

(19) 日本国特許庁 (JP)

(12) 特 許 公 報 (B2)

(11) 特許番号

特許第4829662号
(P4829662)

(45) 発行日 平成23年12月7日 (2011. 12. 7)

(24) 登録日 平成23年9月22日 (2011. 9. 22)

(51) Int. Cl.

F I

G 0 6 T 11/60 (2006. 01)

G 0 6 T 11/60 1 0 0 A

G 0 6 Q 50/00 (2006. 01)

G 0 6 F 17/60 1 3 2

G 0 6 F 17/21 (2006. 01)

G 0 6 F 17/21 5 3 6

請求項の数 12 (全 47 頁)

(21) 出願番号 特願2006-104716 (P2006-104716)
 (22) 出願日 平成18年4月5日 (2006. 4. 5)
 (65) 公開番号 特開2007-279992 (P2007-279992A)
 (43) 公開日 平成19年10月25日 (2007. 10. 25)
 審査請求日 平成21年4月3日 (2009. 4. 3)

(73) 特許権者 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 康德
 (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 細坪 利彦
 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】 自動レイアウトシステムおよび情報提供方法およびサーバーシステム

(57) 【特許請求の範囲】

【請求項 1】

シンジケーションデータを取得し表示可能なクライアントと、前記クライアントに対し
て配信される前記シンジケーションデータを生成するサーバーシステムとを有する自動レ
イアウトシステムであって、

前記サーバーシステムは、

コンテンツデータを、テンプレートデータに従って配置処理するレイアウト手段と、
前記レイアウト手段による配置処理においてエラーが生じた場合、当該エラーに係るデ
ータへのリンク情報を含むシンジケーションデータを生成する生成手段と、

前記生成手段により生成されたシンジケーションデータを前記クライアントに提供する
情報提供手段とを有し、

前記クライアントは、

前記エラーに関連するテンプレートデータと、該テンプレートデータに関連するコンテ
ンツデータとを、前記サーバーシステムから提供されたシンジケーションデータのリンク
情報を基に取得する取得手段と、

取得されたテンプレートデータとコンテンツデータとを基に再レイアウトを行う再レイ
アウト手段と、

再レイアウトによりエラーが解消されたテンプレートデータの保存を前記サーバーシ
ステムに指示する指示手段と

を備えることを特徴とする自動レイアウトシステム。

10

20

【請求項 2】

前記生成手段は、前記シンジケーションデータとして、RSSまたはatom形式のデータを生成し、

前記情報提供手段は、前記シンジケーションデータをRSSまたはatom形式で提供し、前記エラーに係るデータをHTML形式で提供することを特徴とする請求項1に記載の自動レイアウトシステム。

【請求項 3】

前記シンジケーションデータは、前記配置処理において生じたエラーの要約を表す要約情報を含むエラー配信データであることを特徴とする請求項1に記載の自動レイアウトシステム。

【請求項 4】

前記生成手段は、前記レイアウト手段による配置処理において、テンプレートに含まれる枠内にテキストデータが納まらないテキストオーバーフローと、テンプレートに含まれる枠内にイメージデータが納まらないイメージオーバーフローと、枠内に収めるべきコンテンツデータがないコンテナエラーのいずれかが生じた場合に、前記シンジケーションデータを生成することを特徴とする請求項3に記載の自動レイアウトシステム。

【請求項 5】

前記シンジケーションデータは、前記配置処理において生じたエラーに係るデータへのリンク情報として、当該エラーに係るテンプレートデータおよびコンテンツデータそれぞれへのリンク情報を含むことを特徴とする請求項1乃至4のいずれか1項に記載の自動レイアウトシステム。

【請求項 6】

シンジケーションデータを取得し表示可能なクライアントと、前記クライアントに対して配信される前記シンジケーションデータをレイアウトサーバーにより生成してウェブサーバーにより提供する情報提供方法であって、

前記レイアウトサーバーが、コンテンツデータを、テンプレートデータに従って配置処理するレイアウト工程と、

前記レイアウトサーバーが、前記レイアウト工程による配置処理においてエラーが生じた場合、当該エラーに係るデータへのリンク情報を含むシンジケーションデータを生成する生成工程と、

ウェブサーバーが、前記生成工程により生成されたシンジケーションデータを前記クライアントに提供する情報提供工程と、

前記クライアントが、前記エラーに関連するテンプレートデータと、該テンプレートデータに関連するコンテンツデータとを、前記ウェブサーバーから提供されたシンジケーションデータのリンク情報を基に取得する取得工程と、

前記クライアントが、取得されたテンプレートデータとコンテンツデータとを基に再レイアウトを行う再レイアウト工程と、

前記クライアントが、再レイアウトによりエラーが解消されたテンプレートデータの保存を前記レイアウトサーバーに指示する指示工程と

を有することを特徴とする情報提供方法。

【請求項 7】

前記生成工程では、前記シンジケーションデータとして、RSSまたはatom形式のデータを生成し、

前記情報提供工程では、前記シンジケーションデータをRSSまたはatom形式で提供し、前記エラーに係るデータをHTML形式で提供することを特徴とする請求項6に記載の情報提供方法。

【請求項 8】

前記シンジケーションデータは、前記配置処理において生じたエラーの要約を表す要約情報を含むエラー配信データであることを特徴とする請求項6に記載の情報提供方法。

【請求項 9】

前記生成工程は、前記レイアウト工程による配置処理において、テンプレートに含まれる枠内にテキストデータが納まらないテキストオーバーフローと、テンプレートに含まれる枠内にイメージデータが納まらないイメージオーバーフローと、枠内に収めるべきコンテンツデータがないコンテナエラーのいずれかが生じた場合に、前記シンジケーションデータを生成することを特徴とする請求項 8 に記載の情報提供方法。

【請求項 10】

前記シンジケーションデータは、前記配置処理において生じたエラーに係るデータへのリンク情報として、当該エラーに係るテンプレートデータおよびコンテンツデータそれぞれへのリンク情報を含むことを特徴とする請求項 6 乃至 9 のいずれか 1 項に記載の情報提供方法。

10

【請求項 11】

サーバーシステムであって、
コンテンツデータをテンプレートに従ってレイアウトした際に生じたレイアウトエラーをチェックするチェック手段と、

前記チェック手段によるレイアウトエラーのチェック結果を基に、タイプ 1 とタイプ 2 の 2 つのタイプの R S S データを生成し、生成した R S S データを情報配信サーバーへアップロードする R S S データ配信手段とを備え、

前記タイプ 1 の R S S データは、R S S データのアイテム要素のタイトル項目にレイアウトエラーが発生したテンプレートの固有情報を設定し、そのタイトル項目に対応するリンク要素には、当該レイアウトエラーを生じたテンプレートが保存されている保存場所のリンク情報を設定した R S S データであり、

20

前記タイプ 2 の R S S データは、前記チェック手段によりエラーの発生が特定されたテンプレート毎に用意された R S S データであり、夫々の R S S データのアイテム要素のタイトル項目にはレイアウトエラーが発生した日時を設定し、そのタイトル項目に対応するリンク要素にはその日時にレイアウトエラーが発生した際に、当該レイアウトエラーが発生したテンプレートで用いたコンテンツが保存されている保存場所のリンク情報を設定した R S S データであり、

前記サーバーシステムはさらに、前記生成手段により生成されたシンジケーションデータを提供する提供手段を備え、

クライアントに配信された R S S データのリンク要素で設定されたリンク情報を基に当該クライアントがシンジケーションデータの取得要求を行った場合、前記提供手段が、要求されたシンジケーションデータを要求元のクライアントに提供することを特徴とするサーバーシステム。

30

【請求項 12】

請求項 11 に記載のサーバーシステムと、該サーバーシステムにより生成した R S S データをクライアントに配信する情報配信サーバーと、クライアントとを有する自動レイアウトシステムであって、

前記クライアントは制御ユニットを有し、

該制御ユニットは、

前記情報配信サーバーにより提供される R S S データを表示する R S S リーダーを制御し、

40

前記 R S S リーダーは、前記タイプ 1 の R S S データのチャンネルが選択された場合、タイプ 1 の R S S データを受信して、タイプ 1 のアイテム要素のリンク項目を基にリンク情報を表示し、そのリンク情報が選択されたことに応じて前記サーバーシステムからテンプレートを受信し、

前記 R S S リーダーは、前記タイプ 2 の R S S データのチャンネルが選択された場合、タイプ 2 の R S S データを受信して、タイプ 2 のアイテム要素のリンク項目を基にリンク情報を表示し、そのリンク情報が選択されたことに応じて前記サーバーシステムからコンテンツを受信し、

前記制御ユニットはさらに、

50

前記RSSリーダーを介してテンプレートを受信した場合は、その受信したテンプレートを編集するために前記クライアントにインストールされているテンプレート編集アプリケーションを起動し、

前記RSSリーダーを介してコンテンツを受信した場合は、前記クライアントにインストールされているテンプレート編集アプリケーションを起動し、前記受信したコンテンツを用いたプレビューを表示させるよう前記クライアントを制御することを特徴とする自動レイアウトシステム。

【発明の詳細な説明】

【技術分野】

【0001】

本発明は、例えばクライアントからの指示等に基づき、選択されたコンテンツをテンプレートに流し込んでレイアウトを行い、成果物であるドキュメントをクライアントに送信するコンテンツサーバおよびレイアウトシステムに関する。

【背景技術】

【0002】

近年、ユーザの好みに応じてカスタマイズされた電子カタログを配信するWebシステムがある。そのシステムは、Webサーバが提供するWebページにおいて、利用者がテンプレートと、そのテンプレートに流し込むコンテンツを選択し、その結果の電子ドキュメントをクライアントが受け取る。なお、データ(コンテンツデータ)をテンプレートに流し込むとは、コンテンツデータを、テンプレートで定義されたレイアウトに従って配置することである。したがってテンプレートにコンテンツデータが流しこまれると、コンテンツデータと、そのレイアウトを示すレイアウトデータとを組にした、レイアウト済みコンテンツデータが生成される。

【0003】

このようなシステムでは、コンテンツやテンプレートを管理し、提供するコンテンツサーバがそのシステムの一部を形成している。

【0004】

また、このようなシステムにおいては、ユーザ選択に応じた電子ドキュメントを生成するために、自動レイアウトシステムが利用されている。自動レイアウトシステムでは、コンテンツを流し込む各コンテナのサイズやコンテナ間隔を可変設定することが可能である。これにより、テキストやイメージなどのコンテンツのボリュームに応じて、テンプレートに配置されたコンテナのサイズやコンテナ間の距離が変更できる。そのため、一般的に、自動レイアウトシステムでは、一つのテンプレートで、様々なボリュームのコンテンツと、それらコンテンツの組み合わせに対応することが出来る。なおコンテナとは、テンプレート内に定義された枠であり、その枠内に文字や画像などのコンテンツが配置される。

【0005】

しかしながら、自動レイアウトシステムでは、異なるコンテンツや、コンテンツの組み合わせバリエーションの違い等によって、各コンテナやコンテナ間に設定した可変設定の想定を超えるデータが流し込まれることがある。その場合、テキストがコンテナ内に収まり切らないテキストオーバーフローやイメージがコンテナ内に収まり切らないイメージオーバーフローなどのレイアウトエラーが発生することがある。

【0006】

このレイアウトエラーの発生をチェックし、ユーザがテンプレートを再編集し、レイアウトエラーに対する修正を加えることが出来るよう、テンプレートエディタには、いわゆるプリフライト検査を行うブルーフィング機能を備えているものが存在する。プリフライト検査とは、本番の運用に備えて行われる本番運用前検査であり、試験データを用いて正常に動作するか否かを判定する検査を含む。すなわち、技術的な観点からは、自動レイアウトシステムのプリフライト検査とは、自動レイアウトシステムの通常の使用と同じである。運用の目的が、コンテンツデータあるいはテンプレートデータの検証にある点で、プリフライト検査と通常の使用とは異なる。プリフライト検査のことを、単にプリフライト

10

20

30

40

50

と称する場合もある。

【 0 0 0 7 】

ブルーフィング機能を持つ自動レイアウトシステムのプリフライト検査では、コンテンツの試験データがテンプレートに流し込まれて試行的にレイアウトが行われる。そしてそのプリフライト検査の結果が、ユーザに対して表示される。

【 0 0 0 8 】

テンプレート編集者であるユーザは、そのプリフライト検査結果を見ながら、テンプレートを再編集し、レイアウトエラーが発生しないテンプレートとなるよう、各コンテナやコンテナ間に設定した可変設定の値やレベルを変更する。

【 0 0 0 9 】

テンプレート編集者であるユーザは、コンテンツサーバからテンプレートに適用するコンテンツをダウンロードした後、このブルーフィング機能を用いながら、レイアウトエラーが発生しないテンプレートの作成、編集作業を行う。

【 0 0 1 0 】

コンテンツサーバ管理者であるユーザは、テンプレート編集者によって作成されたレイアウトエラーが発生しないテンプレートをコンテンツサーバにアップロードして登録する。登録されたテンプレートは、Webクライアントから電子ドキュメントを要求するユーザがWebページ上で選択可能なように設定、配置される。

【 0 0 1 1 】

文書の自動レイアウトシステムは、たとえば特許文献1等に記載されている。また、Webにおける更新をクライアントに知らせる技術としてRSS(RDF(Resource Description Framework) site summary)という技術がある(例えば非特許文献1, 2等参照)。

【特許文献1】特開2005-216174号公報

【非特許文献1】Dan Brickley他10名、"RDF Site Summary (RSS) 1.0"、[online]2001年5月30日、RSS-DEV Working Group、[2005年3月17日検索]、インターネット<URL:http://purl.org/rss/1.0/spec>

【非特許文献2】M. Nottingham, R. Sayer編、"The Atom Syndication Format draft-ietf-atompub-froamat-03"、[online]2004年10月20日、RSS-DEV Working Group、[2005年3月17日検索]、インターネット<URL:http://atompub.org/2004/10/20/draft-ietf-atompub-froamat-03.html>

【発明の開示】

【発明が解決しようとする課題】

【 0 0 1 2 】

この自動レイアウトシステムにおいては、コンテンツサーバが扱うコンテンツデータは、随時、コンテンツサーバ管理者によって、追加、更新される。そのため、ブルーフィング機能を用いながらレイアウトエラーが発生しないよう編集されたテンプレートも、サーバに登録した後は、新しいコンテンツまたは新しいコンテンツの既存のコンテンツとの組み合わせで、レイアウトエラーが発生する可能性が高い。

【 0 0 1 3 】

上記従来の技術では、テンプレートをサーバに登録した後でレイアウトエラーが発生した際、サーバが記録するレイアウト時のエラーログを見て、問題が発生しているテンプレートを特定し、そのテンプレートを修正する必要があった。その際、管理者は、そのテンプレートで問題を起こしたコンテンツデータのレコード群、および、そのレコード群がテンプレートに流し込まれた順序を、更にログから特定する。そしてその特定したレコード群をコンテンツサーバからエディタ環境に手動で指定してダウンロードする。テンプレート再編集作業のために、更に、レコード群の並び順を手動で指定して、ブルーフィングエラーをエディタで再現させる。このような手間がかかる手順を踏む必要があった。そのため、一つのテンプレートで、異なるケースのブルーフィングエラーが発生した際や、問題が発生したテンプレートが多く存在する場合、テンプレート修正のための再編集作業は、

10

20

30

40

50

多くの労力がかかっていた。

【 0 0 1 4 】

本発明は、上記の問題を鑑みてなされたものであり、サーバに登録した後、レイアウトエラーが発生した際に、エラーを解消すべく行われるテンプレートの編集作業を容易にするコンテンツサーバおよびレイアウトシステムを提供することを目的とする。

【課題を解決するための手段】

【 0 0 1 5 】

上述した課題を解決するために、本発明の自動レイアウトシステムは以下の構成を備える。すなわち、シンジケーションデータを取得し表示可能なクライアントと、前記クライアントに対して配信される前記シンジケーションデータを生成するサーバーシステムとを有する自動レイアウトシステムであって、前記サーバーシステムは、コンテンツデータを、テンプレートデータに従って配置処理するレイアウト手段と、前記レイアウト手段による配置処理においてエラーが生じた場合、当該エラーに係るデータへのリンク情報を含むシンジケーションデータを生成する生成手段と、前記生成手段により生成されたシンジケーションデータを前記クライアントに提供する情報提供手段とを有し、前記クライアントは、前記エラーに関連するテンプレートデータと、該テンプレートデータに関連するコンテンツデータとを、前記サーバーシステムから提供されたシンジケーションデータのリンク情報を基に取得する取得手段と、取得されたテンプレートデータとコンテンツデータとを基に再レイアウトを行う再レイアウト手段と、再レイアウトによりエラーが解消されたテンプレートデータの保存を前記サーバーシステムに指示する指示手段とを備える。

【 0 0 1 6 】

あるいは、サーバーシステムであって、コンテンツデータをテンプレートに従ってレイアウトした際に生じたレイアウトエラーをチェックするチェック手段と、前記チェック手段によるレイアウトエラーのチェック結果を基に、タイプ1とタイプ2の2つのタイプのRSSデータを生成し、生成したRSSデータを情報配信サーバーへアップロードするRSSデータ配信手段とを備え、前記タイプ1のRSSデータは、RSSデータのアイテム要素のタイトル項目にレイアウトエラーが発生したテンプレートの固有情報を設定し、そのタイトル項目に対応するリンク要素には、当該レイアウトエラーを生じたテンプレートが保存されている保存場所のリンク情報を設定したRSSデータであり、前記タイプ2のRSSデータは、前記チェック手段によりエラーの発生が特定されたテンプレート毎に用意されたRSSデータであり、夫々のRSSデータのアイテム要素のタイトル項目にはレイアウトエラーが発生した日時を設定し、そのタイトル項目に対応するリンク要素にはその日時にレイアウトエラーが発生した際に、当該レイアウトエラーが発生したテンプレートで用いたコンテンツが保存されている保存場所のリンク情報を設定したRSSデータであり、前記サーバーシステムはさらに、前記生成手段により生成されたシンジケーションデータを提供する提供手段を備え、クライアントに配信されたRSSデータのリンク要素で設定されたリンク情報を基に当該クライアントがシンジケーションデータの取得要求を行った場合、前記提供手段が、要求されたシンジケーションデータを要求元のクライアントに提供する。

【発明の効果】

【 0 0 1 7 】

本発明によれば、サーバに登録したテンプレートにレイアウトエラーが発生した際、そのブルーフィンギングエラーをクライアントのテンプレートエディタで再現させる環境を容易に構築することが出来る。そのため、問題が発生したテンプレートの修正作業が容易になり、自動レイアウト処理の生産性を向上させることができる。

【発明を実施するための最良の形態】

【 0 0 1 8 】

〔実施形態〕

本発明に好適な実施形態について以下説明を行う。まず本実施形態の自動レイアウトシステムの概略を説明する。

【 0 0 1 9 】

(1) レイアウトサーバは、プリフライト検査または本番運用時に、レイアウトエラーの履歴情報を記録する。レイアウトエラーの履歴は、それが発生したテンプレートの識別情報および流し込んだコンテンツ群の識別情報および流し込みの順序とともに記憶される。なお、テンプレートへのコンテンツの流し込みとは、テンプレートで定義される枠（コンテンツ）内に、テキストや画像、グラフィクスデータといったコンテンツデータを配置する処理をいう。なお本例ではコンテンツデータとは、印刷対象となることから、時間をパラメータとしない静的なデータを指す。またコンテンツデータを単にコンテンツと呼ぶ場合もある。また、レイアウトエラーの履歴情報をプリフライト履歴という。プリフライト検査時に記録したエラーのみならず、本番運用時に収集したレイアウトエラーの履歴も、プリフライト履歴と称する。

