異なる機器に対するＧＵＩの移植性を向上させることを目的とするものである。
【課題を解決するための手段】

【０００８】
上記課題を解決し目的を達成するため、本発明は、動画及び／又は静止画を含むデジタル画像データを利用したＧＵＩのオーサリングにおいて、既存のデジタル画像データを読み込み、読み込んだデジタル画像データに対してイベント及び対応する処理を記述したイベントハンドラを設定し、読み込んだデジタル画像データ内の所定の時刻に対してラベル付けを行い、このイベントハンドラ及びラベルからイベント処理ルールを作成し、イベント処理ルール及び読み込んだデジタル画像データよりＵＩ構成データを構成することを特徴とする。
【０００９】
斯かる本発明によれば、デジタル画像データと、ラベル、イベントハンドラを分離することによって、デジタル画像データ（デザイン）の開発とＧＵＩのアプリケーションプログラムの開発を平行して行うことができる。
【００１０】
また、本発明は、上述イベント処理ルール内の時間に関する記述について、ラベルにより表現することを特徴とする。
【００１１】
斯かる本発明によれば、時刻の代わりにラベルを基にしてイベント処理ルールを記述することにより、デジタル画像データとイベント処理ルールを分離・独立して取り扱うことができる。
【００１２】
また、本発明は、上述イベント処理ルールにおいて、読み込んだデジタル画像データの再生位置によりイベント処理ルールのイベント処理方法が変更されるよう記述することを特徴とする。
【００１３】
斯かる本発明によれば、デジタル画像データの再生位置によってイベント処理方法を変更することができるので、再生側においてより柔軟なＧＵＩを実現することができる。
【００１４】
また、本発明は、オーサリングシステムにより開発された動画及び／又は静止画を含むデジタル画像データを利用したＧＵＩの、デジタル画像データ及びイベント処理ルールから構成される再生用フォーマットのＵＩ構成データを再生する際に、イベントの発生を検出し、発生したイベントが終了イベントではなく、かつこのイベント処理ルールに適合する場合に、発生したイベントに対応したイベント処理ルールの処理内容に従ってデジタル画像データの再生位置を変更する、及び／又は、外部プログラムを起動させる処理を行い、これらの処理が行われたデジタル画像データを再生することを特徴とする。
【００１５】
斯かる本発明によれば、ＧＵＩのデジタル画像データに対し発生したイベントに応じてイベント処理ルールに記述された処理が行われるので、ＧＵＩを簡易に実行することができる。
【００１６】
また、本発明は、上述の発生したイベントが終了イベントではなく、かつイベント処理ルールに適合する場合に、イベント処理ルールの処理内容にデジタル画像データの属性を変更する処理が含まれるときは、デジタル画像データの属性を変更することを特徴とする。
【００１７】
斯かる本発明によれば、再生側で変更可能なデジタル画像データの属性を再生時に変更することで、オーサリングシステム側で作成すべきデジタル画像データ及びデータ再生システム側で用意すべきデジタル画像データをそれぞれ削減することができる。
【００１８】
また、本発明は、上述読み込まれたデジタル画像データが複数の場合、デジタル画像データの処理内容に従って上述の処理を行った後に、一つのデジタル画像データに合成する

ことを特徴とする。

【００１９】

斯かる本発明によれば、複数のデジタル画像データを一つのデジタル画像データに合成することで容易に再生することができる。

【００２０】

また、本発明は、上述イベント処理ルール内の時間に関する記述がラベルにより表現されていることを特徴とする。

【００２１】

斯かる本発明によれば、デジタル画像データに対し時刻の代わりにラベルを基にしたイベント処理ルールの設定がなされていることにより、デジタル画像データとイベント処理ルールを分離・独立して取り扱うことができる。

【００２２】

また、本発明は、デジタル画像データの再生位置によりイベント処理ルールのイベント処理方法が変更されることを特徴とする。

【００２３】

斯かる本発明によれば、デジタル画像データの再生位置によってイベント処理方法が変更されるので、より柔軟なＧＵＩを実現することができる。

【発明の効果】

【００２４】

斯かる本発明によれば、ＧＵＩ開発における大部分の作業を、例えば既存のアニメーション作成ツール等を用いたデジタル画像データ作成により実現できるので、簡易にデジタル画像データが付いたＧＵＩを作成することができる。

【００２５】

また、本発明において、デザイナーがアニメーション作成ツール等を用いて作成した既存のデジタル画像データが利用できるので、イメージ通りのＧＵＩのアニメーションの挙動を実現できるようになり、プログラマとの意思疎通の失敗によるＧＵＩ開発期間の長期化等のリスクが発生することがないという効果がある。

【００２６】

また、本発明において、例えば図１のシステム構成図に示すように、ＧＵＩ開発のために必要な機能が単純なため、オーサリングツールを簡易に実現することができるという効果がある。

