

(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2005-301422
(P2005-301422A)

(43) 公開日 平成17年10月27日(2005.10.27)

(51) Int.CI.⁷**G06T 11/60**
G06F 17/21

F 1

G06T 11/60
G06F 17/21
G06F 17/21

テーマコード(参考)

5B009
5B050

586E

審査請求 未請求 請求項の数 7 O L (全 16 頁)

(21) 出願番号
(22) 出願日特願2004-113213 (P2004-113213)
平成16年4月7日 (2004.4.7)

(71) 出願人 000001007
キヤノン株式会社
東京都大田区下丸子3丁目30番2号

(74) 代理人 100076428
弁理士 大塚 康徳

(74) 代理人 100112508
弁理士 高柳 司郎

(74) 代理人 100115071
弁理士 大塚 康弘

(74) 代理人 100116894
弁理士 木村 秀二

(72) 発明者 鶴田 裕文
東京都大田区下丸子3丁目30番2号 キ
ヤノン株式会社内

最終頁に続く

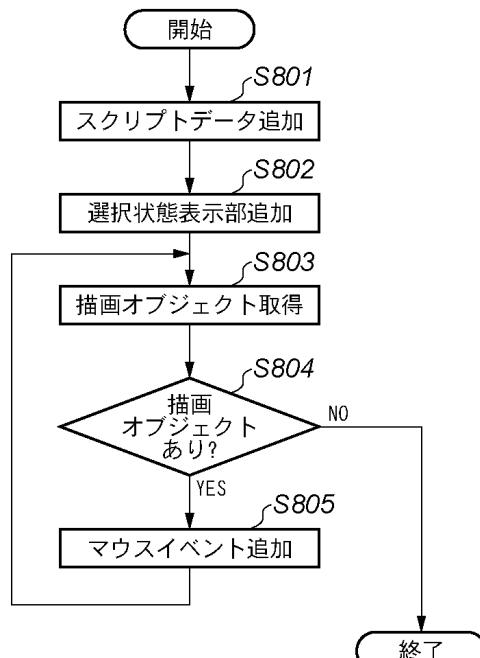
(54) 【発明の名称】文書データ処理装置および方法

(57) 【要約】

【課題】 いったん編集スクリプトが削除されたことより編集が不可となった文書データを、再編集を可能に状態に回復させること。

【解決手段】 編集機能を実現するための編集スクリプトを含む文書データに基づいて画像を表示し、前記編集スクリプトに基づいて、表示された画像に対するユーザの操作入力に応じた前記文書データの編集を行う。編集後の文書データの保存の指示が入力された場合、前記文書データから前記編集スクリプトを削除し、その編集スクリプトが削除された文書データを所定の記憶媒体に保存する。その後、前記保存された文書データの再編集の指示が入力された場合、その保存された文書データに、所定の編集スクリプトを挿入し、これにより文書データを編集可能な状態にする。

【選択図】 図8



【特許請求の範囲】**【請求項 1】**

編集機能を実現するための編集スクリプトを含む文書データに基づいて画像を表示する手段と、

前記編集スクリプトに基づいて、表示された前記画像に対するユーザの操作入力に応じた前記文書データの編集を行う手段と、

編集後の前記文書データを保存すべき旨のユーザからの指示を受け付ける手段と、

編集後の前記文書データを保存すべき旨の指示を受け付けた場合に、当該文書データから前記編集スクリプトを削除し、その編集スクリプトが削除された文書データの保存を行う手段と、

前記保存された文書データを再編集する旨のユーザからの指示を受け付ける手段と、

前記保存された文書データを再編集する旨の指示を受け付けた場合に、当該文書データに、編集機能を実現するための編集スクリプトを挿入する手段と、

を有することを特徴とする文書データ処理装置。

【請求項 2】

前記文書データは XML (eXtensible Markup Language) 文書であることを特徴とする請求項 1 に記載の文書データ処理装置。

【請求項 3】

前記 XML 文書は、は SVG (Scalable Vector Graphics) で記述されたものであることを特徴とする請求項 2 に記載の文書データ処理装置。

【請求項 4】

編集機能を実現するための編集スクリプトを含む文書データに基づいて画像を表示するステップと、

前記編集スクリプトに基づいて、表示された前記画像に対するユーザの操作入力に応じた前記文書データの編集を行うステップと、

編集後の前記文書データを保存すべき旨のユーザからの指示を受け付けるステップと、

編集後の前記文書データを保存すべき旨の指示を受け付けた場合に、当該文書データから前記編集スクリプトを削除し、その編集スクリプトが削除された文書データを所定の記憶媒体に保存するステップと、

前記保存された文書データを再編集する旨のユーザからの指示を受け付けるステップと、

前記保存された文書データを再編集する旨の指示を受け付けた場合に、当該文書データに、編集機能を実現するための編集スクリプトを挿入するステップと、

を有することを特徴とする文書データ処理方法。

【請求項 5】

コンピュータにより実行可能なプログラムであって、

編集機能を実現するための編集スクリプトを含む文書データに基づいて画像を表示するためのコードと、

前記編集スクリプトに基づいて、表示された前記画像に対するユーザの操作入力に応じた前記文書データの編集を行うためのコードと、

編集後の前記文書データを保存すべき旨のユーザからの指示を受け付けるためのコードと、

編集後の前記文書データを保存すべき旨の指示を受け付けた場合に、当該文書データから前記編集スクリプトを削除し、その編集スクリプトが削除された文書データを所定の記憶媒体に保存するためのコードと、

前記保存された文書データを再編集する旨のユーザからの指示を受け付けるためのコードと、

前記保存された文書データを再編集する旨の指示を受け付けた場合に、当該文書データに、編集機能を実現するための編集スクリプトを挿入するためのコードと、

を含むことを特徴とするプログラム。

10

20

30

40

50

【請求項 6】

請求項 5 に記載のプログラムを格納したコンピュータ読みより可能な記憶媒体。

【請求項 7】

編集機能を実現するための編集スクリプトを含む文書データを記憶するサーバと、ネットワークを介して前記サーバに接続されるクライアントとを含む文書データ処理システムであって、

前記クライアントは、

前記文書データを前記サーバから取得する手段と、

取得した前記文書データに基づいて画像を表示する手段と、

前記文書データにおける前記編集スクリプトに基づいて、表示された前記画像に対する操作入力に応じた前記文書データの編集を行う手段と、

編集後の前記文書データを保存すべき旨のユーザからの指示を受け付ける手段と、
編集後の前記文書データを保存すべき旨の指示を受け付けた場合に、当該文書データから前記編集スクリプトを削除し、その編集スクリプトが削除された文書データを保存する手段と、