10

【 0 0 2 0 】

(2) コンテンツサーバは、このレイアウトエラーのプリフライト履歴を、テンプレート名を見出しにして、RSSでシンジケーション（情報配信サービス）する。すなわち、テンプレートの識別名を見出しとし、そのテンプレートデータの識別名やURI及びエラー数等の履歴情報をRDF形式に編集し、一定の保存場所に保存する。RSSのデータは、一般に利用されているものであり、無償あるいは有償で流通しているRSSリーダソフトウェアや、ウェブブラウザや電子メールソフトのRSSリーダ機能で処理可能な形式を持つ。本例では、RSSリーダ機能とウェブブラウザ機能とを併せ持つクライアントソフトウェアが用いられる。このタイプのソフトウェアは、ユーザインターフェースの上ではウェブページデータとRSSデータとを特に区別せずに閲覧できる。たとえばRSSデータに含まれるハイパーリンクをクリックすれば、そのリンク先が通常のHTMLデータであっても、ウェブブラウザ機能のコンポーネントを用いてHTMLデータを表示できる。また、ウェブブラウザ機能により表示するデータに関連づけられたアプリケーションプログラムを起動して、ブラウザのペインの中で、そのアプリケーションプログラムによる処理を実行することもできる。

20

【 0 0 2 1 】

(3) クライアントは、レイアウトエラーのプリフライト履歴のシンジケーションをニュースフィードとして、ブラウザのRSSリーダで表示する。表示されるのは、エラーの内容と、エラーに係るテンプレートデータの識別名、エラーに係るコンテンツデータの識別名等である。テンプレートデータの識別名およびコンテンツデータの識別名は、いずれもそれぞれのデータそのものへのリンク情報を関連づけられており、それらしき別名をクリックされると、リンク先のデータがアクセスされる。

30

【 0 0 2 2 】

(4) クライアントブラウザのRSSリーダで、レイアウトエラーが発生したテンプレート名（URIリンク）の一つが管理ユーザによりクリック（プリフライト履歴の選択）されると、サーバに対してテンプレート名の選択が伝達される。サーバは、レイアウトエラーのプリフライト履歴に記録された、選択されたテンプレートデータをクライアントに送信する。

40

【 0 0 2 3 】

(5) クライアントブラウザは、テンプレート編集プログラム（テンプレートエディタともいう）を起動する。テンプレートエディタは、ローカル（クライアント自身）にインストールされたり、ActiveXコンポーネントやJava（登録商標）等で記述されたものがサーバからダウンロードされる。

【 0 0 2 4 】

(6) サーバは、クライアントに送信したテンプレートで発生したレイアウトエラーの履歴を、レイアウトエラー（の識別名）を見出しにして、RSSフィード（ニュースフィード）する。すなわち、レイアウトエラーの発生を見出しとし、そのレイアウトエラーを生じたコンテンツデータの識別名やURIをRDF形式に編集し、一定の保存場所に保存する。

50

【 0 0 2 5 】

(7) クライアントブラウザの R S S リーダでの表示は、テンプレート固有のレイアウトエラー (プリフライト履歴) のチャンネルに切り替わる。

【 0 0 2 6 】

(8) クライアントブラウザの R S S リーダにフィードされたレイアウトエラー (URL リンク) の一つがユーザによりクリック (プリフライト履歴の選択) された際、サーバは、そのレイアウトエラーにリンクされたコンテンツ群をクライアントに送信する。

【 0 0 2 7 】

(9) クライアントブラウザは、サーバから送信されたコンテンツ群をテンプレートエディタに渡す。テンプレートエディタは、そのコンテンツ群をテンプレートに流し込み、レイアウトプレビューを生成する。コンテンツ群をテンプレートに流し込むとは、コンテンツデータをテンプレートに従ってレイアウトすることである。

10

【 0 0 2 8 】

(1 0) クライアントブラウザのユーザは、テンプレートエディタでレイアウトチェックを実行する。すなわち (8) で得たコンテンツデータの、(4) で得たテンプレートデータにしたがう配置を試みる。その結果、R S S リーダに表示される、フィードされたレイアウトエラーと同じエラーが試行の結果として得られる筈である。ユーザはその結果を見ながらレイアウトの再編集を行い、レイアウトエラーを修正して行く。

【 0 0 2 9 】

(1 1) クライアントブラウザのユーザは、R S S リーダに表示される別のレイアウトエラーのリンクをクリック (プリフライト履歴の選択) できる。これにより、他のコンテンツ群をサーバから受け取り、テンプレートエディタでそのレイアウトエラーを修正して行く。

20

【 0 0 3 0 】

(1 2) クライアントブラウザのユーザは、レイアウトエラーを修正し、再編集を終えたテンプレートをサーバにアップロード (再登録) する。

【 0 0 3 1 】

(1 3) 編集されたテンプレートの再登録 (更新) を受けたサーバは、そのテンプレートに関する履歴情報を削除する。また、プリフライト履歴のシンジケーションから、そのテンプレートに関する情報を削除し、再び、テンプレート名を見出しにして R S S フィード (ニュースフィード) を行う。すなわち、エラーが解消されたことを配信する。

30

【 0 0 3 2 】

以上のようにして、本実施形態のレイアウトシステムによれば、レイアウトエラーの発生が検出されると、エラーの発生を示す情報と、エラーに係るテンプレートとコンテンツデータとを特定できる情報を、R S S フィードにより配信する。フィードを受信したクライアント (管理者等) は、受信したテンプレートとコンテンツデータとを特定できる情報とから、発生したレイアウトエラーを直ちに再現することができる。それによってエラーの原因を特定し、エラーが生じないように、テンプレートデータやコンテンツデータを修正する作業を容易に行える。また、修正後の再試験も直ちに出来る。

【 0 0 3 3 】

このため、レイアウトエラー解消のための作業が軽減され、迅速化できる。これにより、エラー解消のための修正作業の生産性に加えて、レイアウトされたレイアウト済みデータを生成する処理の生産性を向上させることができる。

40

【 0 0 3 4 】

また、エラーの発生の情報とその修正のための情報がクライアントに通知されるため、レイアウトサーバを常時監視し続ける必要がなく、監視のために割り当てられる資源を減少させることができる。

【 0 0 3 5 】

< システム構成図 >

本発明の実施形態を説明する前に、本発明を適用可能なシステム構成、およびアプリケ

50

ーション構成について説明する。

【 0 0 3 6 】

図 1 は、クライアントからのリクエストに応じ、サーバでバリアブルデータから電子カタログのドキュメントを生成し、レスポンスとしてクライアントに配信するカタログ配信システム 1 0 0 を図示している。

【 0 0 3 7 】

図 2 は、図 1 に示されるサーバやクライアント端末を実現するコンピュータのブロック図である。図 2 のコンピュータ 2 0 1 は、管理クライアントコンピュータモジュール 1 0 1、クライアントコンピュータモジュール 1 0 2、コンテンツサーバ 1 0 3、ファイルサーバ 1 0 4、レイアウトサーバ 1 0 5、Webサーバ 1 0 6 のハードウェア構成を示す。システム 1 0 0 上で実施可能となるテンプレート編集アプリケーションプログラム 1 2 1 などは、コンピュータ 2 0 1 により実行される。テンプレート編集や印刷のステップはコンピュータ 2 0 1 によって実行されるソフトウェアの指示によって実施される。ソフトウェアは例えば以下に記述されるような記憶装置を含むコンピュータの可読媒体に格納される。ソフトウェアはコンピュータの可読媒体からコンピュータにロードされ、コンピュータ 2 0 1 によって実行される。そのようなソフトウェアやコンピュータプログラムを記録した可読媒体はコンピュータプログラム製品である。

【 0 0 3 8 】

コンピュータモジュール 2 0 1 はキーボード 2 0 2 やマウス 2 0 3 のようなポインティングデバイスなどの入力装置をつなぎ、ディスプレイ装置 2 1 4 や状況に応じてはローカルプリンタ 2 1 5 を含む出力装置を連結する。入力/出力インターフェース 2 0 8 はコンピュータモジュール 2 0 1 をネットワーク接続 2 0 7 から接続してシステム 1 0 0 の他のコンピュータ装置につなげることができる。そのネットワーク接続 2 0 7 の典型はローカルエリアネットワーク (LAN)、あるいはワイドエリアネットワーク (WAN) である。

コンピュータモジュール 2 0 1 は、典型的には少なくとも 1 つのプロセッサユニット 2 0 5、例えば半導体のランダムアクセスメモリ (RAM) やリードオンリーメモリ (ROM) から構成されるメモリユニット 2 0 6 を含む。また、ビデオインタフェース 2 0 7 を含む INPUT/OUTPUT (I/O) インターフェース、キーボード 2 0 2 やマウス 2 0 3 のための I/O インターフェース 2 1 3 を含んでいる。記憶装置 2 0 9 は典型的にハードディスクドライブ 2 1 0 やフレキシブルディスクドライブ 2 1 1 を含んでいる。図には示されていないが磁気テープドライブもまた使用される可能性がある。CD-ROM ドライブ 2 1 2 は不揮発性のデータソースとして提供される。コンピュータモジュール 2 0 1 は、コンピュータシステムの常套的な方法によって、相互接続バス 2 0 4 を介して通信を行うコンピュータモジュール 2 0 1 のプロセッサ 2 0 5 から I/O インターフェース 2 1 3 を利用する。

【 0 0 3 9 】

図 2 に記述したコンピュータの例では IBM 互換 PC や SUN 社の Sparc Station、あるいはそれらを含んだコンピュータシステムが考えられる。

【 0 0 4 0 】

テンプレート編集アプリケーションプログラム 1 2 1、および WEB ブラウザアプリケーションプログラム 1 2 2 または 1 2 5 は典型的にハードディスクドライブ 2 1 0 に常駐し、プロセッサ 2 0 5 により実行、読み込み、コントロールされる。プログラム 1 2 1、1 2 2、1 2 5 の媒介記憶装置とネットワーク 1 0 7 からフェッチされるデータはハードディスクドライブ 2 1 0 に呼応して半導体メモリ 2 0 6 を使用する。いくつかのインスタンスではアプリケーションプログラム 1 2 1、1 2 2、1 2 5 が CD-ROM やフレキシブルディスク上でエンコードされ、対応するドライブ 2 1 2 や 2 1 1 を通じて読み込まれユーザに提供される。あるいはもう一つの方法としてアプリケーションプログラム 1 2 1、1 2 2、1 2 5 はネットワーク接続 2 0 7 からユーザによって読み込まれるかもしれない。さらにソフトウェアは、コンピュータ可読媒体からコンピュータモジュール 2 0 1

内にロードされる可能性もある。その媒体には、磁気テープ、ROM、集積回路、光磁気ディスク、無線インターフェースや赤外線通信で接続された他の機器、PCMCIAカードのようなコンピュータ可読カード、インターネットやイントラネットで接続された他の機器を含む。前述は単に関連するコンピュータ可読メディアの模範である。他のコンピュータ可読媒体も使用される可能性はある。

【0041】

またテンプレート編集アプリケーション121はバリエブルデータ印刷を行うための2つのソフトウェアコンポーネントを含んでいる。これらのうち1つめのコンポーネントはレイアウトエンジン127であり、これは長方形の範囲内で与えられた制限やサイズによって矩形と線の位置を計算するためのソフトウェアコンポーネントである。2つめのコンポーネントであるユーザインターフェース128はユーザにドキュメントテンプレートを作成させ、ドキュメントテンプレート内でデータソースと関連付けるメカニズムを提供する。ユーザインターフェース128とレイアウトエンジン127はコミュニケーションチャンネル123を介して通信する。ドキュメント生成のためのバリエブルデータのデータソースは、一般的にコンテンツを管理するデータベースアプリケーションを動かしている他のコンピュータによって構成されたコンテンツサーバ103上にある典型的なデータベース119である。管理クライアントコンピュータ101はネットワーク接続107の手段によってコンテンツサーバ103と通信する。テンプレート編集アプリケーションプログラム121は管理クライアントコンピュータ101が一般的に他のコンピュータで構成されるファイルサーバ104に保存されるドキュメントテンプレートを生成する。またテンプレート編集アプリケーションプログラム121はドキュメントテンプレートにデータを流し込んでドキュメントを生成する。これらのドキュメントは管理クライアントコンピュータ101のローカルファイルシステムに保存されるか、ファイルサーバ104に保存される。

【0042】

WEBブラウザアプリケーション122、125は、WEBページを閲覧するためのアプリケーションプログラムである。WEBブラウザアプリケーション122、125は、他のコンピュータによって構成されたWEBサーバ106が提供するWEBページへのアクセス、およびWEBサーバ106が提供するリクエストフォームを介したWEBサーバ106への情報送信を行う。リクエストフォームは、たとえばHTMLフォームなどである。また、WEBブラウザアプリケーション122は、コミュニケーションチャンネル124を介してテンプレート編集アプリケーションプログラム121と通信し、WEBサーバ106から受信したデータの仲介や、WEBサーバ106へのデータ送信の仲介を行う。

【0043】

レイアウトサーバ105は典型的なコンピュータである。レイアウトサーバ105は、コンテンツサーバ103から供給されたデータとファイルサーバ104のファイルシステムに保存されたドキュメントテンプレートとを、レイアウトエンジン126によって結合し、ドキュメントを生成する。すなわち、コンテンツデータを、テンプレートに従ってレイアウトすることができる。

【0044】

レイアウトサーバ105には、後述の例えば図30、図31に示される手順を実行して、レイアウトエラーの通知に関するRSSデータを編集するRSSフィールドエディタ131が備えられている。作成されたRSSデータはウェブサーバ106にアップロードされる。もちろんRSSフィールドエディタ131はレイアウトサーバ105に、その一部として含まれていても良い。

【0045】

WEBサーバ106は典型的なコンピュータであり、WEBクライアントであるWEBブラウザアプリケーション122、125に送付したリクエストフォームの返信を受信する。そしてリクエストフォームに記載されたユーザのリクエスト内容に応じて、コンテンツサーバ103に、要求されたコンテンツとテンプレートでのドキュメント生成を依頼し

、依頼により生成されたドキュメントをレスポンスとしてクライアントに返す。

【 0 0 4 6 】

またウェブサーバ 1 0 6 には、RSS フィードエディタ 1 3 1 により RSS データが作成されている場合には、RSS データ 1 3 2 がアップロードされて保存される。保存された RSS データは、RSS のクライアント (RSS リーダ) によりたとえば定期的にポーリングされて、それによりクライアントの持つエラーに関する情報が更新される。また、ウェブサーバは、RSS データに埋め込まれたハイパーリンク情報に対するリクエストを受信して、必要な処理をコンテンツサーバ等に要求して実行させ、クライアントからのリクエストに応じたデータをクライアントに応答する。これは上述の通りである。

【 0 0 4 7 】

プリントサーバ 1 0 8 は直接ネットワークにつながっていないプリンタにネットワーク機能を提供するコンピュータである。プリントサーバ 1 0 8 とプリンタ 1 0 9 は典型的な通信チャネル 1 2 9 を介して接続される。

【 0 0 4 8 】

コンテンツサーバ 1 0 3 は典型的なコンピュータであり、アプリケーションサーバでもある。

【 0 0 4 9 】

<アプリケーション構成図>
(メインウインドウ)

図 3 で参照されるように、ユーザインターフェース 1 2 8 は、操作時にビデオディスプレイ 1 4 4 に表示されるアプリケーションウインドウ 3 0 1 によって形成されたユーザインターフェースを含んでいる。ウインドウ 3 0 1 は、メニューバー 3 0 2 とツールバー 3 0 3、そしてワークエリア 3 0 6 とオプションのパレット 3 1 1 とカーソル/ポインタデバイス 3 1 3 から特徴付けされる。メニューバー 3 0 2 は、非表示にすることや、スクリーン上の色々な場所に移動することが可能である。ワークエリア 3 0 6 は、マウス 2 0 3 の位置・動作によって場所を移動可能である。

【 0 0 5 0 】

メニューバー 3 0 2 は、周知の技術として知られているように、メニューオプションの階層の下に拡張される多くのメニューアイテム 3 0 4 を持つ。

【 0 0 5 1 】

ツールバー 3 0 3 は、アプリケーションの特別なモードによって非表示にする、または表示することが可能な多くのツールボタンとウィジェット 3 0 5 を持つ。

【 0 0 5 2 】

オプションのルーラー 3 0 8 はワークエリア内のポインタ、ページ、ライン、マージンガイド、コンテナまたはオブジェクトの位置を示すために使われる。

【 0 0 5 3 】

パレット 3 1 1 はバリアブルデータライブラリのような追加機能にアクセスするために使われる。パレット 3 1 1 は移動、リサイズ、クローズをするためのウインドウコントロール 3 1 2 を持つ。パレット 3 1 1 はオプションで、ワークエリアの前面に表示される、あるいはオブジェクトの背面に隠される。パレット 3 1 1 はアプリケーションウインドウ 3 0 1 の範囲内のみに表示されることを制限される、あるいはアプリケーションウインドウ 3 0 1 の外側に部分的・全体を表示することを許される。

【 0 0 5 4 】

ツールバーエリア 3 0 3 は少なくとも、次のユーザ選択可能な『ボタン』を持つ (図 4 を参照)。

(1) 選択ツールボタン 4 0 3 : コンテナの辺を選択、移動、サイズ変更、リサイズそしてロック・ロック解除するために使われる。コンテナは、(複数)コンテナの周りに選択ボックスをドラッグする、あるいは複数コンテナを選択する間にCTRLキーを押しつづけることによって、複数選択を可能にする。

(2) イメージコンテナツールボタン 4 0 5 : スタティックあるいはバリアブルイメージ

10

20

30

40

50

を持つコンテナを作成するために使われる。

(3) テキストコンテナツールボタン 404 : スタティックあるいはバリアブルテキストを持つコンテナを作成するために使われる。

(4) リンクツールボタン 406 : コンテナ間の距離をコントロールするリンクを作成するために使われる。

これらのボタンは、周知の技術であるように操作状況に合わせて変化するアイコンのツールチップとして実装される。

【0055】

(ドキュメントテンプレート)

ワークエリア 306 はドキュメントテンプレートのデザインを表示・編集するために使われる。これはユーザに下準備で印刷されたドキュメントの概観をデザインすること、そしてマージされたドキュメントが、バリアブルデータの量・サイズに基づいてどのように変化するかを理解することを可能にする。

【0056】

もし外部データソースがテンプレートにリンクされていたら、現在のドキュメントのプレビューができるように、バリアブルテキストとイメージがそれらのコンテナに表示される。

【0057】

ドキュメントの構造とバリアブルデータコンテナの描写をする視覚的な手がかりは、カーソルをコンテナ上に移動させたときや、コンテナを選択したときにいつも表示される。

【0058】

ワークエリア 306 はスクロールバー 307 とオプションのルーラー 308 とドキュメントテンプレート 309 に特徴付けられる。ドキュメントテンプレート 309 はページが複数あることを示すことができる。

【0059】

与えられたドキュメントテンプレートのページサイズは、周知の技術としてユーザによって指定される。それぞれのドキュメントでの実際のページ数は、バリアブルデータによって変化するかもしれない。もし1ページ内にフィットできなかった時、追加のページは自動的に作成される。

【0060】

それぞれのページ内の境界線は、ページ上の印刷可能なオブジェクトの最大幅を示す、任意のページマージン 310 である。

【0061】

また、図4は1ページのドキュメントテンプレート 309 上に表示することが可能なオブジェクトの例である。それらは、複数のコンテナ 407、408 と、任意に適用するアンカーアイコン 409 と固定されていない辺 410、リンク 412 そしてスライダー 413 を持つ。

【0062】

(コンテナ)