【００２７】

また、本発明において、ラベル付けによりデジタル画像データ中のある瞬間を状態として切り出し、状態（ラベル）をベースにイベント処理ルールが設定されていることによって、デジタル画像データとイベント処理ルールを分離・独立して取り扱うことが可能となり、入力デジタル画像データをデザインや挙動、長さなどが異なる別のデジタル画像データに容易に差し替えることができる。さらに、例えば処理性能や利用可能な色数といった表現能力の異なる機器に対してＧＵＩの移植作業を行う場合にも、イベント処理ルールを変更する必要がないので、効率的に移植作業を進めることができるとともに、移植性が向上するという効果がある。

【発明を実施するための最良の形態】

【００２８】

以下、図１～図８を参照して、本発明の実施の形態の例について説明する。本発明のオーサリング方法は、既に存在する動画や静止画のデジタル画像データに対し、イベントハンドラ及びラベルを設定することによりユーザ・インターフェイス（UI;User Interface）を構築するものである。本例のような方法をＡＯＡ（Animation Oriented Authoring）と呼ぶこととする。

【００２９】

図１は、本発明の一実施の形態の例に係る、デジタル画像データとして動画（アニメーション）を基にしたＧＵＩをオーサリングするオーサリングシステムの構成図を示すもの

である。上述オーサリング方法（ＡＯＡ）は、このオーサリングシステムを、例えばＰＣ（パーソナルコンピュータ）などの情報処理装置にインストールされたオーサリングツール（アプリケーションソフトウェア）に組み込むことで実現される。

【００３０】

上述オーサリングシステムを搭載する情報処理装置は、例えば、図示しない装置全体の制御を司るとともにオーサリング処理等の所定の処理や演算を行うＣＰＵ（Central Processing Unit）などの演算処理装置と、演算処理装置が所定の制御及びオーサリング処理等を実行するためのプログラムが記録されているＲＯＭ（Read Only Memory）と、ＣＰＵがプログラム実行時のワークエリアとして利用する主記憶装置としてのＲＡＭ（Random Access Memory）を備えている。ＲＡＭとのアクセスは、演算処理装置の他、ＤＭＡ（Direct Memory Access）方式により他デバイス間と直接行えるようにしてもよい。

【００３１】

なお、オーサリングを行うためのプログラム（オーサリングツール）は、この装置内にフラッシュメモリ等の不揮発性メモリを設けて記録されているものでもよい。また、ＣＤ－ＲＯＭ等のリムーバブルな記録媒体に記録されたオーサリング用アプリケーションソフトウェアをＲＡＭに読み込み、ＲＡＭに読み込まれたアプリケーションソフトウェアを演算処理装置が実行するようなものでもよい。

【００３２】

図１に示されるオーサリングシステムは、例えばアニメーション作成ツールで作成された既存のアニメーション（オリジナルデータ）を読み込むアニメーション読込部（画像データ読込部）１と、読み込んだアニメーションに対してユーザ操作によりイベントハンドラを設定するイベントハンドラ付加部２と、アニメーション内の時刻に対してラベル付けを行うラベル設定部３と、アニメーション読込部１、イベントハンドラ付加部２及びラベル設定部３のモジュールを利用してＵＩ構成データの編集が行われるＵＩ構成データ編集部４と、編集されたＵＩ構成データを保持するＵＩ構成データ保持部５と、ＵＩ構成データをＵＩ再生用フォーマットで出力する再生用データ出力部６からなるものである。

【００３３】

この再生用データ出力部６から出力されるＵＩ構成データのフォーマットは、後述するディスプレイ装置を備えた据え置き型機器又は携帯型機器で動作するプレーヤーで再生可能なファイルフォーマットであって、上述のオーサリングシステムで作成した該ファイルフォーマットのＵＩ構成データをそれらの機器が読み込むことにより、機器の電源投入やキー入力等のイベント発生に応じて読み込んだＵＩ構成データが再生されるというものである。このとき、オリジナルデータのアニメーションは変更されず、再生側においてイベント処理内容に従い属性情報が付加されたインスタンスとして再生される。

【００３４】

ＧＵＩ開発者は、例えば、Ｌｉｇｈｔ Ｗａｖｅ ３Ｄ（登録商標）などの市販アニメーション作成ツールにより作成されたアニメーション（オリジナルデータ）を、リムーバブルな記録媒体やインターネット等の通信回線などを介して、あるいは機器内で直接データ通信を行い、図１のオーサリングシステムのアニメーション読込部１に読み込ませる。そして、読み込んだアニメーションに対し、マウスやキーボード等の入力装置を用いラベル及びイベントハンドラを設定してイベント処理ルールを作成する。作成したイベント処理ルールはＵＩ構成データ編集部４にてアニメーションとともにＵＩ構成データとして編集され、ＵＩ構成データ保持部５に記録される。

【００３５】

図１のＵＩ構成データ編集部４にて編集されるＵＩ構成データのアニメーション、ラベル、イベントハンドラ間の相互の関連付けは時間を基に行われる。アニメーションは、開始時刻、表示時間の情報を持ち、例えば、１０～５０ｍｓの間表示、５０～１００ｍｓの間非表示、１００～１２０ｍｓの間表示などのように、各アニメーションについて表示される時間及び表示されない時間の情報より表示時間テーブルが作成される。この表示時間テーブルはプレーヤーでＵＩ構成データを再生する際に利用され、プレーヤーは再生ヘッ