前記保存された文書データを再編集する旨のユーザからの指示を受け付ける手段と、を有し、

前記サーバは、

前記保存された文書データを再編集する旨の指示が前記クライアントにおいて受け付けられた場合に、当該文書データに、編集機能を実現するための編集スクリプトを挿入する手段を有する

ことを特徴とする文書データ処理システム。

【発明の詳細な説明】**【技術分野】****【0001】**

本発明は、編集可能な文書データの管理技術に関する。

【背景技術】**【0002】**

構造化文書（特にSVGのようなグラフィックス記述言語）は一般に、その多彩な表現力を利用した閲覧ページの実現を目的として利用される。従来は、このような閲覧ページは文字どおり、もっぱら閲覧のみを目的としたものであり、閲覧する者が編集を行なうこととは考慮されていなかった。

【0003】

しかし近年、インターネットの普及やパーソナルコンピュータの性能向上に伴い、一般に広く知られる構造化文書を用いて、文書の編集を行う方法が提案されている（例えば、特許文献1を参照。）。文書の編集は例えば、構造化文書に編集スクリプトを埋め込むことによって実現される。

【0004】

その一方で、編集スクリプトを構造化文書に埋め込むことは構造化文書の文書サイズを増大させることになる。そのため、編集が完了し、閲覧のみを目的とした文書の保存、転送、配布に際しては、通信負荷や記憶に必要な容量が増大するという問題があった。また、文書中に編集スクリプトが残ることによって、作成者の意図しない文書の改変がなされる可能性があるという問題点もある。

【0005】

このような問題に対処すべく、本発明者は、編集後に文書データの保存の指示が入力された場合、その文書データから編集スクリプトを削除したうえでその文書データを所定の記憶媒体に保存することを別途、提案した。

【0006】**【特許文献1】特開2001-290811号公報****【発明の開示】**

10

20

30

40

50

【発明が解決しようとする課題】**【0007】**

上記のように文書データから編集スクリプトを削除したうえでその文書データを保存するようすれば、たしかに記憶サイズを削減でき、なおかつ、作成者の意図しない改変を防止することができる。しかしその一方で、例えば作成者の意図に従って文書を元の状態に回復させ、再度編集を行いたい場合もある。しかしながら、従来は、いったん削除した編集スクリプトを回復させることはできなかった。

【0008】

本発明は上記課題に鑑み、いったん編集スクリプトが削除されたことより編集が不可となつた文書データを、再編集を可能に状態に回復させることを目的とする。

10

【課題を解決するための手段】**【0009】**

本発明の一側面に係る文書データ作成装置は例えば、編集機能を実現するための編集スクリプトを含む文書データに基づいて画像を表示する手段と、前記編集スクリプトに基づいて、表示された前記画像に対するユーザの操作入力に応じた前記文書データの編集を行う手段と、編集後の前記文書データを保存すべき旨のユーザからの指示を受け付けた場合に、編集後の前記文書データから前記編集スクリプトを削除し、その編集スクリプトが削除された文書データを保存する手段と、前記保存された文書データを再編集する旨のユーザからの指示を受け付ける手段と、前記保存された文書データを再編集する旨の指示を受け付けた場合に、当該文書データに、編集機能を実現するための編集スクリプトを挿入する手段とを有することを特徴とする。

20

【発明の効果】**【0010】**

本発明によれば、いったん編集スクリプトが削除されたことより編集が不可となつた文書データを、再編集を可能に状態に回復させることができる。

【発明を実施するための最良の形態】**【0011】**

以下、図面を参照して本発明の好適な実施形態について詳細に説明する。

【0012】

図1Aは、本実施形態における文書データ処理システムのハードウェア構成を示す図である。

30

【0013】

この文書データ処理システムは、図示の如く、クライアント111とサーバ101とを含み、両者がネットワーク（LAN、インターネットなど）120を介して接続された構成である。同図には参考番号111で示されるクライアントが1つだけ存在しているが、クライアントの数は問題ではなく、複数存在していてもよい。クライアント111およびサーバ101はそれぞれ、例えば一般的のパーソナルコンピュータやワークステーションで実現が可能である。もっとも、クライアント111はパーソナルコンピュータではなく、いわゆるPDAや携帯電話機などの端末によっても実現可能である。

40

【0014】

サーバ101は、このサーバ101の構成要素全体の制御を統括するCPU1、固定的なプログラムやデータを記憶するROM2、主記憶装置として機能するRAM3をはじめ、以下の構成を有する。HDD4は外部記憶装置としてのハードディスク装置である。VRAM5は表示しようとするイメージデータを展開するメモリであり、ここにイメージデータ等を展開することでCRT6に表示させることができる。なお、6には表示装置の一例としてCRTを使用するが、CRTではなく液晶モニタ等を使用することももちろん可能である。7および8はそれぞれ、各種設定を行うためのキーボードおよびマウスである。また、9はインターネット12に接続するためのインターフェースである。

【0015】

クライアント111もサーバ101と同様のハードウェア構成であり、CPU11～I

50

/ F 1 9 はそれぞれ、サーバ 1 0 1 の C P U 1 ~ I / F 9 に対応している。

【 0 0 1 6 】

サーバ 1 0 1 およびクライアント 1 1 1 のハードウェア構成は概ね上記のとおりであるが、それぞれの役割を果たすために、サーバ 1 0 1 の H D D 4 とクライアント 1 1 1 の H D D 1 4 とでは、インストールされる内容が相違する。サーバ 1 0 1 の H D D 4 には、O S や 1 又は 2 以上の文書データが記憶されるほか、W E B サーバ機能を実現するための W E B サーバプログラム、H T M L 文書のひな型となるテンプレートデータ、後述する編集処理によって所定コードが削除された文書を復元するための編集コード復元プログラムなどがインストールされている。一方、クライアント 1 1 1 の H D D 1 4 には、O S および W E B ブラウザプログラム（以下、単に「W E B ブラウザ」という。）のほか、W E B ブラウザのアドオンソフトウェアとしての表示プラグイン、追加要素選択プラグイン、編集プラグインがインストールされている。

【 0 0 1 7 】

図 1 B は、本実施形態における文書データ処理システムの機能構成を示す図である。

【 0 0 1 8 】

1 0 2 は文書供給部であり、クライアントからネットワーク 1 2 0 を介して H T T P (Hypertext Transfer Protocol) リクエストを受信し、H D D 4 中に保管されるテンプレートデータ、文書データ等を本文 (body) とする H T T P レスポンスを、ネットワーク 1 2 0 を介してクライアントに返信する。これは、上記した W E B サーバプログラムによって実現される。