コンテナは、ドキュメントテンプレート内の固定(静的)あるいは可変(バリアブル)テキスト・イメージを持つスペースであり、他のコンテナやオブジェクトとレイアウトされる。コンテナはマウス 203 動作でポインタ 313 を使い、ユーザインターフェースで示されるように移動、サイズ調整そして再作成される。

【0063】

より正確にはコンテナは、設定の集まり、視覚的表現そしてインタラクションと編集動作をもっている。下記はコンテナの定義の全部分である。

(1) コンテナは固定あるいは可変のコンテンツを持つ。可変コンテンツは、データソースからとってきて、異なるドキュメントでは異なるかもしれない、という意味でダイナミック(動的)である。可変コンテンツは印刷されないような時間によって変更されたり、アニメーションするコンテンツを含むことは意図していない。同様に、固定コンテンツは

10

20

30

40

50

コンテナを使って生成される全てのドキュメントで、同じように表示される。けれども、可変コンテナ（動的オブジェクト）の動作によって、固定コンテンツはそれぞれのドキュメントで位置が異なるかもしれない。

（２）コンテナは、コンテンツに適用される背景色、ボーダー、フォント・スタイルのようなテキスト設定のような装飾機能を持っている。

（３）コンテナはドキュメントを生成したときにデータソースからのデータとマージされる。装飾機能は、どんな固定コンテンツでもそうであるように、典型的に印刷された出力物で可視である。可変コンテンツはデータソースから特定のデータの表示をもたらす。コンテナのこの表現は例えば印刷されるか、スクリーン 144 上で表示されるか、その両方が可能である。

（４）コンテナはユーザインターフェースを持つ、例えばコンテナの編集そして表示設定のためのインタラクティブな GUI を持つ。インターフェースの要素は典型的にスクリーン 144 上に表示される、しかしドキュメントには印刷されない。ユーザインターフェース 128 は、背景色やフォントのようなコンテナの装飾機能のいくつかを表示する、そしてコンテナの設定の編集や表示を許すための機能を追加する。ユーザインターフェース機能の特別な目的の例としては、ボーダー、あるいは、コンテナのサイズや位置を対話的に変更、表示するための角アイコンがある。あるいはコンテナがデータソースからデータをマージされたとき、コンテナの動作を示すための上塗りした数、線、アイコン、テキストがある。このように本実施形態のレイアウト編集のユーザインターフェースは、直接編集技術とコンテナの GUI コンポーネントに伴う表示方法の集まりである。

【 0064 】

（コンテナ制約）

公知の技術によると、コンテナはそれぞれのドキュメントで表示されるコンテンツを、どのように結びつけるか制御する制約がある。これらの制約（固定・可変コンテンツをコンテナと結びつけると共に）は、ユーザが一つのドキュメントテンプレートから、多数のドキュメントの生成を制御する主要な方法である。制約の例は『このコンテナのコンテンツの高さは、最大値 4 インチです。』である。もうひとつの制約は『コンテナのコンテンツの左端は、それぞれのドキュメントで同じ水平位置で表示しなければならない。』である。ここに記述される内容は、GUI を使ってこのような制約を表示、編集するためのいろいろな方法である。

【 0065 】

イメージがページ上に定義された場所を持っているように、固定コンテンツの配置を指定するコンテンツプレースホルダーは、デジタル印刷技術でよく知られている。下記の検討で、コンテナは位置とサイズを持ち、それらは公知の技術で知られている手法で編集され、表示される。また、この検討では、パリアブルデータ印刷に特化した方法での表示・編集に焦点を合わせて説明する。

【 0066 】

コンテナは、ユーザにドキュメントのコンテンツのサイズ・位置を指定することを可能にする。いくつかのドキュメントは一つのドキュメントテンプレートから生成されるので、コンテナは多数の可能性と制約を指定・表示するためにユーザインターフェースを使わなければならない。

【 0067 】

1 つのコンテナの辺は、関連付けられたコンテンツがドキュメント内で表示される仮想の境界線を定義する。このように、この特許でコンテナの左辺を論じることが、関連付けられたコンテンツがどんなドキュメント内でも表示可能である最も左の辺を論じることと同じである。同様に、コンテナの高さを論じることが生成されたドキュメントで関連付けられたコンテンツの高さの制約を論じることとして理解される。この特許明細書では、ユーザインターフェース 128 を参照してコンテナの辺あるいは大きさを論じるところで、この区別は明らかにされるであろう。

【 0068 】

下記の検討では、コンテンツの表示を制限するために使われるいくつかの値を定義している用語『固定』は、全てのドキュメントで同じである。

(1) もしコンテナの幅が固定なら、そのコンテナに関連付けられたコンテンツに割り当てられる幅は、全てのドキュメントで同じになるだろう。

(2) もしコンテナの高さが固定なら、関連付けられたコンテンツに割り当てられる高さは、全てのドキュメントで同じになるだろう。

(3) もし距離の制約が固定なら、指定された距離は全てのドキュメントのための制約である。

(4) もしコンテナの左右辺が固定なら、左右辺の水平位置が、全てのドキュメントで同じであることを意味している、しかしコンテナの高さあるいは垂直方向の位置は、変わるかもしれない。例えば、もしコンテナの左辺が固定なら、そのコンテナに関連付けられたコンテンツは、あるドキュメントでページの上に近く表示され、他のドキュメントでページの下に近く表示されることがある。しかし左辺は全てのケースで同じ水平位置である。

(5) もしコンテナの上下辺が固定なら、辺の垂直位置がページに関して全てのドキュメントで同じであることを意味している、しかしコンテナの幅あるいは水平位置は変わるかもしれない。

(6) コンテナの垂直軸はコンテナの右と左辺の平行で、そして中間に位置される想像上の垂直線である。もしコンテナの垂直軸が固定なら、コンテナの左右辺の水平位置の平均は、すべてのドキュメントで同じである。この制約で、コンテナの幅は変化するかもしれない。左右辺両方が異なったドキュメントで、垂直軸にもっとも遠いかもっとも近いかもしれない。しかし軸は全てのドキュメントで同じ水平位置にある。コンテナの高さと水平位置はこの制約によって影響されない。

(7) 同様に、もし水平軸が固定なら、コンテナの上そして下辺が垂直に位置されることを制約する、けれども高さは、この制約によって影響されない。

(8) もし水平、垂直軸両方が固定なら、コンテナの中心位置が固定されていることを意味する、しかし、幅・高さはこの制約によって影響されない。

(9) もしコンテナの角、コンテナの辺の中間位置、あるいはコンテナの中心位置が固定なら、すべてのドキュメントで同じ場所で、そしてコンテナに関連付けられた同じ場所で表示される。例えば、もしコンテナの左上角が固定なら、配置されたコンテナの左上位置が全てのドキュメントで同じになることを意味している。

(10) 垂直辺あるいは軸はページの左辺、あるいは右辺、あるいは左ページマージン、あるいは右ページマージン、あるいは他の水平位置に関連付けられて固定になる。同様に、水平辺あるいは軸はページの上あるいは下辺あるいはマージン、あるいは他の垂直位置に関連付けられて固定になる。

【0069】

『固定』の反対は、辺、軸、角、中間位置、あるいはドキュメント制約がドキュメント間で変化するかもしれないことを意味している『可変』である。しかしドキュメントの特定の設定でそうする必要が無い場合もある。例えば、変更による辺の実際の好ましい位置のような他の外部制約がある。しかし、もし外部制約が適用されないなら、辺は固定されてないものとしてラベル付けされているので辺の位置は変更できる。

【0070】

(コンテナ表示・編集)

[新規コンテナの作成方法]

コンテナは、テキストコンテナとイメージコンテナの2種類で記述される。テキストコンテナはテキスト、そして埋め込みのイメージを持つ。イメージコンテナは、イメージだけを持つ。

【0071】

新規テキストコンテナとイメージコンテナは、テキストコンテナツール404、あるいはイメージコンテナツール405をマウス203でクリックし、テンプレート309に四角形をドラッグすることによって、ドキュメントテンプレート309上に作成される。

【 0 0 7 2 】

代わりにコンテナは、適切なツール 4 0 4、4 0 5 をアクティブにした後に、ドキュメントテンプレート 3 0 9 上でクリックすることによって簡単に作成される。デフォルトサイズのコンテナが挿入されるか、新規コンテナの寸法を入れるために、ダイアログボックスあるいは他のプロンプトが提供される。いくつかのコンテナは自動的に前もって定義されたか、計算されたスキーマによって、作成・配置される。また他の案が考えられるかも知れない。

【 0 0 7 3 】

[コンテナの表示方法]

図 5 A - 5 D はコンテナの模範的な辺ルールを例示している。アプリケーション 1 2 1 は、コンテナの辺の状態をこれら図に示された標示オブジェクトを用いて表現する。すなわち、アプリケーション 1 2 1 は、塗りつぶし線 5 0 3 あるいは点線 5 0 4 で辺を描き、アンカー、ハンドル、スライダー (、拡張アイコン 5 0 5、そして色でコンテナを表示する。アンカーとは、辺の近くに描画された符号 5 0 6、5 0 7、5 0 9 によって示されるような線、形状、あるいはアイコンである。ハンドルとは、移動、修正するために辺、形の近くに描画されたコントロール点 5 0 2 である。スライダーとは、辺の両サイドに描画された短い並行線 4 1 3 である。

【 0 0 7 4 】

図 5 A - 5 D のコンテナ表示方法のルールは、つぎの通りである。

- (1) それぞれの辺を固定するために、塗りつぶし線で描画する。
 - (2) もし幅が固定なら、左と右の辺を塗りつぶし線で描画する。
 - (3) もし高さが固定なら、上と下の辺を塗りつぶし線で描画する。
 - (4) 軸は描画しない。
 - (5) まだ描画されていない全ての辺は、それぞれの辺の近くに拡張アイコンが描画され、点線になる。
 - (6) 垂直辺あるいは軸のそれぞれのペアで、もし両者が固定なら、交差点にアンカーが描画される。
 - (7) それぞれの固定辺で、もし辺のどこにもアンカーが描画されていなければ、エッジの中央にスライダーが描画される。
 - (8) 垂直辺あるいは軸のそれぞれのペアで、もしアンカーやスライダーが描画されていなければ、交差点にハンドルが描画される。
- ルール (1)、(2)、(3) に従う線は、もしそれらが固定あるいは制限されているなら、塗りつぶし線で描画される。ルール 5 に従う可変の辺は、点線で描画される。ルール 6、7、8 に従う固定された点は、アンカーを表示し、いくつかの固定された辺はスライダーを表示し、他はハンドルを表示する。

【 0 0 7 5 】

上記で、辺はただ一度描画されることが必要である。それでもし、ルールが描画されるべき辺に影響すれば、そのとき後のルールはふたたび描画されるべき辺に影響しないだろう。表示上の都合によっては、例えばもしコンテナがとても小さくアイコンがお互い重なるか、あるいは他の表示機能を不明瞭にするならば、アイコンは異なって、あるいは省略されて描画されてもよい。

【 0 0 7 6 】

可変の辺が描画される場所は、コンテナのコンテンツに依存する。後で記述されるように、ドキュメントテンプレートにコンテンツがマージされて、ユーザインターフェースで可視になることを意味する、『動的な校正処理』が使われる。代替の実行手段としては、すべてのドキュメントで平均化されるコンテナのコンテンツエリア、あるいは可変の辺がユーザインターフェースで、どこにレイアウトされるべきか決定するほかの手段で使われることができる。

【 0 0 7 7 】

これらのコンテンツ表現は、コンテナの辺の状態を表示するグラフィカルな手段を提供

する。その表現の解釈は下記のとおりである。

(1) 図 4 の辺 4 1 0 のように、点線はコンテナのコンテンツに依存して、ドキュメント内の辺の位置を意味する。

(2) 塗りつぶし線は制限された辺を意味する、なぜなら、辺は固定されている (辺 4 1 4)、あるいはコンテナの幅・高さが固定されている (コンテナ 4 0 8 では両方が固定されている) ため。

(3) アンカーは辺そして軸が交差した場所が固定されていることを意味する。それゆえ、アンカー点は、すべてのドキュメントの水平、垂直位置で現れるだろう。当然アンカーは固定される。図 4 のアイコン 4 0 9 は、交差する辺 4 1 4 が固定されていることを意図している、アンカーアイコンの例である。

10

(4) スライダーは関係付けられた辺が固定されていることを意味する、しかしコンテナは辺に沿って多くの位置『スライドの長さ』で位置を定められる。例えば、図 4 で、スライダー 4 1 3 はコンテナ 4 0 8 のコンテンツが、ドキュメント内で特定のダイアグラムで表される位置の、左あるいは右に表示される場合もありえる。

【 0 0 7 8 】

これらのアイコン・辺のいくつかあるいは全ては、どのツール、どのコンテナを選択・ハイライトあるいはアクティブにするかによって、描画されたりされなかったりする。一般的に、コンテナの辺・アイコンはドキュメントテンプレートのデザインの手助けであるため、印刷物には描画されない。

【 0 0 7 9 】

幅・高さの最小値・最大値の設定は、副次的なダイアログウインドウに表示される。

20

【 0 0 8 0 】

図 5 A で、コンテナ 5 0 1 は幅・高さ両方が固定されていない。固定された辺 5 0 3 は塗りつぶし線で表現される。可変の辺 5 0 4 は点線で表現される。拡張アイコン 5 0 5 は、隣接する辺 5 0 4 が可変であることを示す、追加あるいは代替のインジケータである。

【 0 0 8 1 】

図 5 B で、コンテナ 5 0 1 は幅・高さ両方が可変である。アンカーアイコン 5 0 6 は、交差している両方の辺 5 0 3 が固定されていることを、追加あるいは代替に示している。

30

【 0 0 8 2 】

図 5 C で、コンテナ 5 0 1 は、コンテナの拡大あるいは縮小が任意のアンカーアイコン 5 0 7 で示されるような中心点の周りを平等に広がるという状態で、幅・高さ両方が可変である。

【 0 0 8 3 】

図 5 D で、コンテナ 5 0 1 は、上辺 5 0 8 が固定されていることを除けば、幅・高さ両方が可変である。上辺 5 0 8 の中心に位置付けられて示されるアンカーアイコン 5 0 9 は、固定されている、そしてまたコンテナの左・右辺が、アイコンを通り垂直に描画される中心軸 (垂直軸) の周りを、拡大・縮小することを示す。

【 0 0 8 4 】

40

(リンク)

リンクは、コンテナとコンテナの関連を示している。関連とはコンテナ間の距離を示しており、リンクによって関連付けられたコンテナ同士は、互いのレイアウト変更の影響を受けてレイアウトを計算する。図 4 の 4 1 2 で示されているものがリンクであり、この図ではコンテナ 4 0 7 と 4 0 8 とを関連付けている。リンクの設定方法および、リンクで関連付けられたコンテナの動的レイアウト計算方法については、後述する。

【 0 0 8 5 】

(リンクの設定方法)

図 6 はリンクの設定方法を示したフロー図である。また図 7 は UI 例を示している。それらを用いてコンテナにリンクを設定する方法について説明をする。

50

【0086】

まず、リンクを設定するためには、リンクを設定するためのコンテナ（最低2つ）を作成する（0601）。図7は、2つのコンテナを作成してリンクを設定する場合の例を示している。次に、前述したリンクツール406を選択した状態にする（0602）。図7（A）-（C）は、コンテナを作成し、リンクツールが選択されていた状態から、リンクを設定する動作を示している。図7（A）-（C）を順に説明していく。

【0087】

図7（A）のコンテナ0701と0702は、前述した図4のコンテナ407と408と同じであり、固定されている辺を意味する。また、オブジェクト0703と0704は、オブジェクト409と同じであり、アンカーを意味する。オブジェクト0705はマウスポインタを意味している。まず、リンクを設定する片方のコンテナをクリックして選択する（0603）。次に図7（B）で示されるように、もう片方のコンテナまでマウスポインタを移動して、クリックする（0604）。図7（B）の線0706は図7（A）でクリックをした位置と移動したマウスポインタを結んだ線を示しており、どの位置にリンクが設定されるのかをユーザに示すUIである。処理0604が終わった後、設定した場所に0707で示されるリンクUIが表示される（0605）。コンテナは図7（C）の状態になる。また、リンクが設定されたことにより、コンテナのUIも自動的に変更される（0606）。0708は、点線で示されている辺であり、前述した通り可変の辺を示している。図7（C）のようにコンテナの辺の状態が変化したのは、リンクを設定したことにより、コンテナの辺を可変にする必要があるためである。この処理は、リンクを設定したにもかかわらず、全ての辺が固定という矛盾を引き起こすことを防ぐために、自動的に行われる。また、オブジェクト0709は、図5のオブジェクト505と同じで、リンクを設定したことにより、コンテナが変化できる方向をユーザに視覚的に示したしるしである。図7（C）の例では、左のコンテナの右辺と右のコンテナの左辺が可変に変化した。これは一例であり、右コンテナが、図4の413で示したスライダーを持つ設定に変化してもかまわない。

< 動的レイアウト計算方法（全体フロー） >

図8は動的レイアウト計算のフローを示している。まず、プレビューモードを選択する（0801）。前述した自動レイアウトシステムでは、その動作モードには、レイアウトモードとプレビューモードとがある。レイアウトモードでは、コンテナを作成して、そのコンテナ間の関連付けを行ってテンプレートデータを作成する。プレビューモードでは、作成したテンプレートにレコードを挿入して、実際にレコードが挿入された後のレイアウト結果をプレビューする。このプレビューモードにおいて、実際のレコードが挿入され、レイアウトが計算される。ただし、プレビューモードは、表示上でのレイアウト計算である。実際に印刷する場合においても、レコードを挿入してレイアウトを計算する。その際の計算方法も同じである。プレビューモードになったら、プレビューするレコードを選択して挿入する（0802）。レコードの挿入を行うと、そのレコードをレイアウトするために計算を行う。（0803）。処理0803で計算されたレイアウトを表示する（0804）。他のレコードについてもプレビューを行うかどうかを判断する（0805）。処理0805で、他のレコードについてプレビューを行う必要がないと判断した場合は、プレビューモードを終了する（0807）。他のレコードについてプレビューを行うのであれば、他のレコードを選択して再度レイアウト計算を行い、プレビューを行う（0806）。プレビューモードでなく印刷時においては、印刷するレコード全てについて順にレイアウトの計算を行う。したがって、処理0805、0807は存在しない。全てのレコードについて印刷が終了した時点で終了する。

【0088】

< 動的レイアウト計算方法（詳細フロー） >

図9は動的レイアウト計算の詳細を示したフロー図である。また、図10はそのときのUI表示例を示した図である。

【 0 0 8 9 】

まず、レイアウトを計算するコンテナの集合を求める (0 9 0 1)。レイアウト計算は、関連付けられたコンテナを一つの集合として計算を行う。例えば図 1 1 を参照すると、ページ上に 4 つのコンテナがレイアウトされており、各コンテナに関連付けが設定されている。この場合、コンテナ A とコンテナ B、そしてコンテナ C とコンテナ D がリンクによって関連付けされている。したがって、コンテナ A ・ B が集合 1、コンテナ C ・ D が集合 2 となる。図 1 1 において、符号 1 1 0 1 はアンカー、符号 1 1 0 2 は固定された辺、符号 1 1 0 3 はコントローラ、符号 1 1 0 4 は可変の辺の変化方向を示している矢印、符号 1 1 0 5 は可変の辺、符号 1 1 0 6 はリンク、符号 1 1 0 7 はスライダを示している。