ドの位置とこの時間情報テーブルを利用して、表示するオブジェクトを決定し画面に出力する。また、ラベルは時刻と文字列を対応させた情報により定義され、イベントハンドラは、ラベルとイベント条件、イベント発生時の処理内容で定義される。本例では、時刻の代わりにラベルを指定することにより、再生ヘッドをアニメーション再生位置の所望の位置（例えば、あるアニメーションの先頭など）に移動させるようにしている。

【００３６】

このように、アニメーションと、ラベルによる状態表現と、イベントハンドラを分離することによって、アニメーション（デザイン）の開発とＧＵＩのアプリケーションプログラムの開発を平行して行うことができる。また、イベントハンドラを設定するだけでアニメーションを簡単につなぎ合わせることができるので、ＧＵＩの開発効率を向上させることができる。また、アニメーションの時刻にラベルを付けることで、入力アニメーションをデザインや挙動の異なる別のアニメーションに容易に差し替えることが可能となる。なお、イベントハンドラの設定において、ラベルでなく時刻を直接指定することもできるが、その場合、ラベルを用いることによる効果は得られない。

【００３７】

図２は、アニメーションを基にしたＧＵＩを実行するシステムの構成図を示す。この図２で示されるシステムは、図１のオーサリングシステムで作成されたＵＩ再生用フォーマットのＵＩ構成データを読み込み、ＵＩとして再生を実行するデータ再生システムであって、ディスプレイ装置を搭載する機器上で動作するアプリケーションソフトウェアに組み込まれることで製品のＵＩを実現するものである。

【００３８】

本例のデータ再生システムは、図１のオーサリングシステムにより作成されたＵＩ構成データの再生用データを読み込む再生用データ読込部１１と、再生用データに含まれるイベントハンドラ及びラベルを基に構成されるイベント処理内容を記述したイベント処理ルールを管理するイベント処理ルール保持部１２と、読み込んだアニメーションを管理するアニメーション保持部（画像データ保持部）１３と、イベント検出手段（図示略）により検出されたイベントを通知するイベント検出部１４と、発生したイベントに応じた処理を行うイベント処理部１５とを備える。

【００３９】

さらに、データ再生システムは、イベント処理部１５のイベント処理内容に従いアニメーションの再生位置を変更するアニメーション再生位置変更部（画像データ再生位置変更部）１６、及びボタンやキャラクタといったアニメーション中に存在するパーツの色（テクスチャ含む）や模様、大きさ、位置といった属性を変更するアニメーション変更部（画像データ変更部）１７、読み込んだ複数のアニメーションを一つのアニメーションとして扱うアニメーション合成部（画像データ合成部）１８、アニメーション合成部の作成したアニメーションを再生しディスプレイ装置に表示させるアニメーション再生部（画像データ再生部）１９とを備えてなる。

【００４０】

本例のイベントとして、ユーザのマウスクリック、ドラッグ＆ドロップ、キー入力などの操作によるユーザイベント、又は、例えばタイマーによる再生アニメーション切り替えなどデータ再生システムが引き起こすシステムイベント等、種々のイベントが考えられ、後述するプレーヤーが受け取ることが可能なものはイベントとして利用することができる。例えば、あるプログラムが、バッテリーの残量が少なくなったことを検出し、プレーヤーに向けて「バッテリー残量が少ない」旨を表示するというイベントを発行し、ＧＵＩ上に警告表示をするなども可能である。

【００４１】

上述のイベント処理ルール保持部１２におけるイベント処理ルールには、処理を開始する条件としてアニメーション再生位置の範囲と、トリガーとなるイベントが記述されている。また処理内容として、イベント処理ルールに基づきアニメーション再生位置変更部１６及びアニメーション変更部１７への指示、あるいは外部プログラムの実行などの処理が

記述されている。アニメーション変更部１７にて、色や模様、大きさ、位置といったアニメーションの属性を再生時に変更することで、オーサリングシステム側で作成すべきアニメーション及びデータ再生システム側で用意すべきアニメーションを、それぞれ削減することができる。

【００４２】

例えば、テレビジョン放送番組表のようなＵＩを作成する場合、将来の番組内容は決まっていないため適宜番組内容（画像）を入れ替える必要がある。また、入力装置としてのマウスについてくるようなキャラクタ表示を実現するような場合には、キャラクタの座標を変更する必要がある。このように、機器のＵＩ実行時にしか決まらない属性を変更するための機能を持つシステムが図２のデータ再生システムである。このような操作は、上述のようにアニメーション変更部１７がサポートすることで実現するようにしてもよいし、外部プログラムとして記述しこれを実行することでも実現することができる。

【００４３】

本例のデータ再生システムをプレーヤーに組み込み、デジタルカメラ、テレビジョン受像機、ＤＶＤ（Digital Versatile Disc）再生機器など、多くの機器上で動作させる。この例でプレーヤーの定義は、オーサリングシステムからのＵＩ再生用フォーマットのデジタル画像データが再生できることであり、どのように実装されているかは問題としない。例えば、デジタルカメラで動作するプレーヤーとテレビジョン受像機で動作するプレーヤーのように、機能は同じであるが、プログラムは異なってもよい。オーサリングツールよりこれらのプレーヤーが再生可能なフォーマットでＵＩ構成データが出力される。例えば、デジタルカメラとテレビジョン受像機では、ボタン等のインターフェース（入力装置）が異なるが、設定ファイル（機器間の違いを埋めるためのイベント対応表など）を変更することで同一イベントとして対応が可能になるなどの整合性を保つ作業をプレーヤー側で行う。そのようにすることにより、オーサリングツールで作成されたＵＩはプレーヤーが存在する環境であれば、どの機器上でも同様に動作させることができる。