【 0 0 1 9 】

この文書供給部 1 0 2 は編集画面作成部 1 0 3 および編集コード復元部 1 0 4 を含む。編集画面作成部 1 0 3 は、編集対象の文書のリクエストを文書供給部 1 0 2 を介して受け付け、その編集文書を編集する編集画面の H T M L を作成して、文書供給部 1 0 2 を介して送信を行なう。本実施形態においては、上記リクエストに応じてテンプレートデータを H D D 4 より取得して、編集文書として返却する。一方、編集コード復元部 1 0 4 は、文書の再編集のリクエストを文書供給部 1 0 2 を介して受け付け、その再編集対象の文書に編集用の記述を付け加えることで、編集可能な文書に復元して返却する。

【 0 0 2 0 】

1 0 5 は文書データやテンプレートデータを蓄積するデータベースとしての文書蓄積部で、H D D 4 によって実現される。

【 0 0 2 1 】

クライアント 1 1 1 において、1 1 2 は文書閲覧編集部で、サーバ 1 0 1 の文書供給部 1 0 2 より送られてくる H T M L データを解釈し C R T 1 6 への表示処理を行う。この文書閲覧編集部 1 1 2 は W E B ブラウザによって実現される。

【 0 0 2 2 】

文書閲覧編集部 1 1 2 は、表示拡張部 1 1 3 、追加要素選択拡張部 1 1 4 、編集拡張部 1 1 5 を含む。表示拡張部 1 1 3 は、H T M L データが指示するユーザの所望する編集文書を C R T 1 6 に表示可能にするための機能拡張手段である。この拡張機能は表示プラグインにより提供される。

【 0 0 2 3 】

追加要素選択拡張部 1 1 4 は、H T M L データが指示する編集メニューデータをユーザが使用可能をすべく表示ならびに操作可能とするための機能拡張手段である。この拡張機能は追加要素選択プラグインにより提供される。

【 0 0 2 4 】

編集拡張部 1 1 5 は文書の編集を実現するための機能拡張手段であり、その文書に埋め込まれたスクリプト記述を解釈する。この拡張機能は編集プラグインにより提供される。

【 0 0 2 5 】

図 2 は、編集画面作成部 1 0 3 によってテンプレートデータに基づいて作成される編集画面の H T M L ソースの例である。

10

20

30

40

50

【 0 0 2 6 】

201はH T M Lヘッダ部、202はスクリプト記述部である。スクリプト記述部202には、H T M L内に配置されているボタン等の操作インターフェースをトリガとしてコールされる関数が記述されている。図中の例では、後述する保存処理起動記述部213によって起動されるスクリプトが記述されている。

【 0 0 2 7 】

203は外部スクリプトファイル定義部であり、スクリプト記述部202と同様にH T M L内に配置されているボタン等の操作インターフェースをトリガとしてコールされる関数が記述される。図中の例では、後述する再編集処理起動記述部212によって起動されるスクリプトが記された外部データが記述されている。なお、外部データは、本H T M Lを解釈することにより文書閲覧編集部112が自動的に読み込みを行い、その読み込まれたデータは、本H T M L中に記述されたスクリプトと同様に取り扱うことが可能である。
10

【 0 0 2 8 】

204は編集文書表示部で、編集対象となる本H T M Lの外部データとしての編集文書データが記述される。本H T M Lを表示する際には、文書閲覧編集部112が自動的に読み込みを行い、展開、表示する。

【 0 0 2 9 】

編集文書表示部204において、211は編集対象の文書データ名である。本実施形態においては、W3Cにより標準化されているS V G (Scalable Vector Graphics)形式の文書を用いることとする。
20

【 0 0 3 0 】

212は再編集処理起動記述部で、本システムにて編集保存された文書の再編集処理を記述したスクリプトを呼び出す記述がなされている。

【 0 0 3 1 】

213は保存処理起動記述部で、編集対象の文書データ名211が指示する文書を保存する保存処理を記述したスクリプトを呼び出す記述がなされている。

【 0 0 3 2 】

以上説明した図2のH T M Lの表示例については、図5を用いて後述する。

【 0 0 3 3 】

図3は、図2のH T M Lソースにおける編集文書表示部204に記述された文書データ211の記述内容の一例を示す図である。上記のとおりこの文書データ211はS V G文書であり、図1において説明した文書閲覧編集部112にて展開され、表示拡張部113にて解釈されて表示されるものである。
30

【 0 0 3 4 】

図3において、301は編集文書ヘッダ部分である。本実施形態においては、X M L (eXtensible Markup Language)で規定されているヘッダ部を用いている。302は外部スクリプト定義部である。ここで定義されている外部スクリプト中には、本編集文書であるS V Gに配置されているオブジェクトに対する処理、例えば、処理対象となるオブジェクトの移動処理などが記述されている。なお、外部スクリプト記述部302に記載される外部データは、本S V Gを参照しているH T M Lを解釈することにより文書閲覧編集部112が自動的に読み込みを行い、該読み込まれたデータは、本S V G中に直接記述されたスクリプトと同様に取り扱うことが可能である。
40

【 0 0 3 5 】

303はオブジェクト定義部である。本定義部により、一つ以上の図形オブジェクトやイメージオブジェクトを一つの集合として定義することが可能であり、表示、編集時にはあたかも一つの図形として取り扱うことが可能になっている。図3中のオブジェクト定義部303の< g >から< /g >までの記述が单一のオブジェクト集合である。オブジェクト定義部303には更に、追加要素選択拡張部114によって表示される編集メニュー中に表示を行う個々のオブジェクトも記述される。

【 0 0 3 6 】

304は描画データ部である。表示拡張部113中に実際に描画するデータはすべて、この描画データ部304に記述される。描画データ部304において、オブジェクト定義部303にて定義したオブジェクトを用いるには、記述部"`<use xlink:href="`"の直後に、前記、オブジェクト定義部303にて定義したオブジェクトのidを指定することにより描画が可能となる。

【0037】

なお、オブジェクト定義部303に記述されたオブジェクトへの参照を用いずに、オブジェクトを直接記述することで描画することも可能である。

【0038】

305は選択矩形データ部で、編集操作時に選択されている図形を判別するための外接矩形の描画を実現する矩形オブジェクトや、選択した図形の回転や拡大縮小を行なうためのハンドルオブジェクトを描画する。この選択矩形データ部305に記述されるオブジェクトは、オブジェクト定義部303に記述されたオブジェクトと関連づけることにより、オブジェクト定義部303に記述されたオブジェクトに対して、例えば拡大縮小等の操作を拡張することなどに用いられる。