【 0 0 9 0 】

次に、ステップ 0 9 0 1 で求めたコンテナの集合から、レイアウトを計算するために一つを選択する (0 9 0 2)。そして、選択したコンテナの集合について、レイアウトの計算を行う。ここでは、レイアウトされるコンテナのサイズが、実際のコンテンツのサイズとできる限り差が少なくなるように、レイアウトの最適化を行う (0 9 0 3)。レイアウトの最適化は、動的にサイズを変化することが可能のように関連付けられたコンテナにおいて、それぞれに挿入されるコンテンツのサイズとレイアウトされるサイズとの差が、できる限り同じになるように行われる。レイアウトの最適化を行い、ルールに違反していた場合は、再度ルールを違反しないように計算をする (0 9 0 4)。ここで記述したルールとは、レイアウト作成時にユーザによって設定される制限であり、コンテナのサイズと位置、リンクの長さなどである。ルールを違反しないようにレイアウトが計算されたら、その集合のレイアウトは完成される。そして、ステップ 0 9 0 2 ~ ステップ 0 9 0 4 をページ上のすべての集合について施し、ページ全体のレイアウトを計算する (0 9 0 5)。

【 0 0 9 1 】

ステップ 0 9 0 4 において、解消できないレイアウトエラーが生じた場合には、本実施形態における R S S フィードを用いたエラー配信が実行される。たとえばレイアウト計算を 2 度行ってもエラーが解消できない場合には、解消不能なエラーが発生したものと判断される。この手順や説明は、図 1 9 等を参照して後で説明する。

【 0 0 9 2 】

図 1 0 はレイアウト計算時の UI 例である。図 1 0 (A) は、定義済みのテンプレートにあるレコードが挿入されレイアウトが決定されている状態を表している。オブジェクト 1 0 0 1 とオブジェクト 1 0 0 2 はアンカーである。オブジェクト 1 0 0 3 とオブジェクト 1 0 0 4 は固定された辺である。オブジェクト 1 0 0 5 は可変の辺、オブジェクト 1 0 0 6 は可変の辺の変化方向を示している矢印、オブジェクト 1 0 0 8 はリンクをそれぞれ示している。この状態において、テンプレートに挿入されるコンテンツデータのレコードを変更し、異なったサイズのコンテンツデータをコンテナに挿入する。図 1 0 (B) は図 1 0 (A) の状態に新しいコンテンツのサイズを重ねて示している。2 点鎖線 1 0 0 9 はそれぞれのコンテナに挿入される、新たなレコードのコンテンツのサイズを表している。そして、レイアウト計算が行われる。図 1 0 (C) はレイアウト計算された結果を示している。計算後の各コンテナのサイズは、実際挿入されるコンテンツのサイズと同等に差異があるように計算され、且つ前述したルールを違反しないように計算される。図 1 0 (C) で示されるように、図 1 0 (B) で示した挿入されるコンテンツサイズ (1 0 0 9) と計算後のコンテンツサイズ (1 0 1 0) は、双方において同等な差異がある。

【 0 0 9 3 】

< 可変リンクの設定 >

図 1 2 では一般的な可変リンクの状態を表している。図 4 と同様にアプリケーションウィンドウ 3 0 1 とツールバー 3 0 3 があり、ドキュメントテンプレート 3 0 9 上にコンテナ 1 2 0 3 とコンテナ 1 2 0 4 が存在する。それぞれのコンテナは典型的にアンカーアイコン 1 2 0 1、アンカーアイコン 1 2 0 2 と固定された辺 1 2 0 5、辺 1 2 0 6 から成り立つ。各エッジコンテナ 1 2 0 3 と 1 2 0 4 の間には可変サイズのリンク 1 2 0 9 があり、それぞれのコンテナを結んでいる。コンテナ 1 2 0 3 とコンテナ 1 2 0 4 の間にはリン

クが設定されているのでそれぞれの右辺 1 2 0 7 と左辺 1 2 0 8 は点線で表現されている。このため各コンテナにインジケータ 1 2 1 0、インジケータ 1 2 1 1 が表示され、これは辺 1 2 0 7 と辺 1 2 0 8 が可変であることを示している。

【 0 0 9 4 】

また図 1 4 はリンク 1 2 0 9 の情報がセットされているダイアログウィンドウ 1 4 0 1 の例である。このダイアログは典型的にタイトルバー 1 4 0 2、ツールボタン 1 4 0 3、ダイアログウィンドウの開閉を行うボタン 1 4 0 4、各種の情報をセットするエリア 1 4 0 9 で構成されている。このダイアログウィンドウではリンクタイプが可変 (1 4 0 7) あるいは固定 (1 4 0 6) の選択や、可変の場合、リンクの長さの最大値 (1 4 1 0) そして最小値 (1 4 1 2)、また現在の値 (1 4 1 1) が設定できる。

10

【 0 0 9 5 】

図 1 3 は前述のリンク設定方法を踏まえ、例えば図 1 5 のコンテナ A とコンテナ B 間に張られている固定サイズのリンク 1 5 0 3 の状態から可変サイズのリンク 1 2 0 9 を示している図 1 2 の状態までの操作遷移を表している。マウスによりリンク 1 5 0 3 をクリックし、選択する (1 3 0 2)。マウスの右クリックあるいはキーボードの特定のキーにより選択したリンク 1 5 0 3 のプロパティダイアログウィンドウ 1 4 0 1 を表示する (1 3 0 3)。この状態ではリンクサイズは可変ではなく固定であるため、Fixed Length 1 4 0 6 が Link Type 1 4 0 5 において選択されている。リンクを固定サイズから可変サイズに変更するために、Link Type 1 4 0 5 においてリンクサイズを可変に設定する Flexible Length 1 4 0 7 を選択する (1 3 0 4)。これにより Link Distance 1 4 0 8 内に配置されている Max. Distance 1 4 1 2、Min. Distance 1 4 1 0、Distance 1 4 1 1 が有効化され、数値の設定が可能となる。そこでリンクの可変サイズを設定するために、そのリンクの長さの最大値を Max. Distance 1 4 1 2、最小値を Min. Distance 1 4 1 0、現在の値を Distance 1 4 1 1 において設定する (1 3 0 5)。一般的なダイアログウィンドウ開閉ボタン 1 4 0 4 によって設定を適用すると図 1 2 のリンク 1 2 0 9 のような状態にリンクの UI 表示が変化する (1 3 0 6)。このダイアログウィンドウ 1 4 0 1 の設定情報はメモリに格納される。

20

【 0 0 9 6 】

図 1 5 は固定サイズのリンクを使用した場合のレイアウト結果を示している。レイアウト計算方法は前述したとおりに従って行われる。例えば図 1 2 においてコンテナ 1 2 0 3 とコンテナ 1 2 0 4 にそれぞれ違ったサイズのデータが挿入された時、それぞれのコンテナはデータの大きさに適した大きさに変化する。たとえば、コンテナ 1 2 0 3 は挿入されたイメージサイズの枠 1 5 0 4 (最適コンテナサイズ) に近づこうと右方向へ大きくなるように変化する。同様に、コンテナ 1 2 0 4 も挿入されたイメージサイズの枠 1 5 0 5 (最適コンテナサイズ) に近づこうと左方向へやや大きくサイズを変更しようとする。ここでコンテナ 1 2 0 3 とコンテナ 1 2 0 4 との間には固定サイズのリンクが設定されているとする。コンテナ 1 2 0 3 とコンテナ 1 2 0 4 はそれぞれアンカー 1 2 0 1 とアンカー 1 2 0 2 によってそれぞれ左辺 1 2 1 2 と右辺 1 2 1 3 は移動出来ない。そのため、コンテナの大きさをイメージに合わせようとする、変更後のコンテナサイズは、リンク長の制約を満足することができない。リンクサイズは固定されているためレイアウト計算時に優先的に計算される。そのためコンテナ 1 2 0 3 とコンテナ 1 2 0 4 のサイズがリンク長に合わせて変更されることになる。その結果、コンテナ 1 2 0 3 とコンテナ 1 2 0 4 はデータに合わせた最適なサイズを確保することが出来ず、最終的に図 1 5 のコンテナ 1 5 0 1 とコンテナ 1 5 0 2 のように最適なサイズ (枠 1 5 0 4、枠 1 5 0 5) よりも小さくなってしまう。リンク 1 5 0 3 のサイズは固定であるためコンテナ 1 5 0 1 とコンテナ 1 5 0 2 は最適サイズを達成していない。

30

40

【 0 0 9 7 】

図 1 6 はリンクを可変サイズにした場合を示している。この場合、上記の例でコンテナ 1 2 0 3 とコンテナ 1 2 0 4 の間には図 1 6 の通り可変サイズのリンクが設定されている

50

とする。コンテナ 1 2 0 3 とコンテナ 1 2 0 4 のサイズが変更される際に、リンクサイズが縮まることでコンテナ 1 2 0 3 とコンテナ 1 2 0 4 のサイズが上記の例より大きくなることできる。このため、挿入されるデータサイズに合わせた最適なコンテナサイズを達成できる。あるいは、より挿入データサイズ（最適サイズ）に近づけてコンテナの枠を計算することが出来る。その結果が、図 1 6 のコンテナ 1 6 0 1 とコンテナ 1 6 0 2 である。可変リンク 1 2 0 9 はレイアウト計算の結果、可変リンク 1 6 0 3 のサイズ状態となり、この場合コンテナ 1 6 0 1 とコンテナ 1 6 0 2 はそれぞれ最適なサイズ（データサイズに合った大きさ）になっている。

【 0 0 9 8 】

後述の通り、本実施形態では、図 1 5 のようにエラー（図 1 5 の例ではイメージオーバーフローエラー）が生じた場合に、そのエラーの発生と共にテンプレートデータとコンテンツデータとへのリンク情報を利用者（あるいは管理者）に配信する。それを見た利用者は、テンプレートを編集し、たとえば図 1 6 のようにコンテナ間のリンクを可変に変更することで、エラーが生じないテンプレートを作成できる。

【 0 0 9 9 】

< コンテナのプロパティ設定 >

図 1 7 はコンテナのプロパティを設定するダイアログである。プロパティダイアログウィンドウは典型的にキーボード 1 3 2 やマウス 2 0 3 によりコンテナを選択し、マウスの右クリックメニューの選択、あるいはキーボードからの A l t キー等により表示する。この実施形態ではコンテナのエッジを可変にするかどうか、そしてどのレイアウト方法を適用するか、またはそれに伴う設定に関してのみ、プロパティダイアログウィンドウ内に表示している。このプロパティダイアログウィンドウ 1 7 0 1 は典型的にタイトルバー 1 7 0 2、ツールボタン 1 7 0 3、プロパティダイアログウィンドウのオープン、クローズを行うボタン 1 7 0 4、各種の情報をセットするエリア 1 7 5 1 で構成されている。このプロパティダイアログウィンドウ 1 7 0 1 には、E d g e S t a t u s グループボックス 1 7 1 1 内に、コンテナのエッジ状態を可変し、コンテナ自体を可変設定にする選択オプション 1 7 1 2 ~ 1 7 1 5 がある。この選択オプションを O N にすると C o n t a i n e r S i z e グループボックス 1 7 2 1 内が E n a b l e される。この例の場合、E d g e S t a t u s グループボックス 1 7 1 1 において R i g h t （右辺）1 7 1 4 のみ F l e x i b l e （可変）にチェックが入っているため、このコンテナは横方向（実際は右方向）にしかサイズを変化させることが出来ない。そのため、C o n t a i n e r S i z e 1 7 2 1 において、幅サイズの項目 1 7 2 4 と 1 7 2 6 が E n a b l e となり、高さサイズの項目 1 7 2 8 と 1 7 3 0 は D i s a b l e のままとなっている。図形 1 7 2 2 はコンテナの幅サイズの指定、図形 1 7 2 3 はコンテナの高さサイズの指定を視覚的に補助するビットマップである。ボックス 1 7 2 4 （M i n . S i z e ）にはコンテナの幅の最小値が設定される。現在ここには値 8 0 が設定されている。ボックス 1 7 2 6 （M a x . S i z e ）にはコンテナの幅の最大値が設定される。現在ここには値 1 2 0 が設定されている。同様に 1 7 2 8 はコンテナの高さの最小値、1 7 2 9 はコンテナの基本幅、1 7 3 0 はコンテナの高さの最大値が設定されることがある。リストボックス 1 7 2 7 とリストボックス 1 7 3 1 ではコンテナサイズの単位が設定される。ここでは例として m m が表示されているが、i n c h や p i x e l 等も考えられる。

【 0 1 0 0 】

図 1 8 は上記のコンテナのプロパティ設定における操作によるテンプレート編集アプリケーションプログラム 1 2 1 の動作を示したフローチャートである。まずコンテナが選択され（1 8 0 2）、コンテナのプロパティを表示する（1 8 0 3）。コンテナエッジを可変に設定されているかどうかを各辺の項目に対して判断し（1 8 0 4）、可変に設定されている場合には、コンテナサイズの最大値・最小値のボックスを E n a b l e にする（1 8 0 5）。また同時にレイアウト方法の選択項目も E n a b l e にする（1 8 0 6）。そして O K ボタンなどによりコンテナのプロパティを閉じ、設定をコンテナに適用する（1 8 0 7）。

10

20

30

40

50

【 0 1 0 1 】

< テンプレートに含まれる情報の例 >

以上のように、テンプレートデータ（あるいはレイアウトデータ）には、以下のような情報が含まれる。

（ 1 ）コンテナの属性情報

コンテナの属性情報には、コンテナの位置、コンテナのサイズ、コンテナの辺あるいは頂点について、その位置が可変であるか固定であるかの別（辺の属性ともいう）、流しこまれるオブジェクトの種類（文字・画像等）やオブジェクトの属性（文字サイズ等）などを示す情報が含まれる。

（ 2 ）コンテナ間の情報

コンテナ間の情報には、リンクで示されるコンテナ間の距離や、その距離が固定か可変かの別を示す情報、あるいはコンテナ相互の位置関係を示す情報などが含まれる。

（ 3 ）固定された情報

固定された情報には、テンプレートに固定された文字列や画像などの情報が含まれる。

【 0 1 0 2 】

以上のように、テンプレートには、コンテナの属性情報と、コンテナ間の情報と、固定された情報とが含まれている。利用者は、前述したレイアウトエディタのグラフィカルユーザインターフェースを通して、それらの情報を編集することができる。

【 0 1 0 3 】

< テンプレート編集アプリケーションでのレイアウトチェック機能 >

図 19 はテンプレート編集アプリケーションプログラム 121 のレイアウトチェック機能の動作を示したフローチャートである。図 20 は、テンプレート編集アプリケーションプログラム 121 におけるメニューバー 2001 を示している。メニューリストには複数の項目があり、その中に「バリエブルデータ/レイアウトのチェック(C) . . . 」というメニュー項目 2002 が用意されている。このメニュー項目は、データのプレビュー時のみ有効である。すなわち、テンプレート編集アプリケーションプログラム 121 が、テンプレートにバリエブルデータを流し込み、レイアウトエンジン 127 によりレイアウトが行われた状態で選択可能となる。図 19 の手順は、そのバリエブルデータ/レイアウトのチェック項目 2002 が選択された際に実行される。

【 0 1 0 4 】

テンプレート編集者が、図 20 に示したようなマウスポインタ 2003 を用いて、テンプレート編集アプリケーションプログラム 121 のメニューバー 2001 から、メニュー項目「バリエブルデータ/レイアウトのチェック(C) . . . 」2002 を選択し、テンプレート編集アプリケーションプログラム 121 にレイアウトエラーのチェックを指示すると図 19 の処理が開始される。

【 0 1 0 5 】

図 19 において、まず、プレビューモードであるか判定され（S1901）、次いでレイアウトのチェックであるかが判定される（S1902）。もっとも項目 2002 はプレビューモードの時に限って表示されるから、これらのステップは省略することができる。

【 0 1 0 6 】

テンプレート編集アプリケーションプログラム 121 は、図 21 に示すような「バリエブルデータ/レイアウトのチェック」ダイアログ 2101 を表示する（S1903）。「バリエブルデータ/レイアウトのチェック」ダイアログ 2101 において、ユーザは[開始]ボタン 2102 を押下することができる。テンプレート編集アプリケーションプログラム 121 は、チェックの「開始」ボタン 2102 が指示されたか判定する（S1904）。開始が指示された場合には、テンプレート編集アプリケーションプログラム 121 は、レイアウトエンジン 127 を用いて、プリフライト検査を行いながら、再レイアウトを行う（S1905）。再レイアウト終了後（S1906）、プリフライト検査の結果を、図 22 のように、「バリエブルデータ/レイアウトのチェック」ダイアログ 2201 に表示する（S1907）。その表示によりレイアウトのエラーをユーザに示す。図 22 の[

10

20

30

40

50

該当数]リストボックス 2 2 0 3 には、チェック項目それぞれに該当するエラー数を一覧表示する。1 列目には該当するレコード番号を表示し、2 列目以降に、それぞれのチェック項目でのエラー数を表示する。なお、図 1 9 の手順で用いられるコンテンツデータは、ユーザにより別途指定されたデータファイルから読み込まれる。読み込まれたデータがテンプレートに従ってレイアウトされる。なおバリエーションデータとは、テンプレートに従ってレイアウトされるコンテンツデータのことであり、一つのテンプレートに適用されるバリエーションデータは 1 つのレコードを構成する。

【 0 1 0 7 】

さて表示されたチェック項目には、テキストオーバーフロー、イメージオーバーフロー、コンテナ、一般的なエラー等がある。テキストオーバーフローのチェックとは、テキストがコンテナ内に収まらないときに発生するテキストオーバーフローのチェックである。イメージオーバーフローのチェックとは、イメージがコンテナに入りきらないときに発生するイメージオーバーフローのチェックである。コンテナのチェックとは、データベースに値が存在しなかったため、値の入らない空のオブジェクトが存在するかどうかのチェックである。一般的なエラーのチェックとは、そのほかの一般的なエラーのチェックである。

10

【 0 1 0 8 】

図 2 2 の[エラー / 警告]リストボックス 2 2 0 4 には、[該当数]リストボックス 2 2 0 3 にて選択されたレコードのエラー / 警告詳細が一覧表示される。[エラー / 警告]リストボックス 2 2 0 4 の 1 列目にはエラーか警告かを表示するエラー / 警告アイコンが、2 列目にはエラーが発生したページ番号が、3 列目にはエラー / 警告の内容説明が表示される。

20

【 0 1 0 9 】

テンプレート編集アプリケーションプログラム 1 2 1 のユーザは、このレイアウトチェック機能の結果を用いながら、レイアウトエラーが発生しないテンプレート、もしくは、想定しないレイアウトを生成しないテンプレートの編集作業を行う。

【 0 1 1 0 】

< サーバでのレイアウトチェック実行 >

(コンテンツサーバにおけるレイアウトチェック処理)

図 2 3 はコンテンツサーバ 1 0 3 でのレイアウトチェック実行時の動作を示したフローチャートである。

30

【 0 1 1 1 】

図 2 3 において、コンテンツサーバ 1 0 3 は、WEBサーバ 1 0 6 から仲介された一般ユーザが利用する WEB ブラウザアプリケーション 1 2 5 からのリクエストを受信できる。そのリクエストを受信すると (S 2 3 0 1)、レイアウトサーバ 1 0 5 に、テンプレート指定、バリエーションデータ指定および実データ送信、ドキュメントフォーマット指定とともに、ドキュメント生成をリクエストする (S 2 3 0 2)。

【 0 1 1 2 】

その際、コンテンツサーバ 1 0 3 の設定ファイルにプリフライト結果の取得設定がなされている場合 (S 2 3 0 3)、コンテンツサーバ 1 0 3 は、レイアウトサーバ 1 0 5 に、レイアウト時にプリフライトを実行するよう追加リクエストする (S 2 3 0 4)。プリフライト結果の取得設定がされた設定ファイルとは具体的には以下のような内容を含む。