【００４４】

なお、図２のデータ再生システムをプレーヤーオブジェクトとして図１のオーサリングシステムに組み込み、同一の機器上で作成中あるいは作成したＧＵＩのＵＩ構成データを実行できるようにしてもよいことは勿論である。この場合、ＵＩとしての動作を確認しながらオーサリング作業を進める、またはＧＵＩの完成と同時に全体を通した動作を確認することができるので、オーサリングの作業効率が向上する。

【００４５】

図３は、アニメーション合成部１８（図２）が作成する合成アニメーションの構成図を示すものである。合成アニメーションは１つ以上のアニメーションから構成される。

【００４６】

図中、２４は合成アニメーション内での時間の流れを表しており、ラベル設定における時刻の基準となる時間であるから、例えばアニメーションが繰り返し再生される場合などでは現実の経過時間と相違することを意味する。２１，２２，２３は入力されたアニメーションＸ、Ｙ、Ｚの表示時間（表示範囲）を表している。各表示時間の左端はそれぞれのアニメーションの先頭を示しており、合成アニメーション内での時刻がそれぞれのアニメーションにおける時刻とずれることを許容している。さらに、２５は合成アニメーションの再生位置を表しており、それぞれのアニメーションはこの再生位置で示された時刻のアニメーションが再生される。

【００４７】

この図３の例では、現在の再生位置を示す線２５がアニメーションＸ及びアニメーションＺにかかっているので、この再生位置において双方のアニメーションが同時に表示されることを表している。複数のアニメーションが同時に表示される場合、アニメーションが３Ｄ（３次元）グラフィックスであれば奥行きを利用して混ぜたり、あるいは、混ぜずにディスプレイ装置に重ねて表示したりでき、表現の自由度が大きい。表示時間外では、ディスプレイ装置にアニメーションは表示されない。アニメーションの属性として、開始時

間、開始オフセット、終了オフセットを指定できれば、アニメーションを任意の再生位置に任意の長さで表示することができる。

【００４８】

図４は、本例のオーサリングシステム（図１）を用いたＧＵＩの実現手法を示すフローチャートである。ステップＳ１は、デザイナーがＧＵＩのデザイン及び挙動を考え、ＧＵＩを構成するアニメーションを既存のアニメーション作成ツールを用いて作成することを表現している。ステップＳ２は、作成されたアニメーションを図１のオーサリングシステムに入力し、イベント処理ルールを作成することを表現している。ステップ３は、作成されたイベント処理ルールに従い、アニメーション再生部８によってアニメーションを再生することでＧＵＩを実現することを表現している。このステップＳ３における処理について詳細なフローチャートを図５に示す。

【００４９】

図５は、図１のオーサリングシステムにより作成されたアニメーション付きＧＵＩの実行処理を示すフローチャートである。

【００５０】

まず、例えばユーザ操作によってＧＵＩの実行指令が下されることにより、再生用データ読込部１１から読み込んだアニメーションの再生を開始する。再生中は所定のアニメーションの再生処理を行いながら、イベント処理ルールに設定されているイベントが発生するまで待機する（ステップＳ１１）。

【００５１】

次に、イベント検出部１３にてイベントの発生が検出された場合、検出されたイベントの内容がイベント処理ルールに適合するか否かを判断する（ステップＳ１２）。イベント処理ルール内に適合するイベントが存在する場合には、イベント処理ルールに記述されている当該イベントに対応した処理を行う（ステップＳ１３）。ここで、イベント処理ルールの処理内容が終了処理であるか否かを判断し（ステップＳ１４）、終了処理である場合にはイベント処理内容に従いＧＵＩの実行を終了する。

【００５２】

ステップＳ１４において、イベント処理ルールの処理内容が終了処理でない場合、又はステップＳ１２において検出されたイベントがイベント処理ルールに適合しない場合、アニメーション再生時刻でアニメーションを再生する（ステップＳ１５）。その後、ステップＳ１１に移行し、何らかのイベントが発生するまで上述アニメーションを再生しながら待機する。そして、イベントが発生したらそのイベント内容に応じてステップＳ１２～ステップＳ１５の処理を行う。以降、終了処理が行われるまで上述処理を繰り返す。なお、検出されたイベントがＧＵＩの終了ボタン押圧などの終了イベントであった場合には、イベントが検出された時点でイベント処理ルールに適合するかどうかの判断を待たずにＧＵＩの実行を終了するようにしてもよい。