【0039】

より具体的には、オブジェクト定義部303中に記述されたオブジェクトの`<defs id="Shape">`～`</defs>`間に定義されたオブジェクトの利用のみが可能であり、選択矩形データ部305中の記述部"`<use xlink:href="`"の直後に、オブジェクト定義部303にて定義したオブジェクトのidを指定することで関連付けが行われる。

【0040】

306はイベントハンドラ記述部で、線の描画を行なう際のマウス18等を操作する際に得られる操作イベントを取得して処理を行うためのスクリプトが記述される。本実施形態においては、他のオブジェクトとの関連は無いため、内部の記述は省略している。

【0041】

図4は、保存処理起動記述部213によって起動されるスクリプト記述部202に記載のスクリプトの実行時に呼び出される、文書編集などの記述を削除する処理が記述されたスクリプトの例を示す図である。このスクリプトは、図1で説明した編集拡張部115によって解釈され、処理が実行される。

【0042】

401は、編集操作の際にマウスイベントを識別する際に利用するマウスイベントの配列である。

【0043】

402は、編集コード削除スクリプト記述部で、E C M Aスクリプト(European association for standardizing information and communications systems)によって、E C M A - 262として標準化)により記述された関数である。本スクリプト記述部に記載されるスクリプトが解釈、実行されることにより、本実施形態における編集のためのスクリプト類、マウスイベントハンドラの削除が実現される。

【0044】

411、412、413、414は、図3で説明した編集可能なS V G文書中に記述される識別用タグ中に記述されるidに相当する。

【0045】

411は、オブジェクト定義部303のオブジェクト定義のうち、選択矩形データ部305からのみ参照されているオブジェクト定義のまとめである"`<defs id="shape">`"中の"shape"に相当する(以下、id411、あるいはid"shape"411と称する。)。

【0046】

412は、選択矩形データ部305のデータの"`<g id="Control">`"中の"Control"に相当する(以下、id412、あるいはid"Control"412と称する。)。

【0047】

413は、外部スクリプト定義部302のデータの"`<defs id="ScriptDefs">`"中の

10

20

30

40

50

"ScriptDefs"に相当する(以下、id413、あるいはid"ScriptDefs"413と称する。)。

【0048】

414は、描画データ部304のデータの“`<g id="TOP">`”中の"TOP"に相当する(以下、id414、あるいはid"TOP"414と称する。)。

【0049】

これら411、412、413、414を用いた処理については後述する。

【0050】

図5は、図2に示したHTMLに基づいてクライアント111の文書閲覧編集部112によって表示される編集画面501の一例を示す図である。

10

【0051】

502は、表示拡張部113により編集対象の文書が表示される編集文書表示部である。503は、追加要素選択拡張部114により表示される追加要素選択部である。504は、図2に示したHTMLソースにおける再編集処理起動記述部212の開始タグから終了タグの記述を解釈し、その解釈結果を画面に表示した再編集指示部である。編集拡張部115は、この再編集指示部504がマウスクリックされたことに応答して再編集処理起動記述部212のスクリプトを実行する。

【0052】

505は、図2に示したHTMLソースにおける保存処理起動記述部213の開始タグから終了タグの記述を解釈し、その解釈結果を画面に表示した文書保存指示部である。この文書保存指示部505がマウスクリックされると、この文書保存指示部505に対応するHTML記述である保存処理起動記述部213のスクリプトが編集拡張部115によって解釈され実行される。なお、保存先はクライアントのディスクでもよいが、サーバのディスクを指定することも可能である。

20

【0053】

本実施形態における保存処理起動記述部213は、スクリプト記述部202中の"saveSVGdoc()"関数の呼び出しを行っており、さらに、`saveSVGdoc()`中の`SaveSVGFormat()`の呼び出しにより、図4の編集コード削除スクリプト記述部402がコールされる。そして、この編集コード削除スクリプト記述部402に記述されるスクリプトが実行されることによって編集コードが削除される。

30

【0054】

図6は、本実施形態における編集コード削除処理を示すフローチャートである。

【0055】

まず、ステップS601では、選択矩形データ部305および、線の描画を行なう際のイベントハンドラ記述部306を削除するとともに、選択矩形データ部305から参照されているオブジェクト定義部303中のオブジェクト定義を削除する。これにより、選択モードや描画モードの実現のためのソースコードが保存後のSVG文書から漏洩することを防止することができる。

【0056】

本実施形態における具体的な処理としては、まず、id"Shape"411を編集可能なSVG文書中より検索する。この検索結果は、選択矩形データ部305からのみ参照されているオブジェクト定義のまとめである“`<defs id="shape">`”定義が検索され抽出される。

40

【0057】

次に、抽出された“`<defs id="shape">`”タグから対応する終了タグの間の定義を削除する。

【0058】

更に、id"Shape"411の検索と同様に、id"Control"412を検索することで、選択矩形データ部305の“`<g id="Control">`”定義を抽出し、抽出された“`<g id="Control">`”タグから対応する終了タグの間の定義を削除する。本実施形態においては、“`<g id="Control">`”タグに対応する終了タグ“`</g>`”は、イベントハンドラ記述部3

50

0 6 も包含しているため、イベントハンドラ記述部 3 0 6 も同時に削除される。

【0 0 5 9】

以上、ステップ S 6 0 1 の処理が完了する。

【0 0 6 0】

次のステップ S 6 0 2 では、編集を実現するスクリプト記述を削除する。例えば、外部スクリプト定義部 3 0 2 の外部スクリプト定義を削除する。なお、描画データ部 3 0 4 中には、本ステップで削除を行う編集を実現するスクリプトを呼び出す記述がなされているが、前記呼び出し記述については、後述のステップにより削除を行っている。

【0 0 6 1】

本実施形態におけるステップ S 6 0 2 の具体的処理としては、`id="ScriptDefs" 4 1 3` を検索することにより、外部スクリプト定義部 3 0 のデータの “`<defs id="ScriptDefs">`” 定義を抽出し、抽出された “`<defs id=" ScriptDefs">`” タグから対応する終了タグの間の定義を削除する。