40

```
<?xml version="1.0" encoding="Shift_JIS" ?>
```

```
< layout-server >
```

```
< request >
```

```
< layout >
```

```
< option >
```

```
< proofing > 1 < /proofing > <!-- if 1, request the preflight -->
```

```
< /option >
```

```
< /layout >
```

50

</request>

</layout-server>。

【0113】

リクエスト後、コンテンツサーバ103は、レイアウトサーバ105からのレスポンスを待つ(S2305)。リクエストに対するレスポンスがレイアウトサーバ105から返ってきた際、コンテンツサーバ103は、レスポンスとして、生成ドキュメントを受け取る(S2306)。更に、レイアウト時のプリフライト実行リクエストがされている場合には(S2307)、プリフライト結果を受け取る(S2308)。

【0114】

コンテンツサーバ103は、受け取ったプリフライト結果をチェックする(S2309) 10

レイアウトエラーが発生している場合には(S2310)、そのレイアウトエラーを、リクエストを行った際のテンプレート指定値、バリアブルデータ指定値とともに、コンテンツサーバ103の記憶装置209に、レイアウトエラー履歴として保存する(S2311)。

【0115】

最後に、レスポンスとして受け取った生成ドキュメントを、リクエストを行ったWEBブラウザアプリケーション125に返すために、WEBサーバ106に送信し(S2312)、処理を終了する。

【0116】

20

(レイアウトサーバにおけるレイアウトチェック処理)

図24はレイアウトサーバ105でのレイアウトチェック実行時の動作を示したフローチャートである。図24において、レイアウトサーバ105は、コンテンツサーバ103などからレイアウト時にプリフライトを実行するよう追加リクエストを受け取ることができる。追加リクエストを受けた場合(S2401)、レイアウトサーバ105は、図8および図9に示したようなレイアウト計算を実行し、ドキュメントのページを生成する(S2402)。その際、プリフライトをページごとに行い(S2403)、コンテンツの溢れなどレイアウト上のエラーをページ毎にチェックする。

【0117】

レイアウトサーバ105は、レイアウトエラーを上記チェックで発見したか判定し(S2404)、発見した場合、そのエラーをページ単位で以下のようなデータ形式で保持する(S2405) 30

Pre-Print: 0x13127,

Print-Page#1: 0x13020,0x13022,0x13060,

Print-Page#2: ,

Print-Page#3: ,

Print-Page#4: 0x13020,0x13021,0x13022,0x13060,0x13121,0x13123,

Post-Print: 0x13128,0x13129。

【0118】

上記データ形式における各識別子(レイアウトチェックステップ識別子という)は、エラーが検出された処理ステップを示す。その意味は以下の通りである。 40

Pre-Print:印刷前チェック、

Print-Page#n (n:ページ番号):ページ印刷後チェック、

Post-Print:印刷完了後チェック。

【0119】

また、各識別子に対応するエラー番号の意味は図27に示す通りである。ここで、すべてのエラーコードが、レイアウトの編集を必要とするエラーとは限らない。図27に示すエラーの内、「0x13020」がテキストオーバーフローに、「0x13021」がイメージオーバーフローに相当する。また「0x13060」がコンテナエラーに相当する。

【0120】

50

レイアウトサーバ105は、ドキュメント生成が終了した際(S2406)、発見したレイアウトエラーの情報を、追加リクエストの送信元であるコンテンツサーバ103などクライアントに返し(S2407)、処理を終了する。

【0121】

<サーバでのレイアウトエラー履歴保存>

図25はコンテンツサーバ103でのレイアウトエラーの履歴保存時の動作を示したフローチャートである。図23で示したコンテンツサーバ103でのレイアウトチェック実行時の動作、および、図24で示したレイアウトサーバ105でのレイアウトチェック実行時の動作により、コンテンツサーバ103はレイアウトサーバ105からレイアウトエラーの情報を受け取る。この情報を検査して、リクエストを行ったドキュメントの生成時にレイアウトエラーが発生したか判定する(S2501)。レイアウトエラーが発生している場合、コンテンツサーバ103のデータベース119のレイアウトエラー履歴保存用テーブルに新たなレコードを作成する(S2502)。そのレコードに、発生したレイアウトエラーの情報、利用したテンプレートの識別子、およびコンテンツデータのレコード識別子を格納する(S2503)。

10

【0122】

次に、コンテンツサーバ103は、レイアウトエラー履歴保存用テーブルに保存したデータ(レコード)から、図28に示すような、"Template ID"フィールド2602のテンプレート識別子の情報を見出しにするタイプ(便宜的に「タイプ1」と呼ぶ)のRSSデータを構築する(S2504)。更に、図29に示すような、"Date and Time" フィールド2605のレイアウトエラー発生時間の情報を見出しにするタイプ(便宜的に「タイプ2」と呼ぶ)のRSSデータを構築する(S2505)。次に、コンテンツサーバ103は、生成した上記タイプ1、タイプ2の二つのRSSデータを、WEBサーバ106にアップロードし(S2506)、WEBブラウザアプリケーション122からアクセス可能なようにする。その後、レイアウトエラーの履歴保存の処理を終了する。

20

【0123】

図26は、レイアウトエラー履歴保存用テーブルの記述例を示す図である。図26において、レイアウトエラー履歴保存用テーブル2601には、1レコードに、"Template ID"フィールド2602、"Content IDs String"フィールド2603、"Layout Error String"フィールド2604、"Date and Time" フィールド2605の各情報が格納される。"Template ID"フィールド2602は文字列型のカラムである。"Contents Record IDs String"フィールド2603は数値配列型のカラムである。"Layout Error String"フィールド2604は文字列型のカラムである。"Date and Time" フィールド2605は日付/時刻型のカラムである。

30

【0124】

"Template ID"フィールド2602には、レイアウトエラーが発生したテンプレート識別子の文字列が格納される。例えば、レコード2611では、テンプレートの識別子"MITUMORI"が格納されている。この識別子は、コンテンツサーバ103によって別途管理され、ファイルサーバ104に保存される実際のドキュメントテンプレートのデータすなわちファイルに関連付けられている。

40

【0125】

"Contents Record IDs String"フィールド2603には、レイアウトエラーが発生したコンテンツデータのそれぞれのレコードの識別子の数値が、可変長配列の各要素に格納される。例えば、レイアウトエラー履歴保存テーブル2601のレコード2611では、コンテンツデータ管理テーブルのレコードの各識別子"200", "113", "31", "42938", "3187", "98", . . . が配列の各要素として格納される。図26では便宜上、コンマ区切りの文字列として示している。コンテンツデータ管理テーブルは、コンテンツサーバ103によって管理されるデータベース109

50

に設けられている。コンテンツデータ管理テーブルのレコードの識別子は、実際のコンテンツデータとの関連付けのために使用され、また、コンテンツデータの管理番号ともなっている。また、このコンテンツデータ管理テーブルのレコード識別子の並び順、すなわち、通常インデックス0から始まる配列への格納順は、コンテンツサーバ103からレイアウトサーバ105に送信され、レイアウトサーバ105において、テンプレートに流し込まれたレコードの順序と同じとなっている。

【0126】

"Layout Error String"フィールド2604には、レイアウトサーバ105から受け取ったレイアウトエラー情報の文字列が格納される。"Date and Time"フィールド2605には、レイアウトエラーをレイアウトサーバ105から受け取った日付と時刻の情報が、レイアウトエラー発生時間の情報として格納される。

10

【0127】

以上のようにして、発生したレイアウトエラーの要約を示す情報とともに、エラーに係るテンプレートデータ及びコンテンツデータへのリンク情報を有するRSSデータの形式で、エラーを要約したエラー要約情報が作成される。

【0128】

<RSSデータの作成(タイプ1)>

図25のステップS2504で生成されたタイプ1のRSSデータ"LayoutErrorList.rss"の例を図28に示した。本例では、RSS1.0での記述を用いているが、RSSの他のバージョンやAtomなど他のフィードフォーマットを用いても構わない。

20

【0129】

図28において、2801はXML文書の宣言部である。2802は、RDF名前空間の文書要素である。RDF要素には、RSS1.0の規約として、一つのchannel要素と、一つ以上のitem要素を含まなければならない。2803は、channel要素。2810、2814、2818はitem要素である。channel要素2803には、要約情報をフィードするこのチャンネルの基本情報が記述される。channel要素2803のabout属性には、このRSS自身のURIの情報が示される。channel要素2803のtitle要素2804には、このチャンネルのタイトル情報が記述される。同じくlink要素2805には、このチャンネルの要約対象のURI情報が記述される。同じくdescription要素2806には、このチャンネルの概要説明の情報が記述される。図28の例では、このチャンネルのURIはhttp://www.canon.com/iwBS/5DWF506/LayoutErrorList.rssである。このチャンネルのタイトルは"Recent Layout Error"である。このチャンネルの要約対象のURIはhttp://www.canon.com/iwBS/5DWF506/LayoutErrorList.htmlであり、このチャンネルの概要説明の情報は"Recent Layout Error List"となっている。

30

【0130】

items要素2807は、item要素群2810、2814、2818のインデックスを提供する。Seq要素2808の内容にリストしたli要素2809によって各更新情報URIを示す。item要素2810、2814、2818には、channel要素2803同様、title要素2811、2815、2819、link要素2812、2816、2820、description要素2813、2817、2821がそれぞれ含まれる。

40

【0131】

図30は、図29に示したタイプ1のRSSデータ"LayoutErrorList.rss"を構築する際の動作を示したフローチャートである。すなわち図30は図25のステップS2504の詳細を示す。

【0132】

図25で示したように、コンテンツサーバ103は、データベース119のレイアウトエラー履歴保存用テーブル2601に新たなレコードを作成する。そしてそのレコードに、発生したレイアウトエラーの情報、利用したテンプレートの識別子、およびコンテンツデータのレコード識別子を格納する。レイアウトエラー履歴保存用テーブル2601に新たなレコードを格納したことが判定されると(S3001)、title要素2811が"Erro

50

r Template - テンプレート識別子"という形式の値を持つitem要素 2 8 1 0 を生成する (S 3 0 0 2)。例えば、テンプレート識別子が"MITUMORI"であった際、図 2 8 に示すように、title要素 2 8 1 1 が"Template - MITUMORI"という値を持つitem要素を生成する。コンテンツサーバ 1 0 3 は、link要素 2 8 1 2 の値として、そのテンプレートのWEBサーバ 1 0 6 からのダウンロードURL "http://www.canon.com/iwBS/5DWF506/Template/MITUMORI.dle"を設定する (S 3 0 0 3)。description要素 2 8 1 3 に"Date and Time"フィールド 2 6 0 5 に格納したレイアウトエラー発生時間の情報"2005/10/18 13:48:23.785"を値として設定する (S 3 0 0 4)。更に、コンテンツサーバ 1 0 3 は、レイアウトエラーが発生したテンプレートの識別子と同じ識別子を有するレコードをレイアウトエラー履歴保存用テーブル 2 6 0 1 の最新レコードから順に検索する (S 3 0 0 5)。同じ識別子を有するレコードが検索で見つければ (S 3 0 0 6 - Y E S)、同一レコードカウンタを 1 加算する (S 3 0 0 7)。レコードカウンタの値が初期値の 1 であれば 2 となる。レコードカウンタの値が 4 であるか判定し (S 3 0 0 8)、4 でなければ上記手順によりdescription要素 2 8 1 3 の値にセットされた値"2005/10/18 13:48:23.785"の後に改行コード"¥n"を付加する (S 3 0 0 9)。その後、その見出したレコードの"Date and Time"フィールドの値を、"2005/10/18 13:48:23.785¥n 2005/10/18 12:38:44.008"のように、description要素 2 8 1 3 の値に付加する (S 3 0 1 0)。継続して次の検索を行う (S 3 0 0 5)。

【 0 1 3 3 】

ステップ S 3 0 0 8 で、レコード数のカウンタが 4 となった場合、見出したレコードの"Date and Time"フィールドの値のdescription要素 2 8 1 3 への付加をとりやめ、次の検索に移る。レコード数のカウンタが 4 となった場合とは、すなわち最新の 3 つのレコードの"Date and Time"フィールドがdescription要素 2 8 1 3 の値としてセットされた場合に相当する。

【 0 1 3 4 】

同じ識別子を有するレコードがこれ以上検索で見つからなかった (S 3 0 0 6 - N O) 段階で、文字列"... (" , ") "とともに、カウントされたレコード数を"... (12) "のようにdescription要素 2 8 1 3 の値に付加する (S 3 0 1 1)。

【 0 1 3 5 】

結果として、description要素 2 8 1 3 の値は、"2005/10/18 13:48:23.785¥n 2005/10/18 12:38:44.008¥n 2005/10/18 09:13:14.865... (12) "となる。

【 0 1 3 6 】

コンテンツサーバ 1 0 3 は、作成したitem要素 2 8 1 0 を、WEBサーバ 1 0 6 にアップロードしている既存のRSSデータ"LayoutErrorList.rss"があれば、そのデータのitem要素のリストの最後に付加する (S 3 0 1 2)。更に、そのRSSデータに重複する既存のitem要素があれば、それを削除する (S 3 0 1 3)。尚、既存のRSSデータ"LayoutErrorList.rss"がなければ、前記ステップ S 3 0 1 2 の前ステップで、新規に作成する。

【 0 1 3 7 】

< RSSデータの作成 (タイプ 2) >

図 2 9 に示したタイプ 2 のRSSデータ"LayoutErrorList-MITUMORI.rss"は以下のようなものである。本例では、RSS 1 . 0 での記述を用いているが、RSSの他のバージョンやAtomなど他のフィードフォーマットを用いても構わない。

【 0 1 3 8 】

図 2 9 において、2 9 0 1 はXML文書の宣言部である。2 9 0 2 は、RDF名前空間の文書要素である。RDF要素には、RSS 1 . 0 の規約として、一つのchannel要素と、一つ以上のitem要素を含まなければならない。2 9 0 3 は、channel要素。2 9 1 0、2 9 1 4、2 9 1 8 はitem要素である。

【 0 1 3 9 】

channel要素 2 9 0 3 には、要約情報をフィードするこのチャンネルの基本情報が記述

10

20

30

40

50

される。channel要素2903のabout属性には、このRSS自身のURIの情報が示される。channel要素2903のtitle要素2904には、このチャンネルのタイトル情報が記述される。同じくlink要素2905には、このチャンネルの要約対象のURI情報が記述される。同じくdescription要素2906には、このチャンネルの概要説明の情報が記述される。

図29の例では、このチャンネルのURIは<http://www.canon.com/iwBS/5DWF506/LayoutErrorList-MITUMORI.rss>である。このチャンネルのタイトルは"Recent Layout Error - MITUMORI"である。このチャンネルの要約対象のURIは<http://www.canon.com/iwBS/5DWF506/LayoutErrorList-MITUMORI.html>である。このチャンネルの概要説明の情報は"Recent Layout Error List - MITUMORI"となっている。

10

【0140】

items要素2907は、item要素群2910、2914、2918のインデクスを提供する。Seq要素2908の内容にリストしたli要素2909によって各更新情報URIを示す。item要素2910、2914、2918には、channel要素2903同様、title要素2911、2915、2919、link要素2912、2916、2920、description要素2913、2917、2921がそれぞれ含まれる。

【0141】

図31は、図29に示したタイプ2のRSSデータ"LayoutErrorList-MITUMORI.rss"を構築する際の動作を示したフローチャートである。図25で示したように、コンテンツサーバ103が、データベース119のレイアウトエラー履歴保存用テーブル2601に新たなレコードを作成する。そのレコードに、発生したレイアウトエラーの情報、利用したテンプレートの識別子、およびコンテンツデータのレコード識別子を格納する。その格納が行われると(S3101-YES)、title要素2911の値がレイアウトエラー発生時間の情報を持つitem要素2910を生成する(S3102)。例えば、テンプレート識別子"MITUMORI"のテンプレートを用いたレイアウト処理中にレイアウトエラーが時刻"2005/10/17 03:12:21.009"に発生したとする。その際、図29に示すように、title要素2911が"2005/10/17 03:12:21.009"という値を持つitem要素を生成する。

20

【0142】

コンテンツサーバ103は、link要素2912として、そのレイアウトエラーが発生した際のコンテンツをWEBサーバ106からダウンロードするURL"http://www.canon.com/iwBS/5DWF506/Data/MITUMORI-20051017_031221.009.zip"を値として設定する(S3103)。そして"LayoutErrorString"フィールド2604に格納したレイアウトエラー情報の文字列をメモリ206上に読み出し、まず前述したレイアウトチェックステップ識別子を取り除く(S3104)。レイアウトチェックステップ識別子は、エラーが検出された処理ステップを示し、Pre-Print, Print-Page#n, Post-Printがある。

30

【0143】

次に図27に示したレイアウトエラー番号およびその概要の対応づけを元にレイアウトエラー番号を"TEXT_OVERFLOW"のような概要を表すキーワードに置換する(S3105)。更に簡略化のため、レイアウトエラー番号が3つを超える場合、"...("、")"と共に、レイアウトエラー数を"...(8)"のように付加する。こうして得られたメモリ上の文字列"TEXT_OVERFLOW IMAGE_OVERFLOW MINMAX_CONT_SIZE ... (8)"を、description要素2913の値として設定する(S3106)。

40

【0144】

コンテンツサーバ103は、作成したitem要素2910をWEBサーバ106にアップロードしている既存のRSSデータ"LayoutErrorList-MITUMORI.rss"があれば、そのデータのitem要素のリストの先頭に付加する(S3107)。

【0145】

尚、既存のRSSデータ"LayoutErrorList-MITUMORI.rss"がなければ、ステップS3105の前ステップで、新規に作成する。また、本実施形態では、レイアウトエラーが発生

50

したタイミング毎にシンジケーション用データを作成することとしたが、一定時間毎に作成するようにしても構わない。

【 0 1 4 6 】

< W E B ブラウザからのシンジケーションデータへのアクセス >

図 3 2 は、管理クライアント 1 0 1 の W E B ブラウザ 1 2 2 で、 R S S フィードをダウンロードするための W E B ページ "<http://www.canon.com/iwBS/5DWF506/LayoutError-RSS.html>" を表示した画面である。

図 3 2 において、表示画面 3 2 0 1 に W E B ページが表示される。 W e b ブラウザに組み込まれた R S S リードにより表示されるリスト表示欄 3 2 0 2 (単に R S S リード 3 2 0 2 という) には R S S フィードが表示される。アドレスバー 3 2 0 3 には現在表示されているページの U R I が表示される。「戻る」ボタン 3 2 0 4 が押されると、現在ページの直前に表示されていたページが表示される。「進む」ボタン 3 2 0 5 が押されると、現在ページの直後に表示されていたページが表示される。ボタン 3 2 0 7、3 2 0 9、3 2 1 1、3 2 1 3、3 2 1 5、3 2 1 7 は、 R S S フィードダウンロードボタンであり、これらが押されると、該当する R S S データがダウンロードされる。ラベル 3 2 0 6、3 2 0 8、3 2 1 0、3 2 1 2、3 2 1 4、3 2 1 6 は各 R S S フィードダウンロードボタンのラベルである。バー 3 2 1 8 は、 R S S チャンネル切り替えバーである。

【 0 1 4 7 】

W E B ページの表示画面 3 2 0 1 において、 R S S フィードダウンロードボタン 3 2 0 7 が押下されると、図 3 0 のフローチャートで構築されたタイプ 1 の R S S データ "LayoutErrorList.rss" が、 W E B サーバ 1 0 6 から W E B ブラウザ 1 2 2 にダウンロードされる。 R S S の仕組みでダウンロードされる R S S データをシンジケーションデータと呼び、シンジケーションデータのダウンロードを R S S フィードと呼ぶ。シンジケーションデータを R S S フィードと呼ぶ場合もある。同様に、 R S S フィードダウンロードボタン 3 2 0 9 が押下されると、図 3 1 のフローチャートで構築されたタイプ 2 の R S S データ "LayoutErrorList-MITUMORI.rss" がシンジケーションデータとして W E B サーバ 1 0 6 からダウンロードされ、 R S S フィードされる。また、これらシンジケーションデータを識別するための U R I がリンクとして R S S チャンネル切り替え用プルダウンリストボックス 3 2 1 8 に記憶される。