【００５３】

図６は、図４で説明されるＧＵＩの実現手法を実行した場合の具体例を示したものである。図６ＡはＧＵＩを構成するアニメーションを表す。図中、３３はアニメーション内時間であり、１本のアニメーションを通しで再生したときの開始から終了までの経過時間を示している。この図６Ａにおいて静止画像のような描写となっているが、アニメーションの動きを表現しておりアニメーション内時間上で連続的に立体が回転している様子を示している。この例のアニメーションは、青いキューブ３１が回転するアニメーションと赤いキューブ３１が回転するアニメーションから構成され、例えば青いキューブ３１が１回転した後、青いキューブ３１が赤いシリンダ３２と入れ替わり、赤いシリンダ３２が１回転するというものである。なお、この例では、オーサリングシステムに対する入力は時間的に連続する１つのアニメーションとしているが、この例に限らず、複数であってもよい。

【００５４】

図６Ｂは、入力されたアニメーションの再生時刻に対して設定したラベルの例を示す。ここでは、アニメーション内時間における時刻０ｍｓに対応するラベル名をスタートとし

、同２０００ｍｓに対応するラベル名をＡ、同３９００ｍｓに対応するラベル名をエンドとしている。

【００５５】

図６Ｃは、入力されたアニメーションに対して設定したイベント処理ルールの例である。イベント処理対象時間帯、トリガーイベント（対象イベント）、処理内容をそれぞれ設定し、表中のルール１，２，３のようにイベント処理ルールとして追加する。その処理内容として例えばルール２では、図２のＧＵＩを再生するシステムが組み込まれたアプリケーションからコールバック関数を呼び、外部プログラムのサウンド再生システム（図示略）にアクセスしてサウンド再生を行うように記述されている。このように、処理内容としてアニメーションの再生位置移動だけではなく、指定したサウンドの再生など所望の処理が実行可能である。したがって、データ再生システム内で処理するか、外部プログラムを呼び出して処理するか、アニメーションの再生位置によりイベント処理方法を変更することで、より柔軟なＧＵＩを実現することができる。

【００５６】

図６Ｄは、イベント処理ルール（図６Ｃ）を設定した後に、アニメーション再生部１９（図２）でアニメーションを再生した場合の挙動を示すものである。以下の図６Ｄの説明においては図５のフローチャートとの対応を記載している。

【００５７】

まずスタート位置からキューブ３１が回転するアニメーションの再生を開始し、アニメーションを再生しながらイベント発生まで待機する（ステップＳ１１に相当）。再生時刻が（Ａ－１）に達するとトリガーイベントであることを検出し、終了イベントでないことを確認した後（ステップＳ１２に相当）、イベント内容がイベント処理ルールのルール１に適合することを確認し（ステップＳ１３）、ルール１に従いアニメーション再生位置をスタートに戻す（矢印３５）（ステップＳ１４に相当）。そして、キューブ３１が回転するアニメーションを再度スタートから再生し（ステップＳ１５に相当）、他のトリガーイベントが発生するまでキューブ３１が回転するアニメーションの再生を行う（ステップＳ１１に相当）。

【００５８】

ここで、キューブ３１が回転するアニメーションの再生中にユーザ操作によりボタンが押されると、ルール２に従いアニメーション再生位置をＡへ移動する（矢印３６）とともにサウンドの再生を行う（ステップＳ１２～Ｓ１４に相当）。

【００５９】

そして、ルール２により再生位置がＡに移動した後は、キューブ３１からシリンダ３２が回転するアニメーションへと切り替えられて再生が行われる（ステップＳ１５、ステップＳ１１に相当）。再生時刻がエンドに達するとルール３のトリガーイベントであることを検出し（ステップＳ１２、ステップＳ１３に相当）、ルール３に従いアニメーション再生位置がＡに戻される（矢印３７）（ステップＳ１４に相当）。以降、何らかのイベントが発生するまでシリンダ３２が回転するアニメーションの再生が繰り返し行われる（ステップＳ１５、ステップＳ１１に相当）。

【００６０】

次に、図７及び図８を参照して本例のオーサリング方法によりイベント処理ルールの設定とアニメーションが分離されることについて説明する。

【００６１】

図７は、イベント処理ルールを図６Ｃと同じまま入力アニメーションを再生時間に関して長さの変わらない別のアニメーションに変更する例を説明するものである。

【００６２】

図７Ａは、図６Ａ～Ｄを用いて解説した手順によって作成されたＧＵＩに対し、赤いシリンダ３２が回転するアニメーションを、緑の円錐４２が回転するアニメーションに入れ替えた様子を示している。

【００６３】

オーサリングシステム（図１）のＵＩ構成データ保持部４に円錐４２の回転するアニメーションがない場合には、既存の円錐４２が回転するアニメーションをオーサリングシステムのアニメーション読込部１に読み込み、ＵＩ構成データ保持部４に登録する。そして、円錐４２が回転するアニメーションを再生用データとして再生用データ出力部５から図２のデータ再生システムに出力する。データ再生システムでは、円錐４２が回転するアニメーションを再生用データ読込部１１に読み込んでアニメーション保持部１１に記録されているシリンダ３２が回転するアニメーションと入れ替える。