【0 0 6 2】

以上、ステップ S 6 0 2 の処理が完了する。

【0 0 6 3】

次に、ステップ S 6 0 3 ~ S 6 0 5において、描画データ部 3 0 4 中に存在する、ステップ S 6 0 2 で削除した編集を実現するためのスクリプトの呼び出し記述を削除する。

【0 0 6 4】

ステップ S 6 0 3 では、描画データ部 3 0 4 のデータの “`<g id="TOP">`” から対応する終了タグ “`</g>`” の間に記述されている描画オブジェクトの記述を一つ抽出し、抽出したデータのポインタを返す。

【0 0 6 5】

次に本ステップが呼び出されると、前回抽出したオブジェクトの次のオブジェクトを抽出される。すべての構成要素の抽出が終わると、データの終了を意味するNULLを通知する。

【0 0 6 6】

ステップ S 6 0 4 では、ステップ S 6 0 3 での描画オブジェクトの抽出が成功したかを判定する。ステップ S 6 0 3 の描画オブジェクトの抽出処理の結果として、描画オブジェクトが返されていれば、ステップ S 6 0 5 のマウスイベント属性削除処理に進む。一方、ステップ S 6 0 3 の描画オブジェクトの抽出処理の結果として、すべての描画オブジェクトの抽出が終了して “NULL” が返されたときには、この編集コード削除処理を終了する。

【0 0 6 7】

ステップ S 6 0 3 とステップ S 6 0 4 を組み合わせることにより、描画データ部 3 0 4 のすべての描画オブジェクトを順に抽出し、処理することが可能になる。

【0 0 6 8】

ステップ S 6 0 5 はマウスイベント属性削除処理である。ここでは、ステップ S 6 0 3 での描画オブジェクトの抽出処理により抽出された描画オブジェクト内に図 4 のマウスイベントの配列 4 0 1 中に記述されているマウスイベントの検索を行い、その検索の結果、マウスイベントが存在する場合は、描画オブジェクトよりマウスイベントを削除するものである。

【0 0 6 9】

本実施形態における具体的処理としては、描画データ部 3 0 4 の “`<g id="TOP">`” から対応する終了タグ “`</g>`” の間にあるデータのうち、太字で表示されたデータが削除されることになる。

【0 0 7 0】

以上の編集コード削除処理が実行されることで、保存された S V G 文書から編集用のスクリプトコードが漏洩することが防止され、閲覧のみが可能な S V G 文書の保存が可能になる。

【0 0 7 1】

10

20

30

40

50

次に、本実施形態における編集コードの復元処理について説明する。本実施形態においては、サーバ101における編集コード復元部104によりこの復元処理が行われる。図5中の再編集指示部504がマウスクリックされると、編集文書表示部503に表示されている文書がサーバ101にアップロードされ、これが復元対象の文書データとなる。なお、本実施形態においては、このようにアップロードされたSVG文書を編集コード復元処理の対象とするが、本発明における編集コード復元処理は、編集コード復元部104が編集対象であるSVG文書が認識可能であれば良く、その復元対象となるSVG文書の保存先を問うものではない。また、編集コード復元部104をサーバ101に配することなくクライアント111で実現しても良く、更には、いわゆるスタンドアロン構成においても編集コード復元処理が実現可能であることは言うまでもない。

10

【0072】

編集コード復元処理の説明に際して、まず、図7を用いて、本実施形態におけるイベントハンドラ復元データの説明を行う。

【0073】

図7はイベントハンドラ復元データの一部分を抽出した図である。イベントハンドラ復元データとは、マウスイベントに対応するスクリプト記述(イベントハンドラ)と、イベントハンドラと一緒に対応する識別子(イベントid)とが対をなして記述されるデータテーブルである。イベントidは、図3中の描画データ部304の下線部で示すように、描画オブジェクトの開始タグ内の“`id=`”に後続して記述されている。また、本実施形態においては、必須の記述とされている。

20

【0074】

図7において、701は識別子(イベントid)列である。702はイベントハンドラ列で、識別子列701にて定義されたイベントidに対応するマウスイベントを記述している。711～715は、イベントハンドラ復元データ例である。

【0075】

なお、本データに記述されるイベントidとそのイベントidに対応するマウスイベントハンドラとの組み合わせが、描画オブジェクトに設定されているイベントidとそのイベントidに対応するマウスイベントハンドラとの組み合わせに一致しており、かつ全てのイベントidとそのイベントidに対応するマウスイベントハンドラの組み合わせが本データに定義されていることが必要である。

30

【0076】

図8は、本実施形態における編集コード復元部104による編集コード復元処理を示すフローチャートである。

【0077】

まず、ステップS801では、編集を実現するためのスクリプト記述を挿入する。本実施形態における具体的処理としては、図3の外部スクリプト定義部302に編集用の外部スクリプト記述を追加する。

【0078】

次に、ステップS802で、選択矩形データ部305と線の描画を行なう際のイベントハンドラ記述部306、および選択矩形データ部305から参照されているオブジェクト定義部303中のオブジェクト定義の復元を行う。本実施形態における具体的処理としては、図3のオブジェクト定義部303のオブジェクト定義の末尾に、“`<defs id="shape">`”～“`</defs>`”タグ要素を追加し、このタグ間に拡大縮小指示用ハンドルのアイコン等、選択状態表示と共に使用するアイコンの描画オブジェクトを列記する。

40

【0079】

次に、`<svg ... >`～`</svg>`タグ内の要素の末尾に、図3の選択矩形データ部305の“`<g id="Control">`”要素を追加し、“`<g id="Control">`”～“`</g>`”の中にマウスクリックした際の選択矩形や、マウスが重なった際の選択矩形等、選択矩形の描画オブジェクトを列記する。

【0080】

50

さらに、上記した“`<g id="Control">`”タグ内の要素の末尾にイベントハンドラ記述部306で示される“`<g id="event-group">`”要素を追加する。上記タグ間に記述されるべき内部要素については、描画モードへの切り替え時に記述追加されるので、本ステップでは復元しない。

【0081】

以上、ステップS802の処理が完了する。

【0082】

次に、ステップS803～S805で、描画データ部304中に存在するステップS801で挿入した編集を実現するためのスクリプトの呼び出し記述を復元する。

【0083】

ステップS803では、描画データ部304のデータの“`<g id="TOP">`”から対応する終了タグ“`</g>`”の間に記述されている描画オブジェクトの記述を一つ抽出し、抽出したデータのポインタを返す。なお、次に本ステップが呼び出されると、前回抽出したオブジェクトの次のオブジェクトが抽出される。すべての構成要素の抽出が終わると、データの終了を意味するNULLを通知する。