【 0 1 4 8 】

W E B サーバ 1 0 6 から W E B ブラウザ 1 2 2 にダウンロードされた R S S フィードのデータは、 R S S リード 3 2 0 2 に渡され、 R S S リード 3 2 0 2 によって情報の表示が行われる。

【 0 1 4 9 】

図 3 3 は、 W E B ブラウザ 1 2 2 の R S S チャンネル切り替え用プルダウンリストボックス 3 2 1 8 に記憶されたリンクのリスト表示である。ユーザが、マウスポインタ 3 3 2 1 により、プルダウンボタン 3 3 1 9 を押下した際、プルダウンリストボックス 3 2 1 8 が展開され、リストボックス 3 3 2 0 が表示される。図において、リストボックス 3 3 2 0 には、シンジケーションデータの U R I リンクの識別子がリストされる。

【 0 1 5 0 】

図 3 3 の、 R S S チャンネル切り替え用プルダウンリストボックス 3 2 1 8 には、チャンネルすなわち R S S フォードの識別子がいくつかリスト表示された例を示している。図 3 3 において、識別子 "All Layout Error" 3 3 2 0 a は、 W E B サーバ 1 0 6 上のシンジケーションデータ "LayoutErrorList.rss" への U R I リンクである。また識別子 "MITUMORI Template Layout Error" 3 3 2 0 b は W E B サーバ 1 0 6 上のシンジケーションデータ "LayoutErrorList-MITUMORI.rss" への U R I リンクである。そのほかの識別子も同様にサーバのシンジケーションデータへのリンクが埋め込まれており、そこを選択指示することでリンク先のデータがダウンロードされる。識別子は、たとえばシンジケーションデータの一部としてダウンロードされる。

【 0 1 5 1 】

< クライアントでのレイアウトエラー履歴シンジケーションの利用 >

図 3 4 は管理クライアント 1 0 1 でのレイアウトエラー履歴シンジケーション利用時のテンプレートダウンロード動作例を示したフローチャートである。図 3 4 は、管理クライアント 1 0 1 のユーザが W E B ブラウザ 1 2 2 の R S S チャンネル切り替え用プルダウンリストボックス 3 2 1 8 から、タイプ 1 の R S S データである "All Layout Error" のフィールドを選んだ際の処理の例を示す。そこでまずステップ S 3 4 0 1 では、タイプ 1 の R S S データが選択されたか判定する。タイプ 1 であれば、W E B ブラウザ 1 2 2 の R S S リーダ 3 2 0 2 は、W E B サーバ 1 0 6 に H T T P の取得リクエストを送信し (S 3 4 0 2)、W E B サーバ 1 0 6 からシンジケーションデータ "LayoutErrorList.rss" のレスポンスを受信する (S 3 4 0 3)。シンジケーションデータである R S S フィードのレスポンスデータを受け取った R S S リーダ 3 2 0 2 は、R S S データの表示を行う (S 3 4 0 4)。

10

【 0 1 5 2 】

図 3 5 は、管理クライアント 1 0 1 の W e b ブラウザ 1 2 2 の R S S リーダ 3 2 0 2 でタイプ 1 の R S S フィードのデータ "LayoutErrorList.rss" を表示した画面である。文字列 3 5 3 1、3 5 3 3、3 5 3 5、3 5 3 7 はハイパーリンクである。各文字列は、発生したエラーに関連するテンプレートを示している。図 3 5 において、マウスポインタ 3 3 2 1 を用いてユーザによりハイパーリンクのいずれかが選択された際 (S 3 4 0 5 - Y E S)、R S S リーダ 3 2 0 2 は、その U R I のリソースの取得を、W e b ブラウザ 1 2 2 を介して、W E B サーバ 1 0 6 にリクエストする (S 3 4 0 6)。W e b ブラウザ 1 2 2 はハイパーリンクのリンク先であるテンプレート (テンプレートデータ) をレスポンスとして W E B サーバ 1 0 6 から受け取る。リンク先のテンプレートを受け取ると (S 3 4 0 7 - Y E S)、W e b ブラウザ 1 2 2 は、そのテンプレートを開くためのテンプレート編集アプリケーション 1 2 1 を起動する (S 3 4 0 8)。そのテンプレートをテンプレート編集アプリケーション 1 2 1 に渡し、そのテンプレートを開く (S 3 4 0 9)。

20

【 0 1 5 3 】

図 3 6 は、管理クライアント 1 0 1 の W e b ブラウザ 1 2 2 にインライン表示されたテンプレート編集アプリケーション 1 2 1 でのテンプレートの表示画面の例である。図において、テンプレート表示・編集用ペイン 3 6 4 2 には、テンプレート "MITUMORI.dle" が表示されており、ユーザにより編集可能となっている。図 3 6 の例ではテンプレート "MITUMORI.dle" に対し流し込むコンテンツデータが未指定の状態なので、プレビュー表示用ペイン 3 6 4 1 には、空のページが表示されている。なおテンプレートの編集は、本例ではペイン 3 6 4 2 内で行われる。しかし、テンプレート編集アプリケーション 1 2 1 により提供される編集用ウィンドウで行っても良い。

30

【 0 1 5 4 】

図 3 7 は管理クライアント 1 0 1 でのレイアウトエラー履歴シンジケーション利用時のコンテンツダウンロード動作を示したフローチャートである。図 3 7 は、管理クライアント 1 0 1 のユーザにより、W E B ブラウザ 1 2 2 の R S S チャンネル切り替え用プルダウンリストボックス 3 2 1 8 から、現在 W e b ブラウザ 1 2 2 にインライン表示されたテンプレート編集アプリケーションで編集集中のテンプレートに対応する、タイプ 2 の R S S データ "MITUMORI Template Layout Error" のフィールドが選ばれた際の例を示す。そこでまずステップ S 3 7 0 1 では、タイプ 2 の R S S データが選択されたか判定する。タイプ 2 のフィールドが選ばれた際 (S 3 7 0 1 - Y E S)、W E B ブラウザ 1 2 2 の R S S リーダ 3 2 0 2 は、W E B サーバ 1 0 6 に H T T P の取得リクエストを送信する (S 3 7 0 2)。リクエストに応じて、W E B サーバ 1 0 6 からシンジケーションデータのレスポンス "LayoutErrorList-MITUMORI.rss" を受信する (S 3 7 0 3)。シンジケーションデータである R S S フィードのレスポンスデータを受け取った R S S リーダ 3 2 0 2 は、R S S データの表示を行う (S 3 7 0 4)。

40

【 0 1 5 5 】

50

図38は、管理クライアント101のWebブラウザ122のRSSリーダー3202でタイプ2のRSSフィードのデータ"LayoutErrorList-MITUMORI.rss"を、ステップS2704において表示した画面である。文字列3831、3833、3835、3837はハイパーリンクである。各文字列は、"MITUMORI Template Layout Error"で表されたテンプレートに関するエラー情報に対応している。図38において、ユーザは、文字列(ハイパーリンク)3831、3833、3835、3837のいずれかを選択することができる。

【0156】

そこで図38のステップS3705では、マウスポインタ3321を用いてユーザによりハイパーリンクのいずれかが選択されたか判定する。ハイパーリンクが選択された際(S3705-YES)、RSSリーダー3202は、そのURIのリソースの取得を、Webブラウザ122を介して、WEBサーバ106にリクエストする(S3706)。Webブラウザ122がハイパーリンクのリンク先であるコンテンツデータをレスポンスとしてWEBサーバ106から受け取る。リンク先のコンテンツデータを受け取ると(S3707-YES)、Webブラウザ122は、そのコンテンツをテンプレート編集アプリケーション121に渡す(S3708)。テンプレート編集アプリケーション121は、そのコンテンツをテンプレートに流し込みプレビューを生成する(S3709)。すなわち、テンプレート編集アプリケーション121は、そのコンテンツデータをテンプレートに従ってページ内に配置し、当該ページのプレビュー画像を生成する。

【0157】

図39は、Webブラウザ122のテンプレート編集アプリケーション121でのプレビュー表示画面である。プレビュー表示用ペイン3941には、テンプレート"MITUMORI.dle"にコンテンツデータが流し込まれ、レイアウトされた結果がプレビュー表示されている。

【0158】

図40は、ユーザによる管理クライアント101のWebブラウザ122のRSSリーダー3202およびテンプレート編集アプリケーション121を用いた、あるテンプレートのレイアウトエラー修正およびサーバへのアップロードの操作フローである。すなわち図40は、ユーザによる判断と操作およびその操作に応じた処理内容を示す。ユーザは、管理クライアント101のWebブラウザ122のRSSリーダー3202で、"MITUMORI Template Layout Error"のハイパーリンクの一つをクリックできる。すなわち、"MITUMORI Template Layout Error"というタイトルで示されるエラーのリストの中から、所望のエラーに関する情報を選択してクリックできる。タイトルは、エラーに関連するテンプレートあるいは全てのエラーに関連づけて付与されている。

【0159】

ハイパーリンクがクリックされると(S4001)、編集中的テンプレート"MITUMORI.dle"でエラーが発生したコンテンツをダウンロードし、コンテンツをテンプレートに従って配置して、レイアウト結果のプレビューを得る。ユーザは、テンプレート編集アプリケーション121のメニューバー2001からメニュー項目"バリアブルデータ/レイアウトのチェック(C)..."2002を選択できる。レイアウトチェックが選択されると、バリアブルデータ/レイアウトのチェックを開始し、そのチェック結果を得る(S4002)。ユーザは、「バリアブルデータ/レイアウトのチェック」ダイアログ2201に表示されるチェック結果がユーザにとって許容出来ないエラーか否か判定する。許容できない場合にはテンプレートの再編集作業を指示する。再編集作業が開始されると(S4003-YES)、利用者は、エラーを生じたレイアウトの結果を見ながら、テンプレート上のコンテナのサイズや可変設定に変更を加えるなど、テンプレートの再編集を行う。その編集作業に従ってテンプレートは変更される(S4004)。更に、バリアブルデータ/レイアウトのチェックを行い、エラーが発生しない状態またはエラーが許容できる状態まで変更を加える。

【0160】

ユーザは、エラーが発生しない状態またはエラーが許容できる状態まで変更を加えた後、再編集作業を終了しない場合、ユーザは、管理クライアント101のWebブラウザ122のRSSリーダー3202で、ハイパーリンクの別の一つをクリックする(S4003-NO, S4006-YES)。それにより別のコンテンツデータがダウンロードされる。こうしてレイアウトチェックおよびテンプレートの再編集操作を繰り返す。最終的に、ユーザが、編集作業を終えた場合、保存指示をすることで(S4005-YES)、編集したテンプレートは、サーバへアップロードされる(S4007)。保存は、テンプレート編集アプリケーション121のメニューバー2001から利用者がメニュー項目"保存(S)..."を選択することにより実行される。

【0161】

10

また、利用者は、RSSチャンネル切り替え用プルダウンリストボックス3218から"All Layout Error"を選択し、RSSリーダーに表示される"LayoutErrorList.rss"データから、"PRINTER Template Layout Error"など別のハイパーリンクをクリックすることもできる。この場合、その操作に応じて"Printer.dle"など別のテンプレートがダウンロードされる。そのテンプレートは、テンプレート編集アプリケーション121で開かれる。さらに、利用者がRSSチャンネル切り替え用プルダウンリストボックス3218から"PRINTER Template Layout Error"などを選択すると、その操作に応じてコンテンツがダウンロードされる。そして、別のテンプレートの修正作業に移る。

【0162】

図41は、図40のステップS4002の操作結果に該当する、Webブラウザ122のテンプレート編集アプリケーション121での「バリエーションデータ/レイアウトのチェック」ダイアログ2201の表示画面である。

20

【0163】

図42は、図40のステップS4004の操作が行われる、Webブラウザ122のテンプレート編集アプリケーション121でのコンテナのプロパティ変更中の表示画面である。

【0164】

図43は、図40のステップS4006 S4001での操作が行われる、Webブラウザ122のテンプレート編集アプリケーション121での別のコンテンツのプレビュー表示画面である。

30

【0165】

図44は、図40のステップS4001からS4006の操作の概念図である。図44において、表示欄3202からハイパーリンクが付された文字列が選択されると(STEP1)、リンクにより指し示される宛先のデータがダウンロードされる(STEP2)。そのデータを用いて試験的にレイアウトが行われ、必要に応じて編集される(STEP3)。

尚、管理クライアント101のWebブラウザ122のRSSリーダー3202での"LayoutErrorList.rss"や"LayoutErrorList-MITUMORI.rss"などのRSSデータの表示中に、表示中のRSSデータがサーバ側で更新されることもあり得る。たとえば、図25に示したコンテンツサーバ103でのレイアウトエラーの履歴保存時の動作により、表示中のRSSデータがサーバ側で更新されることがある。その場合、RSSデータの特性上、新たなRSSデータがWEBサーバ106から送信され、RSSリーダー3202により、その新たな送信データでデータ表示が更新される。

40

【0166】

<サーバでのクライアントへのレイアウトエラーコンテンツ送信>

図45は、図37での管理クライアント101でのレイアウトエラー履歴シンジケーション利用時のコンテンツダウンロード動作に対応する、WEBサーバ106のコンテンツ送信動作を示したフローチャートである。

【0167】

50

WEBサーバ106は、"http://www.canon.com/iwBS/5DWF506/Data/MITUMORI - 20051017 031221.009.zip"などのコンテンツダウンロードURLに対する取得リクエストをクライアントとから受け取ることができる。そのリクエストを受け取った際(S4501 - YES)、該当コンテンツの作成とアップロードをコンテンツサーバ103に依頼する(S4502)。WEBサーバ106は、コンテンツサーバ103からレスポンスとして、該当のコンテンツファイルのWEBサーバ106へのアップロードを受けた際(S4503)、該当ファイルのデータをクライアントに送信し(S4504)、処理を終了する。

【0168】

図46は、WEBサーバ106からコンテンツ作成とアップロードのリクエストを受け取った際のコンテンツサーバ103の動作を示したフローチャートである。コンテンツサーバ103は、WEBサーバ106から、コンテンツダウンロードURLのデータ"http://www.canon.com/iwBS/5DWF506/Data/MITUMORI - 20051017 031221.009.zip"と共に、コンテンツ作成とアップロードのリクエストを受け取る。リクエストを受け取った際(S4601 - YES)、そのURL文字列のデータからテンプレート識別子"MITUMORI"、およびレイアウトエラー発生時間の情報"2005/10/17 03:12:21.009"を抽出する(S4602)。

【0169】

抽出したテンプレート識別子"MITUMORI"のデータを下に、レイアウトエラー履歴保存用テーブル2601から"Template ID"フィールド2602の値が合致するレコードを検索する(S4603)。合致するものがあれば(S4604 - YES)、更に、そのレコードの"Date and Time"フィールドの日付/時刻情報が、抽出したレイアウトエラー発生時間の情報"2005/10/17 03:12:21.009"と合致するかどうか照合する(S4605)。合致しなければ(S4606 - NO)、更に、"Template

ID"フィールド2602の値が合致する別のレコードを検索し、"Template ID"フィールドおよび"Date and Time"フィールドの値が両方とも合致するレコードを検索する。もし、合致するレコードがない場合(S4604 - NO)、エラーをWEBサーバ106に返す(S4612)。両方とも合致するレコードが見つかった場合(S4606 - YES)、そのレコードの"Contents Record IDs String"フィールド2603の値、すなわちコンテンツ識別子の配列を参照する(S4607)。その数値配列の各要素の値、たとえば"200"、"113"、"31"、"42938"、"3187"、"98"、...という各識別子に該当するコンテンツのレコードデータを、データベース109のコンテンツデータ管理テーブルから順に取得する(S4608)。コンテンツサーバ103は、コンテンツデータを取得した順に一つのデータファイル"11285.csv"に、その識別子を付加的に保存する(S4609)。最終的に生成されたデータファイルとそのデータファイルから参照されるコンテンツ群を一つのアーカイブファイル"MITUMORI - 20051017 031221.009.zip"にまとめる(S4610)。生成したアーカイブファイルを、リクエストの発行元であるWEBサーバ106にアップロードし(S4611)、処理を終了する。

【0170】

尚、生成されたデータファイルとそのデータファイルから参照されるコンテンツ群は、該当するコンテンツ群がそのデータファイルに設定された順に基づき、該当テンプレートに流し込まれた場合、レイアウト時のプリフライトで、レイアウトエラー履歴保存テーブル2601の該当レコードに保存されたのと同じ結果をもたらすことになる。

【0171】

また、生成されたデータファイル"11285.csv"は図47のファイル4701のような形式である。コンテンツデータ管理テーブルのレコードのフィールドデータが文字列の場合、その文字列がデータファイルに保存される。フィールドデータがイメージファイル名の場合、そのイメージファイルへの参照がデータファイルに保存される。これは他のデータファイル4702についても同様である。また、上記例のアーカイブファイル"MITUMORI - 20051017 031221.009.zip"は、一般的なPK-ZIP形式のファイルである。

【 0 1 7 2 】

以上のように、本実施形態では、いわゆる自動レイアウトあるいはバリアブル印刷時にレイアウトサーバで生じるエラーを、RSSフィードの技術を用いてクライアントに配信することができる。このため、クライアントでは、エラーが生じたことおよび関連するデータへのリンク情報を、ユーザが常時監視し続けるなしに取得できる。また、関連するテンプレートデータおよびコンテンツデータへのリンク情報を得ているために、エラーを再現する試みも容易に行える。そして、行った試みにより生じたエラーを参照しつつ、テンプレートを、エラーが生じないように編集することができる。編集したテンプレートは再びサーバにアップロードして、バリアブル印刷を再度実行できる。テンプレートの再編集は、プリフライト検査時に行うことが望ましいが、本番運用時でも行うことができる。

10

【 0 1 7 3 】

〔 他の実施形態 〕

なおRSSフィードによるシンジケーションは、一定の形式を備えるシンジケーション用のデータ(RSSデータ)をサーバ(たとえばウェブサーバ)に用意しておく。そしてそのRSSデータの置き場所の情報を知得しているクライアント(RSSリーダ)が、所定のイベントの発生時に、サーバにアクセスして、現在取得しているデータよりも新しい情報があればそれを取得する。所定のイベントには、たとえばRSSリーダの起動時や所定時間経過時(即ち定期的)などが含まれる。すなわちデータのシンジケーションとは、サーバからクライアントへのデータの配信や配給を指す。もちろんシンジケーションは、現在実用化されているRSS 1.0のみならず、すべての現在有効な全てのバージョンのRSSや将来実用化されるであろうバージョンのRSSにより実現することもできる。またatomを用いたり、それに類する(すなわちシンジケーションのために利用できる)XMLメタデータの配信により実現できる。

20

【 0 1 7 4 】

また、RSSフィードを用いたエラーの監視は、自動レイアウトあるいはバリアブル印刷に限らない。たとえばウェブサービスでは、アプリケーションサーバによりクライアントから要求された処理が行われる。この処理時のエラー等、処理の状態を示す状態情報の配信のために本実施形態を適用し、エラーの発生及びエラーに係るデータへのリンク情報を、RSSフィードとして、管理用クライアント端末に送信する。そして本実施形態と同様に、管理用クライアントでエラーを再現し、エラーの原因となったプログラムやデータの編集を行って、再度処理を実行する。