【００６４】

図７Ｂは、入力アニメーションを再生時間の長さの変わらない円錐４２が回転するアニメーションに入れ替え、アニメーション再生部１９（図２）でアニメーションを再生した場合の挙動を示す。ここで、図６Ｄと異なる点は、キューブ３２が回転するアニメーションに代えて、円錐４２が回転するアニメーションが再生されるということだけである。イベント処理部１４（図２）は、イベント処理ルールに基づきシリンダ３２に代えて円錐４２が回転するアニメーションについてイベント処理を行うことができる。ラベル及びイベント処理ルールについて何ら変更の必要がない。

【００６５】

このように、入力アニメーションを長さの変わらない別のアニメーションに変更する場合には、入れ替え対象のアニメーションのみを入れ替え、ラベル及びイベント処理ルールを変更することなく、ＧＵＩとして再生することが可能である。

【００６６】

図８は、イベント処理ルールを図６Ｃと同じままとし、入力アニメーションを再生時間の長さの異なる別のアニメーションに変更する例を説明するものである。

【００６７】

図８Ａでは、図６Ａ～Ｄを用いて解説した手順によって作成されたＧＵＩに対し、キューブ３１及びシリンダ３２が回転するアニメーションを、それぞれ長さの異なるキューブ３１及びシリンダ３２の回転するアニメーションに入れ替えた様子を示している。

【００６８】

この場合、図８Ｂに示すラベルと時刻の対応を変更することにより、イベント処理ルールに変更を加えることなくＧＵＩとして再生することが可能となる。例えば、図６Ｂのラベル設定に対し、図８Ｂに示すようにアニメーション内時間における時刻１０００ｍｓに対応するラベル名をＡ、同１９９９ｍｓに対応するラベル名をエンドと変更する。即ち、図６Ａのアニメーションについて再生時間を短縮し、本例はキューブ３１が回転するアニメーションの開始から（１０００－１）ｍｓまで、及びシリンダ３２が回転するアニメーションの１０００ｍｓから１９９９ｍｓまでをＧＵＩの再生に利用する。

【００６９】

図８Ｃは、図８Ａの入力アニメーションをそれぞれ長さの異なるアニメーションに入れ替えた後に、アニメーション再生部１８（図２）でアニメーションを再生した場合の挙動を示す。まずキューブ３１が回転するアニメーションの再生をスタートから開始する。再生時刻が（Ａ－１）に達するとトリガーイベントであることを検出し、イベント処理ルールのルール１に従ってアニメーション再生位置をスタートに戻す（矢印３５ａ）。以降、他のトリガーイベントが発生するまでキューブ３１が回転するアニメーションの再生を繰り返し行う。

【００７０】

ここで、キューブ３１が回転するアニメーションの再生中にユーザ操作によりボタンが押されると、ルール２に従いアニメーション再生位置をＡへ移動する（矢印３６ａ）とともにサウンドの再生を行う。

【００７１】

そして、ルール２により再生位置がＡに移動した後は、キューブ３１からシリンダ３２が回転するアニメーションの再生へと切り替えられて再生が行われる。再生時刻がエンドに達するとルール３のトリガーイベントであることを検出し、ルール３に従いアニメーシ

ョン再生位置がＡに戻される（矢印３７ａ）。以降、何らかのイベントが発生するまでシリンダ３２が回転するアニメーションの再生が繰り返し行われる。

【００７２】

このように、入力アニメーションを再生時間の長さの異なる別のアニメーションに変更する場合には、ラベルの定義時刻を変更するだけでイベントハンドラの変更は必要ない。

【００７３】

なお、外部のアニメーション作成ツールで作成された長さの異なるアニメーションを図１のオーサリングシステムに読み込んで利用することも勿論可能である。

【００７４】

以上述べたような構成によれば、ＧＵＩ開発における大部分の作業を、既存のアニメーション作成ツールを用いたアニメーション作成により実現できるので、簡易にアニメーション付きＧＵＩを作成することができる。

【００７５】

また、デザイナーが既存のアニメーション作成ツールを用いて作成したアニメーションが利用できるので、イメージ通りのアニメーションを実現できるようになり、プログラマとの意思疎通の失敗によるＧＵＩ開発期間の長期化等のリスクが発生しない。

【００７６】

また、図１のシステム構成図に示すように、ＧＵＩ開発のために必要な機能が単純なため、オーサリングツールを簡易に実現することができる。

【００７７】

また、ラベル付けによりアニメーション中の意味のある部分を状態として切り出し、状態（ラベル）をベースにイベント処理ルールが設定されていることによって、アニメーションとイベント処理ルールを分離・独立して取り扱うことが可能となり、入力アニメーションをデザインや挙動、長さなどが異なる別のアニメーションに容易に差し替えることができる。さらに、例えば処理性能や利用可能な色数といった表現能力の異なる機器に対してＧＵＩの移植作業を行う場合にも、イベント処理ルールを変更する必要がないので、効率的に移植作業を進めることができるとともに、移植性が向上する。

【００７８】

なお、上述の実施の形態の例においては、アニメーション読込部１に読み込むデジタル画像データを３Ｄグラフィックスのアニメーションとして説明したが、２Ｄ（２次元）グラフィックスのアニメーションであってもよい。また、読み込むデジタル画像データが静止画の場合にも、イベント処理対象時間帯、対象イベント、処理内容を設定することにより、所定のイベント処理を行わせることができ、上述例と同様に既存のデジタル画像データを利用してＧＵＩを構築することができる。勿論、アニメーションと静止画を組み合わせてＧＵＩを構築するようにしてもよい。