【0084】

ステップS804では、ステップS803におけるオブジェクトの抽出が成功したか否かを判定する。ステップS803の描画オブジェクトの抽出処理の結果として、描画オブジェクトが返されていれば、ステップS805のマウスイベント属性追加処理に進む。ステップS803の描画オブジェクトの抽出処理の結果として、すべての描画オブジェクトの抽出が終わり“NULL”が返されたときには、この編集コード復元処理を終了する。

【0085】

ステップS803とステップS804を組み合わせることにより、描画データ部304のすべての描画オブジェクトを順に抽出し、処理することが可能になる。

【0086】

ステップS805は、マウスイベント属性追加処理である。ステップS803の描画オブジェクトの抽出処理にて抽出された描画オブジェクトに記述されたイベントid属性を、描画オブジェクト中のid属性の指し示す先頭“#”の直後から2文字ずつ取り出す。たとえば、図3の描画データ304内に記述される“`<g id="#AWAYAP-Pen-Insert4" ...>`”を例にとれば、下線で示したAWAYAP部を2文字ごとに分解し、AW, AY, APの3つのイベントid属性として取り出される。

【0087】

次に、取り出した各組の文字と一致するものを図7の識別子列701から検索し、その検索の結果、発見された識別子に対応してイベントハンドラ列702に定義されているマウスイベント属性を描画オブジェクトに追加する。

【0088】

この処理結果としては、描画データ部304の“`<g id="TOP">`”から対応する終了タグ“`</g>`”の間にあるデータのうち、図3における太字で表示されるデータが追加されることになる。

【0089】

以上の編集コード復元処理が実行されることで、保存されたSVG文書に編集用のスクリプトコードを復元することが可能になり、閲覧のみのSVG文書として保存したSVG文書の再編集が可能になる。

【0090】

(他の実施形態)

以上、本発明の実施形態を詳述したが、本発明は、複数の機器から構成されるシステムに適用してもよいし、また、一つの機器からなる装置に適用してもよい。

【0091】

なお、本発明は、前述した実施形態の機能を実現するソフトウェアのプログラムを、システムあるいは装置に直接あるいは遠隔から供給し、そのシステムあるいは装置のコンピ

10

20

30

40

50

ユータがその供給されたプログラムコードを読み出して実行することによっても達成される。その場合、プログラムの機能を有していれば、その形態はプログラムである必要はない。

【0092】

従って、本発明の機能処理をコンピュータで実現するために、そのコンピュータにインストールされるプログラムコード自体およびそのプログラムを格納した記憶媒体も本発明を構成することになる。つまり、本発明の特許請求の範囲には、本発明の機能処理を実現するためのコンピュータプログラム自体、およびそのプログラムを格納した記憶媒体も含まれる。

【0093】

その場合、プログラムの機能を有していれば、オブジェクトコード、インタプリタにより実行されるプログラム、OSに供給するスクリプトデータ等、プログラムの形態を問わない。

【0094】

プログラムを供給するための記憶媒体としては、例えば、フレキシブルディスク、ハードディスク、光ディスク、光磁気ディスク、MO、CD-ROM、CD-R、CD-RW、磁気テープ、不揮発性のメモリカード、ROM、DVD(DVD-ROM, DVD-R)などがある。

【0095】

その他、プログラムの供給方法としては、クライアントコンピュータのブラウザを用いてインターネットのホームページに接続し、そのホームページから本発明のコンピュータプログラムそのもの、もしくは圧縮され自動インストール機能を含むファイルをハードディスク等の記憶媒体にダウンロードすることによっても供給できる。また、本発明のプログラムを構成するプログラムコードを複数のファイルに分割し、それぞれのファイルを異なるホームページからダウンロードすることによっても実現可能である。つまり、本発明の機能処理をコンピュータで実現するためのプログラムファイルを複数のユーザに対してダウンロードさせるWWWサーバも、本発明のクレームに含まれるものである。

【0096】

また、本発明のプログラムを暗号化してCD-ROM等の記憶媒体に格納してユーザに配布し、所定の条件をクリアしたユーザに対し、インターネットを介してホームページから暗号化を解く鍵情報をダウンロードさせ、その鍵情報を使用することにより暗号化されたプログラムを実行してコンピュータにインストールさせて実現することも可能である。

【0097】

また、コンピュータが、読み出したプログラムを実行することによって、前述した実施形態の機能が実現される他、そのプログラムの指示に基づき、コンピュータ上で稼動しているOSなどが、実際の処理の一部または全部を行い、その処理によっても前述した実施形態の機能が実現され得る。

【0098】

さらに、記憶媒体から読み出されたプログラムが、コンピュータに挿入された機能拡張ボードやコンピュータに接続された機能拡張ユニットに備わるメモリに書き込まれた後、そのプログラムの指示に基づき、その機能拡張ボードや機能拡張ユニットに備わるCPUなどが実際の処理の一部または全部を行い、その処理によっても前述した実施形態の機能が実現される。