30

【 0 1 7 5 】

このようにして、管理者が、遠隔のサーバで生じたエラーに関する情報を知得し、遠隔のクライアントにおいてエラー解消のための対策を講じることができる。

【 0 1 7 6 】

なお本発明は、複数の機器(例えばホストコンピュータ、インタフェイス機器、リーダ、プリンタなど)から構成されるシステムに適用しても、一つの機器からなる装置(例えば、複写機、ファクシミリ装置など)に適用してもよい。また本発明の目的は、前述の実施形態の機能を実現するプログラムコードを記録した記録媒体を、システムあるいは装置に供給し、そのシステムあるいは装置のコンピュータが記憶媒体に格納されたプログラムコードを読み出し実行することによっても達成される。この場合、記憶媒体から読み出されたプログラムコード自体が前述した実施形態の機能を実現することになり、そのプログラムコード自体およびプログラムコードを記憶した記憶媒体は本発明を構成することになる。

40

【 0 1 7 7 】

また、本発明には、プログラムコードの指示に基づき、コンピュータ上で稼働しているオペレーティングシステム(OS)などが実際の処理の一部または全部を行い、その処理によって前述した実施形態の機能が実現される場合も含まれる。さらに、記憶媒体から読み出されたプログラムコードが、コンピュータに挿入された機能拡張カ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込まれた場合についても、本発明は適用

50

される。その場合、書き込まれたプログラムコードの指示に基づき、その機能拡張カ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】

【 0 1 7 8 】

【図 1】本発明のコンピュータシステム構成図である。

【図 2】本発明のコンピュータモジュールの概略図である。

【図 3】本発明のメニューバー、ツールバー、ワークエリア、フローティングパレットを含む典型的なアプリケーションのメインウインドウの図である。

【図 4】本発明のコンテナ間のリンク、アンカーやスライダーを持つ典型的なコンテナの一面をスクリーン、ツール、アイコンの図である。

10

【図 5】本発明の典型的なコンテナルールの図である。

【図 6】本発明のリンクを作成するフローの図である。

【図 7】リンク作成時の UI 例の図である。

【図 8】本発明におけるレイアウト計算の全体フローの図である。

【図 9】本発明におけるレイアウト計算の詳細フローの図である。

【図 10】本発明における図 9 のフローに対応する UI 例の図である。

【図 11】本発明のレイアウト計算時におけるコンテナの集合について説明した図である。

。

【図 12】本発明の典型的な可変リンクによるコンテナ配置図である。

20

【図 13】本発明の図 12 の状態までの操作遷移を表している図である。

【図 14】本発明の典型的なリンクの設定ダイアログウインドウの図である。

【図 15】本発明の固定リンクによるレイアウト結果の図である。

【図 16】本発明の可変リンクによるレイアウト結果の図である。

【図 17】本発明のコンテナのプロパティを設定するダイアログの図である。

【図 18】本発明のコンテナのプロパティ設定操作におけるレイアウトアプリケーションプログラムの動作フローの図である。

【図 19】本発明のレイアウト編集アプリケーションプログラムのレイアウトチェック機能の動作フロー図である。

【図 20】本発明のテンプレート編集アプリケーションプログラムのメニューバーの図である。

30

【図 21】本発明のテンプレート編集アプリケーションプログラムの「バリエブルデータ / レイアウトのチェック」ダイアログの図である。

【図 22】本発明のテンプレート編集アプリケーションプログラムの「バリエブルデータ / レイアウトのチェック」ダイアログのプリフライト結果表示の図である。

【図 23】本発明のコンテンツサーバのレイアウトチェック実行時の動作フローの図である。

【図 24】本発明のレイアウトサーバのレイアウトチェック実行時の動作フローの図である。

【図 25】本発明のコンテンツサーバのレイアウトエラーの履歴保存時の動作フローの図である。

40

【図 26】本発明のレイアウトエラー履歴保存用テーブルの記述例の図である。

【図 27】エラー番号と対応するエラー概要文字列とを示す図である。

【図 28】本発明の「タイプ 1」の R S S データの記述例の図である。

【図 29】本発明の「タイプ 2」の R S S データの記述例の図である。

【図 30】本発明のコンテンツサーバのタイプ 1 の R S S データ構築の動作フローの図である。

【図 31】本発明のコンテンツサーバのタイプ 2 の R S S データ構築の動作フローの図である。

【図 32】本発明の R S S フィールドダウンロードページの表示画面の図である。

50

【図 3 3】本発明の W E B ブラウザの R S S チャンネル切り替え用プルダウンリストボックスのリスト表示画面の図である。

【図 3 4】本発明の管理クライアントのレイアウトエラー履歴シンジケーション利用時のテンプレートダウンロードの動作フローの図である。

【図 3 5】本発明の管理クライアントの R S S リーダでのタイプ 1 の R S S フィードの表示画面の図である。

【図 3 6】本発明の管理クライアント W e b ブラウザにインライン表示されたテンプレート編集アプリケーションのテンプレート表示画面の図である。

【図 3 7】本発明の管理クライアントのレイアウトエラー履歴シンジケーション利用時のコンテンツダウンロード動作フローの図である。

【図 3 8】本発明の管理クライアントの R S S リーダでのタイプ 2 の R S S フィードの表示画面の図である。

【図 3 9】本発明の W e b ブラウザのテンプレート編集アプリケーションでのプレビュー表示画面の図である。

【図 4 0】本発明の管理クライアントの R S S リーダおよびテンプレート編集アプリケーションを用いたテンプレートのレイアウトエラー修正およびサーバへのアップロードの操作フローの図である。