【００７９】

また、上述例において、アニメーションの再生位置等を表す時間の単位としてｍｓを使用したが、例えば３秒間を１０区間に分けて１０フレームから構成し、このフレームを利用して再生位置を表すようにしてもよい。

【００８０】

また、本発明に一般的なオーサリングツールの機能の一つであるレイヤーの概念を取り入れ、図３のように複数のアニメーションがある場合にアニメーションのグループ化、親子関係といった構造化を行うことにより、ディスプレイ装置上のアニメーション表示をさらに多様化することができる。

【００８１】

また、本発明は上述した実施の形態の例に限られるものではなく、本発明の要旨を逸脱することなくその他種々の構成を取り得ることは勿論である。

【図面の簡単な説明】

【００８２】

【図１】本発明の一実施の形態の例のオーサリングシステムの構成図である。

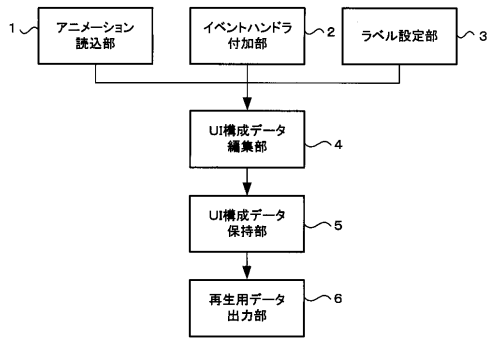
【図２】本発明の一実施の形態の例のデータ再生システムの構成図である。
【図３】合成アニメーションの構成図である。
【図４】ＧＵＩ実現方法の説明に用いるフローチャートである。
【図５】本例のオーサリングシステムにより作成されたＧＵＩの実行フローチャートである。
【図６】ＧＵＩの説明に用いる線図である。
【図７】図６のアニメーションを長さの同じ別のアニメーションに変更した場合についての説明図である。
【図８】図６のアニメーションを長さの異なるアニメーションに変更した場合についての説明図である。

【符号の説明】

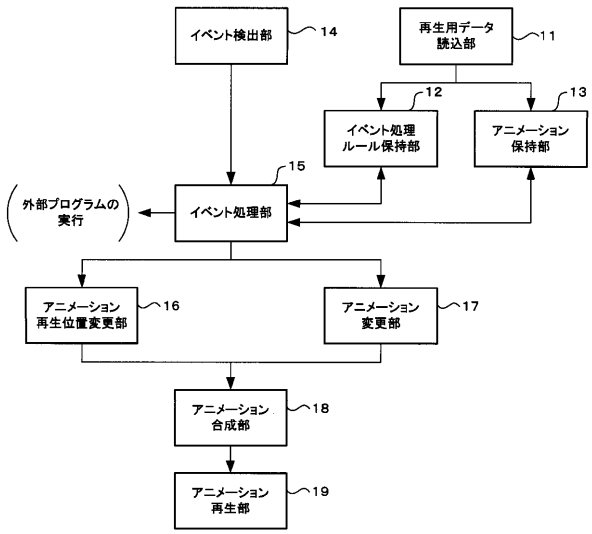
【００８３】

１…アニメーション読込部（画像データ読込部）、２…イベントハンドラ付加部、３…ラベル設定部、４…ＵＩ構成データ編集部、５…ＵＩ構成データ保持部、６…再生用データ出力部、１１…再生用データ読込部、１２…イベント処理ルール保持部、１３…アニメーション保持部（画像データ保持部）、１４…イベント検出部、１５…イベント処理部、１６…アニメーション再生位置変更部（画像データ再生位置変更部）、１７…アニメーション変更部（画像データ変更部）、１８…アニメーション合成部（画像データ合成部）、１９…アニメーション再生部（画像データ再生部）、２１，２２，２３…アニメーション表示時間、２４，３３…アニメーション内時間、２５…再生位置、３１…キューブ、３２…シリンダ、３５，３６，３７，３５ａ，３６ａ，３７ａ…再生位置の移動、４２…円錐