【図面の簡単な説明】

【0099】

【図1A】実施形態における文書データ処理システムのハードウェア構成を示す図である。

【図1B】実施形態における文書データ処理システムの機能構成を示す図である。

【図2】実施形態における編集画面のHTMLソースの一例を示す図である。

【図3】実施形態における編集対象の文書データの一例を示す図である。

10

20

30

40

50

【図4】実施形態における文書編集などの記述を削除する処理が記述されたスクリプトの例を示す図である。

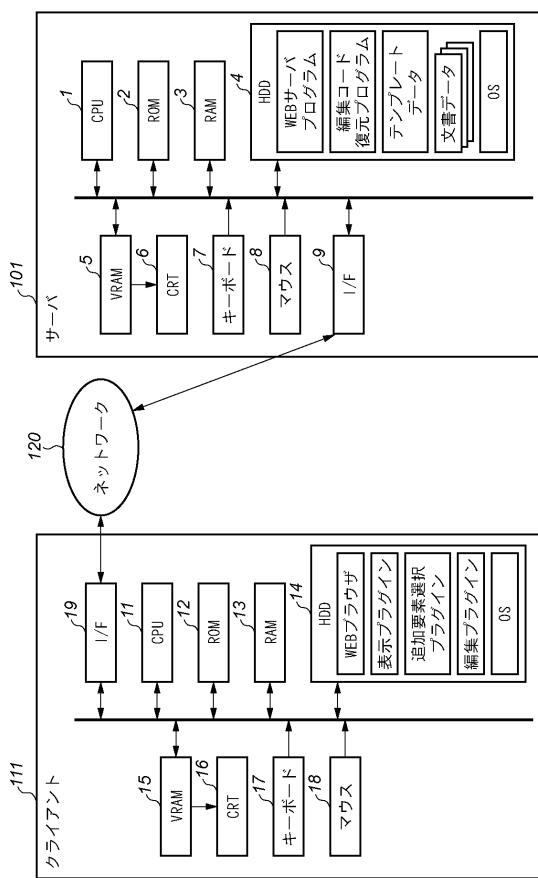
【図5】実施形態におけるクライアントに表示される編集画面の一例を示す図である。

【図6】実施形態における編集コード削除処理を示すフローチャートである。

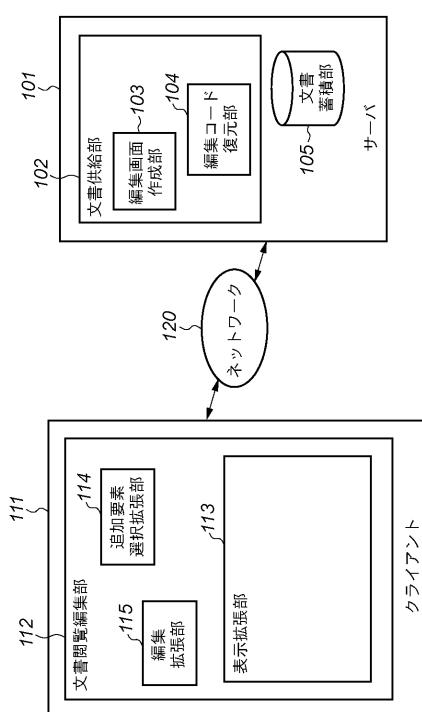
【図7】実施形態におけるイベントハンドラ復元データの一部分を示す図である。

【図8】実施形態における編集コード復元処理を示すフローチャートである。

【図1A】



【図1B】



【図2】

```

<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html>
  <head>
    <meta http-equiv="Content-Type" content="text/html; charset=Shift_JIS">
    <title>DO2003</title>
    <script language="JavaScript1.2">
      function saveSVGdoc( tako ) {
        document.svgEdit.window.SaveSVGFormat();
        var svgtext = document.svgEdit.window.printNode(document).replace(/>/g,>\n<');
        ...
      }
    </script>
  </head>
  <body leftmargin="0" topmargin="0" marginwidth="0" marginheight="0" bgcolor="#B19F7C">
    <script language="javascript" src="js/load.js">
    </script>
    <table width="900" border="2" bordercolor="#B19F7C">
      <tr>
        <td width="900" border="0" height="540">
          <tr>
            <td width="500" height="500" align="right" valign="top">
              <embed name="svgEdit" src="template.svg" width="500" height="500">
            </td>
            <td width="400" align="left" valign="top" height="500">
              <object classid="clsid:XX-XXXX" codebase="http://download.URL"
width="400" height="500" id="movie">
                <param name="movie" value="penpro7.swf" swliveconnect="true">
                <embed src="penpro7.swf" type="application/x-shockwave-flash" width="400" height="500"> </embed>
              </object>
            </td>
          </tr>
          <td align="left" height="30" valign="top" width="400">
            <table width="400" border="1" height="28" bordercolor="#B19F7C">
              <tr>
                <td>
                  <a href="javascript:LoadSvg()">
                    
                  </a>
                  <a href="javascript:savesSVGdoc()">
                    
                  </a>
                </td>
              </tr>
            </table> </td> </tr>
          </table> </td> </tr>
        </table>
      </body>
    </html>
  
```

【図3】

```

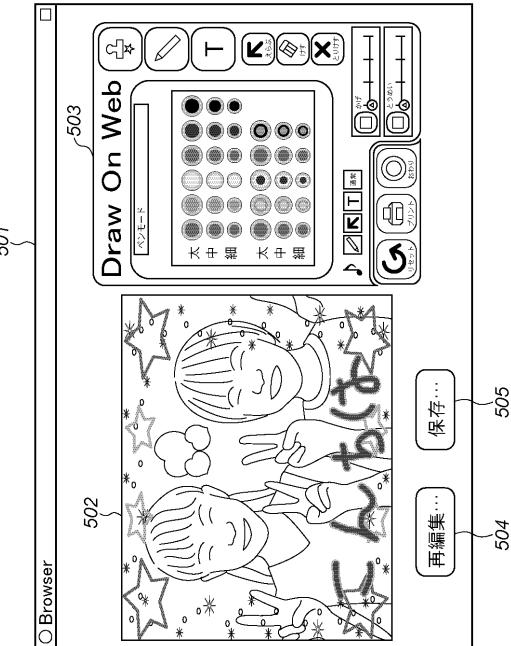
<?xml version="1.0" standard="no"?>
<!DOCTYPE svg PUBLIC "-//W3C//DTD SVG 1.0//EN"
"http://www.w3.org/TR/2001/REC-SVG-20010904/DTD/svg10.dtd"[
<!ENTITY ns_svg "http://www.w3.org/2000/svg">
]>
<svg width="500" height="500" viewBox="0 0 500 500" style="enable-background: new 0 0 500 500"
xmlSpace="preserve" onLoad="initialize();"
xmns="http://www.w3.org/2000/svg" preserveAspectRatio="MidMid meet" zoomAndPan="magnify">
  <defs id="ScriptDefs">
    <script type="text/ecmascript" xlink:href="js/savesvg.js"/>
    ...
  </defs>
  <defs id="PngStampContainer">
    <g id="PngStamp0">
      <image xlink:href="png/hear2.png" x="-50" y="-50" width="90" height="90"/>
    </g>
  </defs>
  <defs id="Shape">
    <polygon id="sh_enknob" style="fill:yellow;stroke:green;stroke-linejoin:round;" points="-4,-4,-4,2,-2,4,0
6' />
    <polygon id="sh_robknob" style="fill:green;stroke:blue;stroke-linejoin:round;" points="-4,-6,4,-6,2,-4,4,2
6' />
  </defs>
  <g id="TOP">
    <defs id="TextObj"/>
    <g id="BASE" onclick="BASEClick(evt)">
      <path id="Canvas" style="fill:#FFFFFF;stroke:red;" d="M500,500H0V0h500V500"/>
      <rect id="PhotoImage" x="0" y="23" width="500" height="453" xlink:href="image/san1.jpg"/>
      <g id="WAVAYAP-Pen-Insert4" onmousedown="setDown(evnt)" onmouseover="setOver(evnt)" onmouseout="resetOver(evnt)" transform="translate(123.5,123) scale(1,1) rotate(0)"/>
      <g id="Path-Pen-Insert4" transform="translate(-123.5,-123) scale(1,1) rotate(0)"/>
      <g id="L0-Pen-Insert4" fill="none" stroke-linecap="round" stroke-linejoin="round" stroke-width="3" stroke="green" style="stroke-dasharray: 3 3; visibility: hidden;">
        <use xlink:href="#LP-Pen-Insert4-1"/>
      </g>
      <g id="LP-Pen-Insert4" fill="none" stroke-linecap="round" stroke-linejoin="round" stroke="none" stroke-width="2">
        <use xlink:href="#LP-Pen-Insert4-1"/>
      </g>
      <g id="bbox-Pen-Insert4" x="-63" y="-51.5" width="126" height="103" fill="none" stroke="none"/>
      <g id="WACAYAP-PngStamp-Insert2" onmousedown="setDown2(evnt)" onmouseover="setOver2(evnt)" onmouseout="resetOver(evnt)" transform="translate(168.91,32) rotate(0)"/>
      <g id="PngStamp0" x="0" y="0" width="182.363" height="75" fill="none" stroke="blue" stroke-linejoin="round" stroke-width="3" visibility="hidden" transform="translate(87.50,64.5) scale(1,1) rotate(0)" stroke-opacity="0.3"/>
      <use id="enknob" xlink:href="#sh_enknob" transform="translate(233.95,156.95) rotate(0)" visibility="visible" style="stroke-dasharray: 3 3; visibility: visible;"/>
      <use id="robknob" xlink:href="#sh_robknob" transform="translate(233.95,25.05) rotate(0)" visibility="hidden" style="stroke-dasharray: 3 3; visibility: hidden;"/>
    </g>
  </g>
  <g id="Control">
    <rect id="wfoverbox" x="0" y="0" width="131.905" height="131.905" fill="none" stroke="red" stroke-dasharray="3,2" stroke-width="2" visibility="hidden" transform="translate(69.04,82.04) scale(1,1) rotate(0)"/>
    <rect id="wfselbox" x="0" y="0" width="182.363" height="75" fill="none" stroke="blue" stroke-linejoin="round" stroke-width="3" visibility="hidden" transform="translate(87.50,64.5) scale(1,1) rotate(0)" stroke-opacity="0.3"/>
    <use id="enknob" xlink:href="#sh_enknob" transform="translate(233.95,156.95) rotate(0)" visibility="visible" style="stroke-dasharray: 3 3; visibility: visible;"/>
    <use id="robknob" xlink:href="#sh_robknob" transform="translate(233.95,25.05) rotate(0)" visibility="hidden" style="stroke-dasharray: 3 3; visibility: hidden;"/>
  </g>
</g>
</svg>
  
```

【図4】

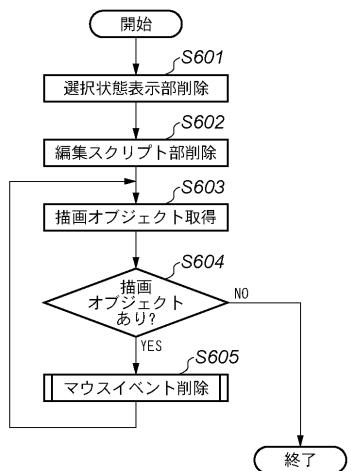
```

/////////////////////////////
// savesvg.js:SVG変換関連処理ファイル
//
// マウスイベントの定義文
var _mouseEvent = new Array();
_mouseEvent[0] = "onmousedown";
_mouseEvent[1] = "onmouseup";
_mouseEvent[2] = "onmousemove";
_mouseEvent[3] = "onmouseover";
_mouseEvent[4] = "onmouseout";
_mouseEvent[5] = "onclick";
/////////////////////////////
// 開算用データへの変換メイン処理
//
function SaveSVGFormat()
{
  // 選択矩形データ
  var defShape = svgDoc.getElementById("Shape");
  if (defShape) {
    defShape.parentNode.removeChild(defShape);
  }
  // Control
  var defControl = svgDoc.getElementById("Control");
  if (defControl) {
    defControl.parentNode.removeChild(defControl);
  }
  // script
  var defScript = svgDoc.getElementById("ScriptDefs");
  if (defScript) {
    defScript.parentNode.removeChild(defScript);
  }
  // イベントハンドラの削除
  var elmTOP = svgDoc.getElementById("TOP");
  if (elmTOP) {
    // Object Mouse Event Handler
    for( var cl = elmTOP.firstChild ; cl ; cl = cl.nextSibling ) {
      // マウスイベントの削除
      for( var i=0, i<_mouseEvent.length; i++ ) {
        if ( cl.hasAttribute(_mouseEvent[i]) ) {
          cl.removeAttribute(_mouseEvent[i]);
        }
      }
    }
  }
}
  
```

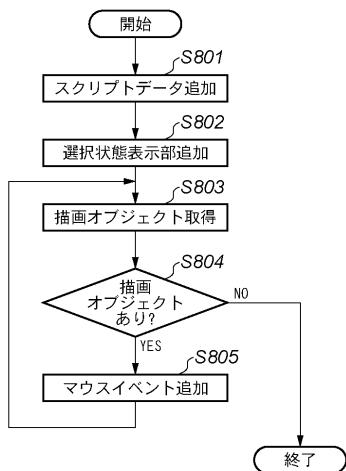
【図5】



【図6】



【図8】



【図7】

識別子	イベントハンドラ	
AW	onmousedown="_setDown (evt)"	～711
AR	-	～712
AC	onmousedown="_setDown2 (evt)"	～713
...	...	
AY	onmouseover="_setOver (evt)"	～714
...	...	
AP	onmouseout="_resetOver (evt)"	～715
...	...	

フロントページの続き

(72)発明者 内藤 起久雄
東京都大田区下丸子3丁目30番2号 キヤノン株式会社内

(72)発明者 田中 光治
東京都大田区下丸子3丁目30番2号 キヤノン株式会社内

(72)発明者 羽鳥 和重
東京都大田区下丸子3丁目30番2号 キヤノン株式会社内

(72)発明者 野口 利之
東京都大田区下丸子3丁目30番2号 キヤノン株式会社内

F ターム(参考) 5B009 QA06 QB03 QB12 QB13 TA06 TB11 VC03
5B050 BA06 BA10 BA16 CA07 CA08 EA19 FA02