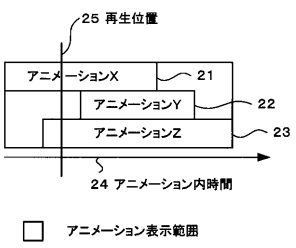
【図１】



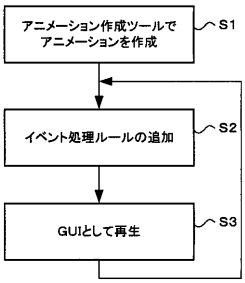
【図２】



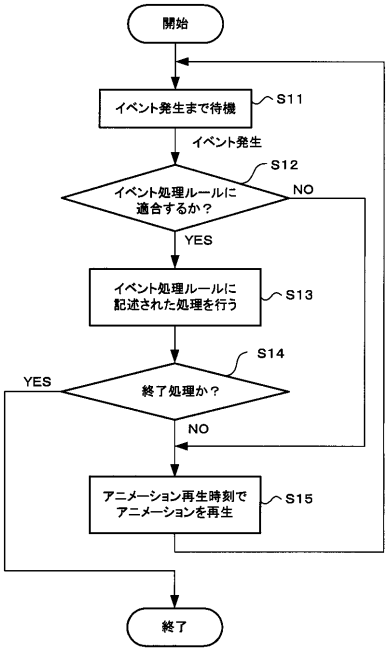
【図３】



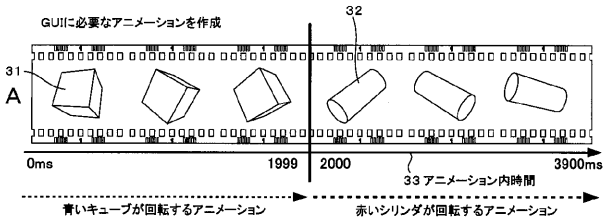
【図４】



【図５】



【図６】



ラベルを設定

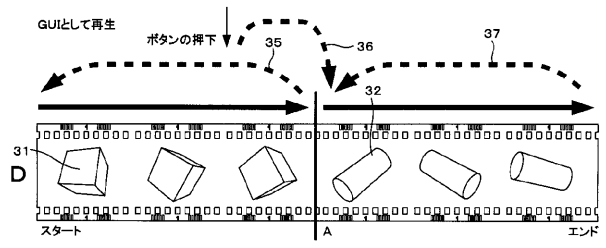
B

ラベル名	時刻(ms)
スタート	0
A	2000
エンド	3900

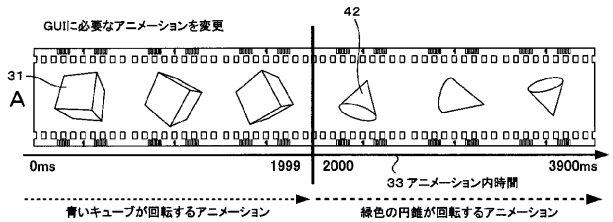
イベント処理ルールを追加

C

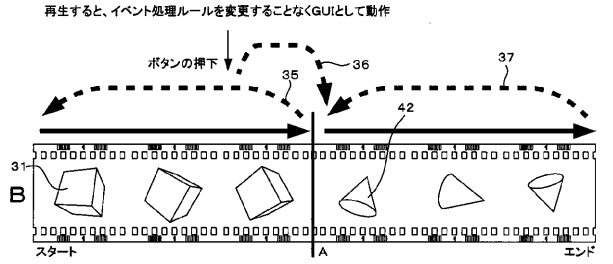
ルール	対象時間帯	トリガーイベント	処理内容
ルール1	A−1	再生時刻＝A−1	アニメーション再生位置をスタートへ
ルール2	スタートからA−1	ボタンが押された	アニメーション再生位置をAへ サウンドを再生
ルール3	エンド	再生時刻＝エンド	アニメーション再生位置をAへ



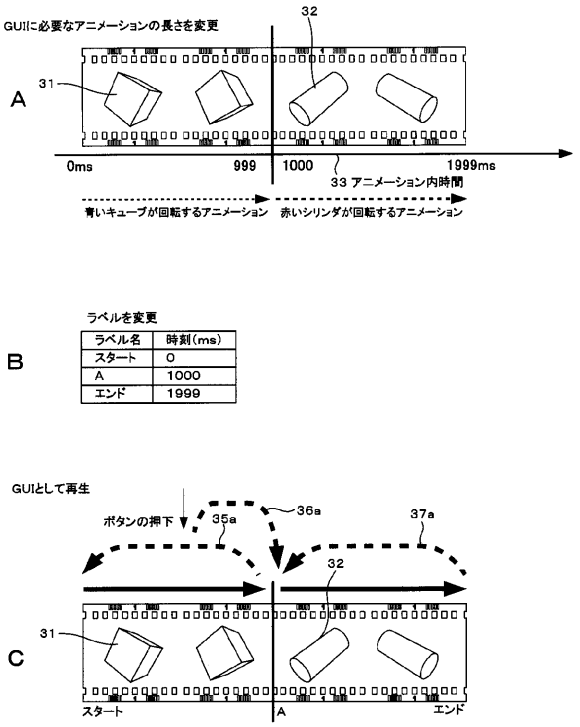
【図７】



再生すると、イベント処理ルールを変更することなくGUIとして動作



【図８】


GUIに必要なアニメーションの長さを変更
A
31
32
0ms
999
1000
1999ms
33 アニメーション内時間
青いキューブが回転するアニメーション
赤いシリンダが回転するアニメーション
ラベルを変更
B
ラベル名
時刻(ms)
スタート
0
A
1000
エンド
1999
GUIとして再生
C
ボタンの押下
35a
36a
37a
32
31
スタート
A
エンド

フロントページの続き

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

Ｆターム(参考) 5B069 AA01 DD01 DD16 DD17
5B076 DC05 DD06 DF08
5C053 FA14 GB06 GB21 JA16
5E501 AA01 AC16 BA05 DA06 FA14 FA15 FA41