【図 4 1】本発明の W e b ブラウザのテンプレート編集アプリケーションでの「バリエーションデータ/レイアウトのチェック」ダイアログの表示画面の図である。

【図 4 2】本発明の W e b ブラウザのテンプレート編集アプリケーションでのコンテナのプロパティ変更中の表示画面の図である。

【図 4 3】本発明の W e b ブラウザのテンプレート編集アプリケーションでの別コンテンツのプレビュー表示画面の図である。

【図 4 4】本発明の W e b ブラウザのテンプレート編集アプリケーションでの操作の概念図である。

【図 4 5】本発明の W E B サーバのコンテンツ送信動作フローの図である。

【図 4 6】本発明のコンテンツサーバのコンテンツ作成とアップロードのリクエストを受け取った際の動作フローの図である。

【図 4 7】本発明のコンテンツサーバの生成データファイルの記述例の図である。

【符号の説明】

【 0 1 7 9 】

1 0 1 管理クライアントコンピュータ

1 0 2 クライアントコンピュータ

1 0 3 コンテンツサーバ

1 0 4 ファイルサーバ

1 0 5 レイアウトサーバ

1 0 6 W E B サーバ

1 0 7 ネットワーク

1 1 9 データベース

1 2 1 テンプレート編集アプリケーションプログラム

1 2 2 管理クライアントコンピュータの W E B ブラウザプログラム

1 2 5 クライアントコンピュータの W E B ブラウザプログラム

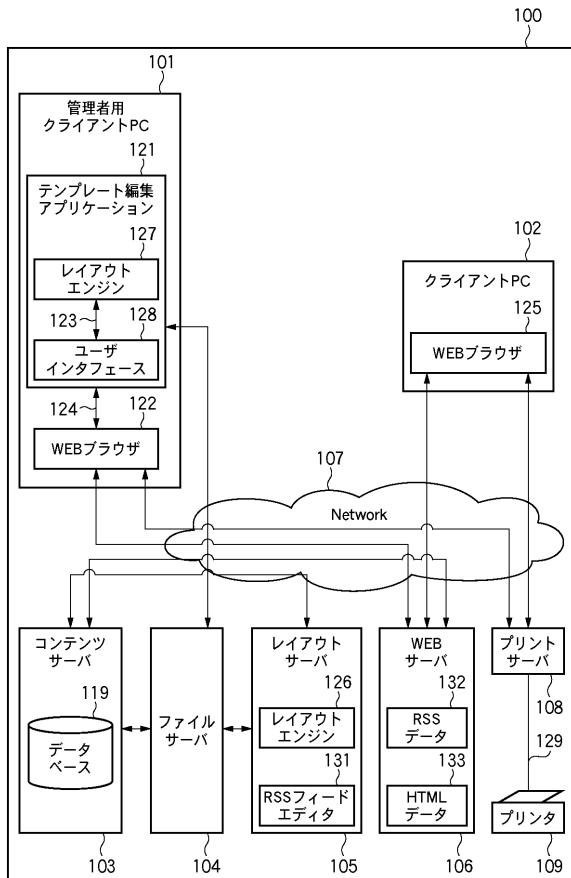
10

20

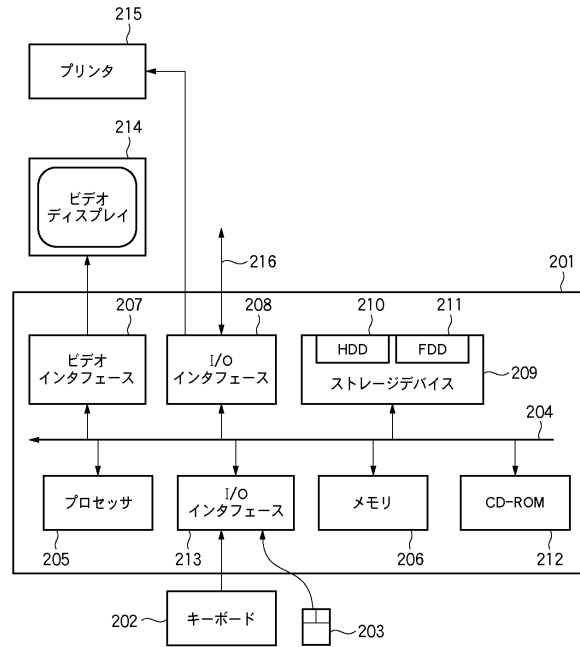
30

40

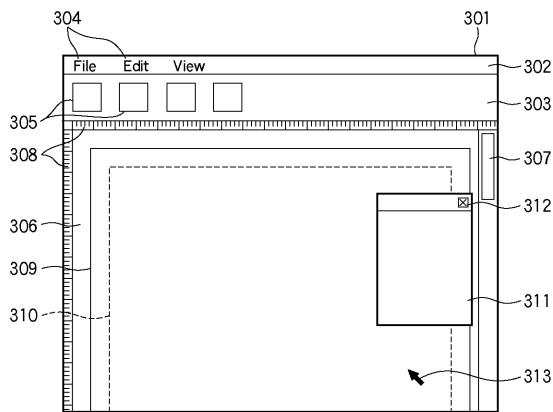
【図 1】



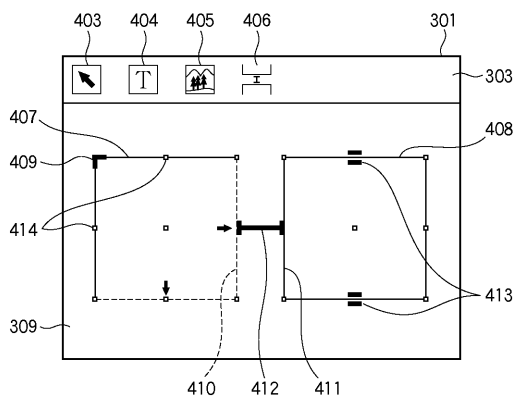
【図 2】



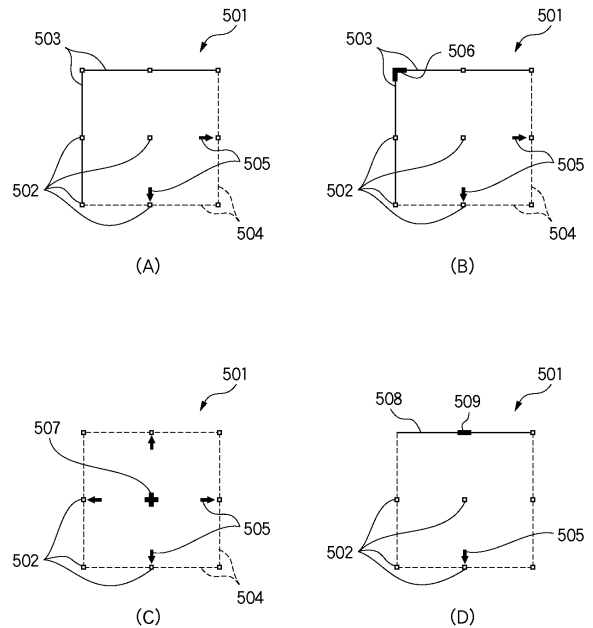
【図 3】



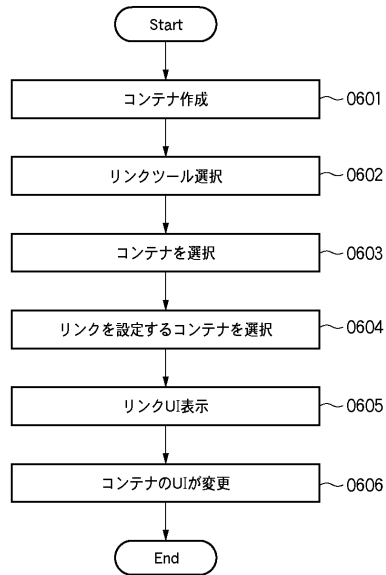
【図 4】



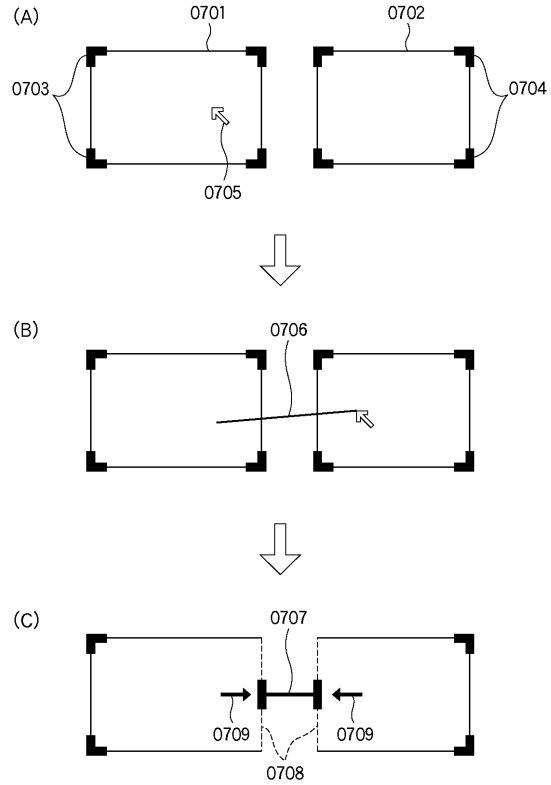
【図 5】



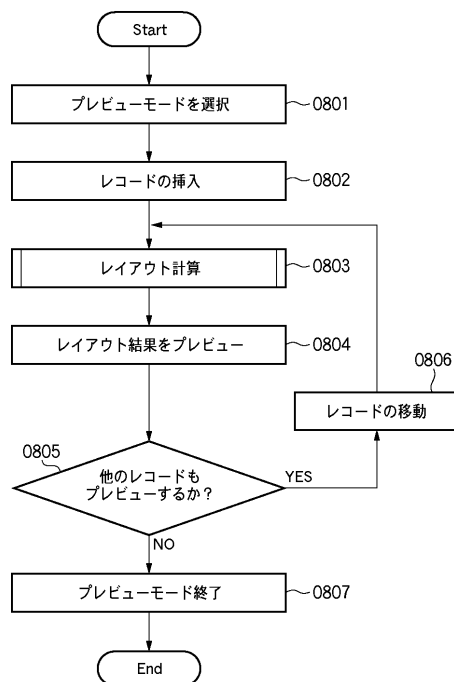
【図 6】



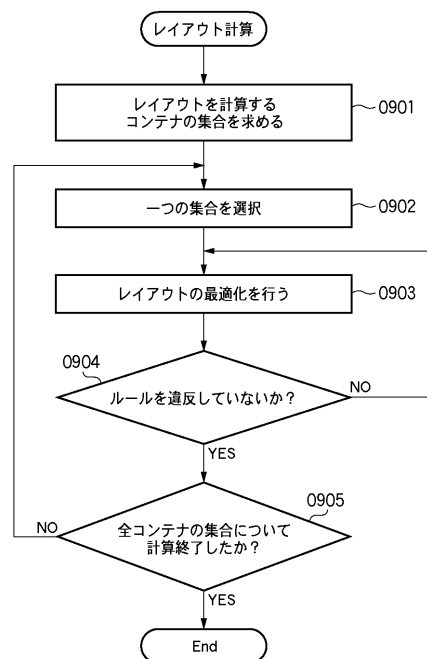
【図 7】



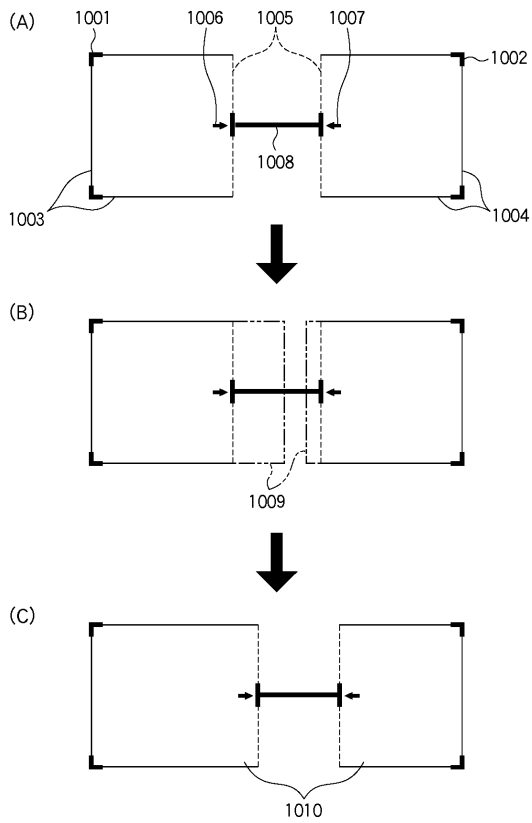
【図 8】



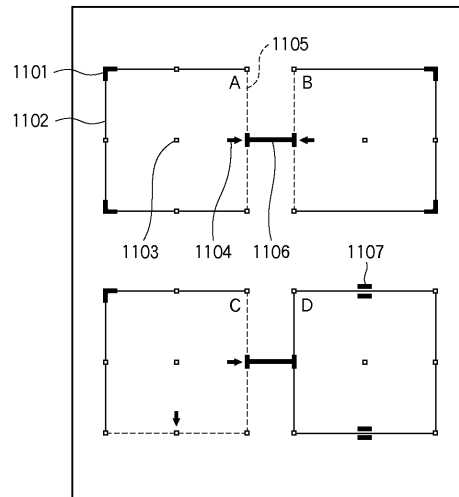
【図 9】



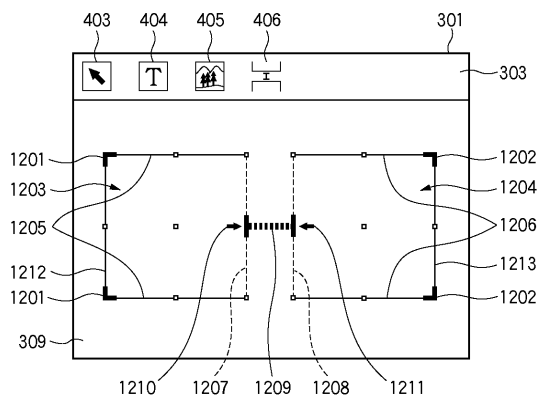
【図 10】



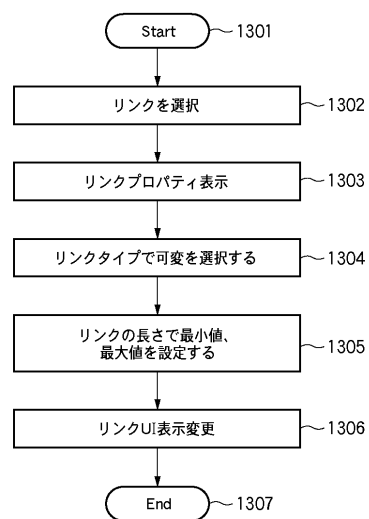
【図 11】



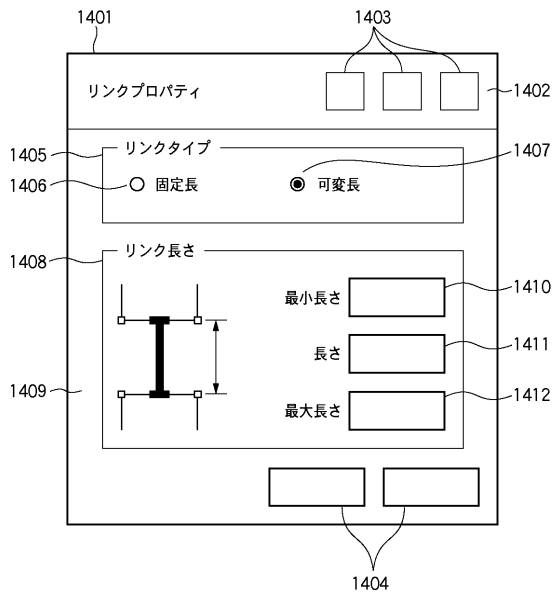
【図 12】



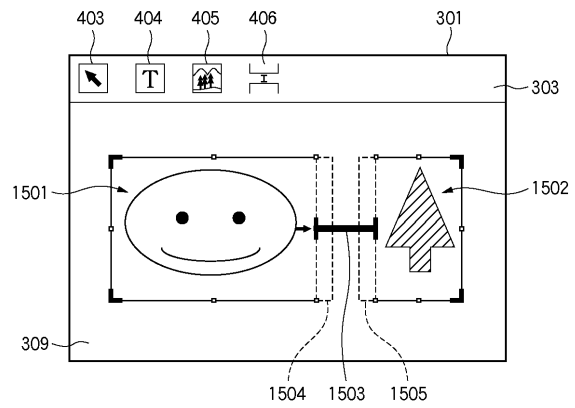
【図 13】



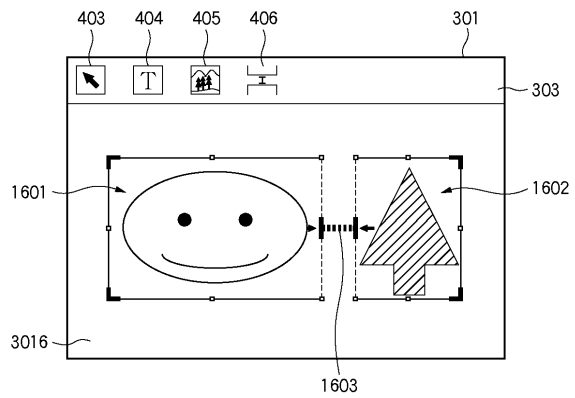
【図 14】



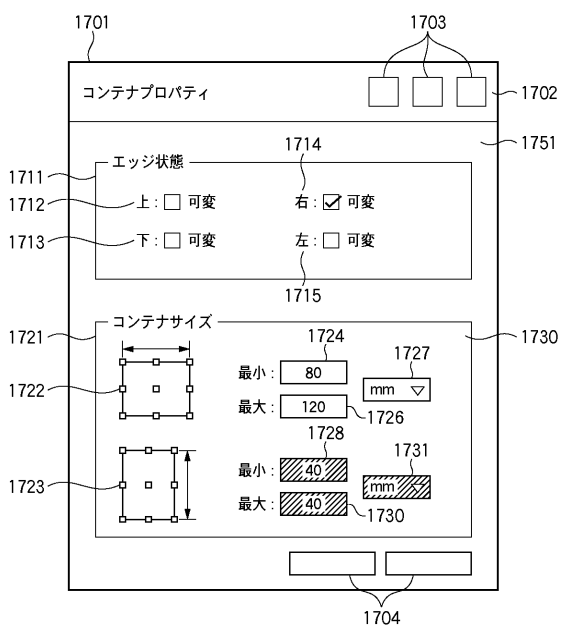
【図 15】



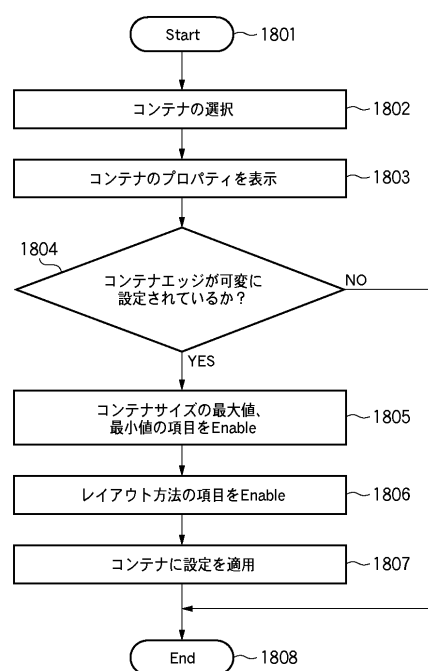
【図 16】



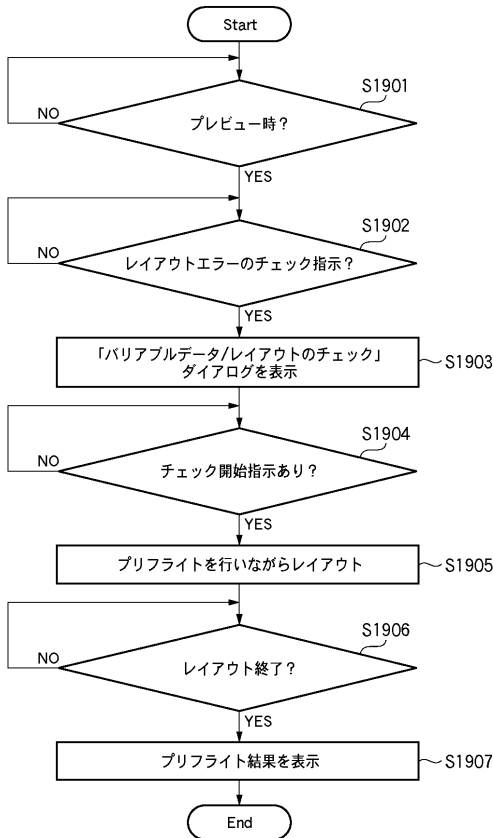
【図 17】



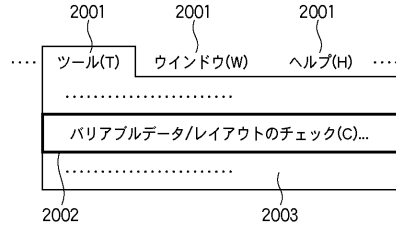
【図 18】



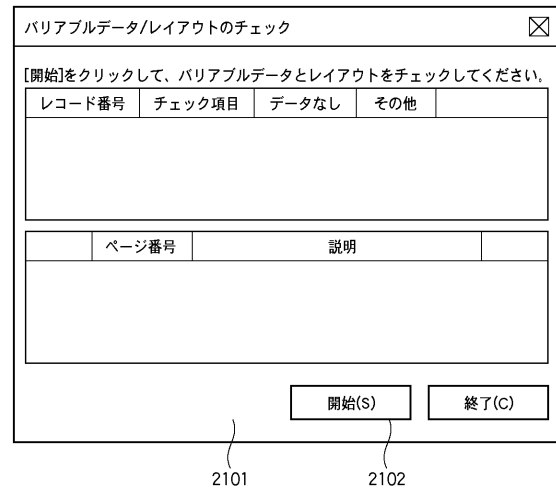
【図 19】



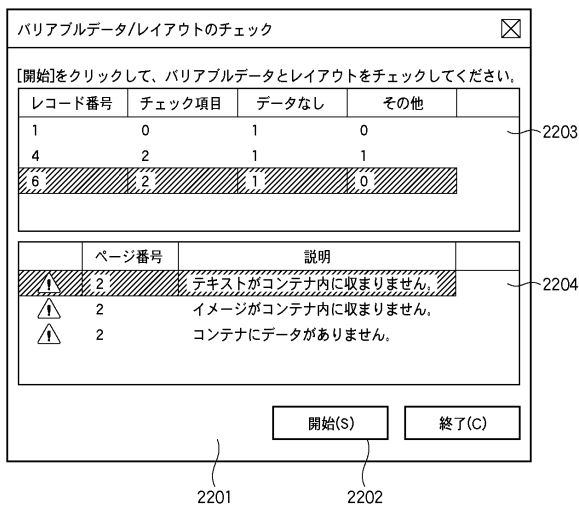
【図 20】



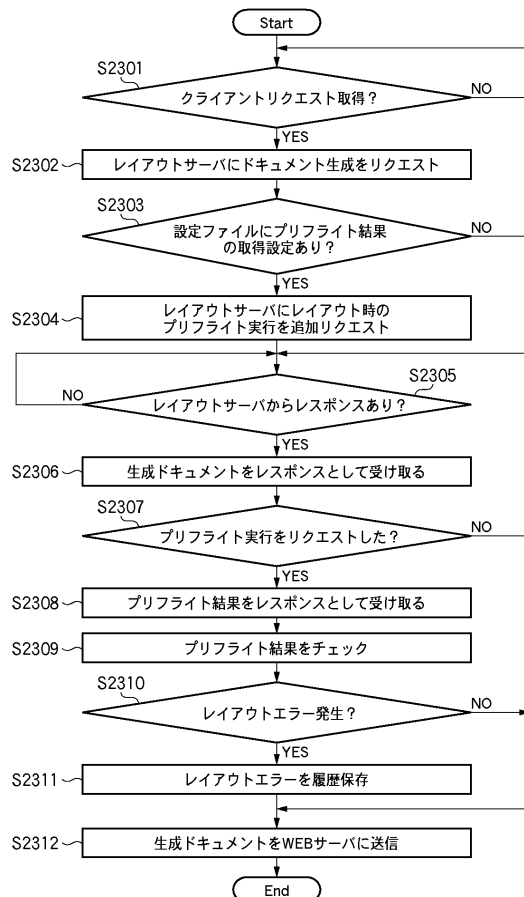
【図 21】



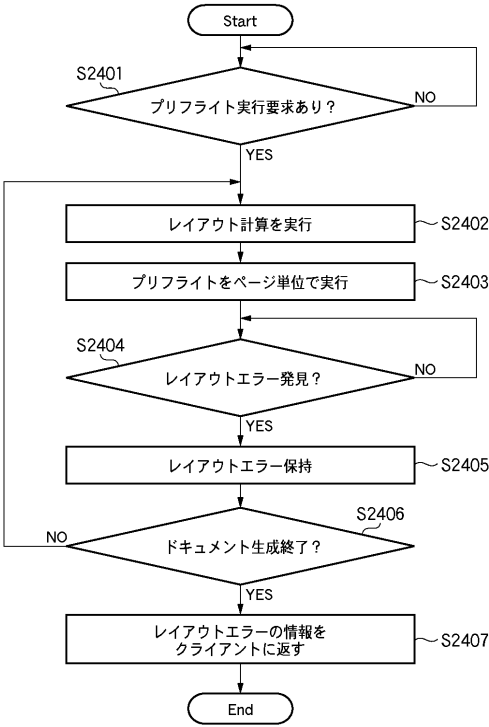
【図 22】



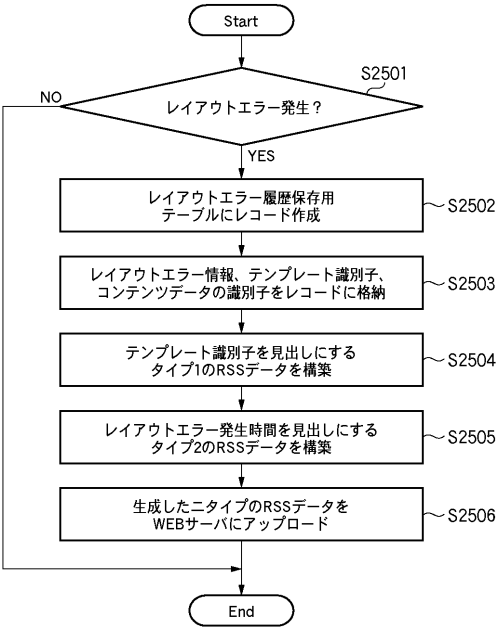
【図 23】



【図 2 4】



【図 2 5】



【図 2 6】

Template ID	Contents Record IDs String	Layout Error String	Data and Time
MITUMORI	200, 113, 31, 42838, 3187, 98, ...	Pre-Print:0x13127* Page#1:0x13020 0x13 ...	2005/10/14 13:45:52.385
PRINTER	113, 87, 8, 1898, 100, 183, 11 ...	Pre-Print* Page#1:0x13020 0x1 ...	2005/10/15 18:07:23.987
HOUKOKU	87, 9131, 13889, 57, 450, 6789, ...	Pre-Print:0x13128 0x13129* Page#1:0x13 ...	2005/10/17 02:11:52.305
MITUMORI	345, 89, 18, 37, 25910, 200, 3 ...	Pre-Print:0x13127* Page#1:0x13020 0x13 ...	2005/10/17 03:12:21.009
...

【図 2 7】

エラー番号	エラー概要
0x13020	TEXT OVERFLOW
0x13021	IMAGE OVERFLOW
0x13022	DESIGN OVERFLOW
0x13040	MINMAX CONT SIZE
0x13041	MINMAX LINK SIZE
0x13042	MINMAX FONT SIZE
0x13043	IMAGE RESOLUTION
0x13060	EMPTY OBJECTS
0x13061	DESIGN JUMPED
0x13100	OBJECTS OVERLAP
0x13101	DIFF BASIC SIZE
0x13120	FONTER
0x13121	IMGFILENOTFOUND
0x13123	INVALID FIELD
0x13124	INVALID VARIABLE
0x13125	INVALID UNIT
0x13126	CONDDVAR NODESIGN
0x13127	CONDDVAR NOFIELD
0x13128	CONDDVAR NOHIT
0x13129	CONDDVAR NOTUSED

【図 28】

```

2801 <?xml version="1.0" encoding="UTF-8"?>
2802 <rdf:RDF
  xmlns="http://purl.org/rss/1.0/"
  xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
2803 <channel rdf:about="http://www.canon.com/iwBS/5DWF506/
  LayoutErrorList.rss">
2804 <title>Recent Layout Error</title>
2805 <link>http://www.canon.com/iwBS/5DWF506/LayoutErrorList.html</link>
2806 <description>Recent Layout Error List</description>
  </channel>
2807 <items>
2808 <rdf:Seq>
2809 <rdf:li rdf:resource="http://www.canon.com/iwBS/5DWF506/
  LayoutErrorList-MITUMORI.html"/>
  <rdf:li rdf:resource="http://www.canon.com/iwBS/5DWF506/
  LayoutErrorList-PRINTER.html"/>
  <rdf:li rdf:resource="http://www.canon.com/iwBS/5DWF506/
  LayoutErrorList-HOUKOKU.html"/>
  <!--rdf:li要素のリスト-->
  </rdf:Seq>
  </items>
2810 <item rdf:about="http://www.canon.com/iwBS/5DWF506/LayoutErrorList-
  MITUMORI.html">
2811 <title>Template-MITUMORI</title>
2812 <link>http://www.canon.com/iwBS/5DWF506/Template/
  MITUMORI.dle</link>
2813 <description>2005/10/18 13:48:23.785%n 2005/10/18 12:38:44.008%n
  2005/10/18 09:13:14.865...(12)</description>
  </item>
2814 <item rdf:about="http://www.canon.com/iwBS/5DWF506/LayoutErrorList-
  PRINTER.html">
2815 <title>Template-PRINTER</title>
2816 <link>http://www.canon.com/iwBS/5DWF506/Template/PRINTER.dle</link>
2817 <description>2005/10/18 13:13:44.006%n 2005/10/03 12:38:44.038%n
  2005/09/27 23:08:58.915...(8)</description>
  </item>
2818 <item rdf:about="http://www.canon.com/iwBS/5DWF506/LayoutErrorList-
  HOUKOKU.html">
2819 <title>Template-HOUKOKU</title>
2820 <link>http://www.canon.com/iwBS/5DWF506/Template/
  HOUKOKU.dle</link>
2821 <description>2005/10/18 13:12:03.885%n 2005/10/18 12:38:44.008%n
  2005/10/18 09:13:14.865...(117)</description>
  </item>
  <!--item要素のリスト-->
</rdf:RDF>

```

LayoutErrorList.rss

【図 29】

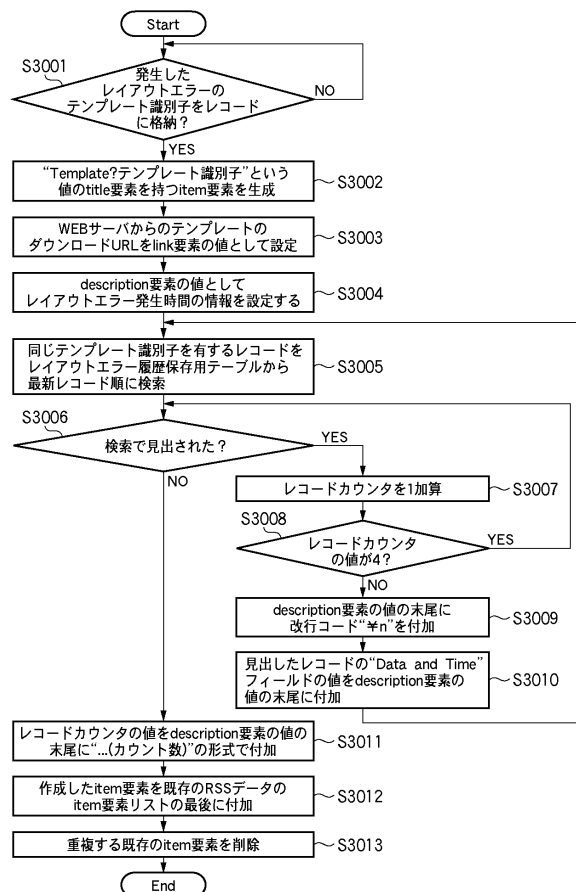
```

2901 <?xml version="1.0" encoding="UTF-8"?>
2902 <rdf:RDF
  xmlns="http://purl.org/rss/1.0/"
  xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
2903 <channel rdf:about="http://www.canon.com/iwBS/5DWF506/
  LayoutErrorList-MITUMORI.rss">
2904 <title>Recent Layout Error-MITUMORI</title>
2905 <link>http://www.canon.com/iwBS/5DWF506/LayoutErrorList-
  MITUMORI.html</link>
2906 <description>Recent Layout Error List-MITUMORI</description>
  </channel>
2907 <items>
2908 <rdf:Seq>
2909 <rdf:li rdf:resource="http://www.canon.com/iwBS/5DWF506/
  LayoutError-MITUMORI-20051017_031221.009.html"/>
  <rdf:li rdf:resource="http://www.canon.com/iwBS/5DWF506/
  LayoutError-MITUMORI-20051016_124358.120.html"/>
  <rdf:li rdf:resource="http://www.canon.com/iwBS/5DWF506/
  LayoutError-MITUMORI-20051014_134552.385.html"/>
  <!--rdf:li要素のリスト-->
  </rdf:Seq>
  </items>
2910 <item rdf:about="http://www.canon.com/iwBS/5DWF506/LayoutError-
  MITUMORI-20051017_031221.009.html">
2911 <title>2005/10/17 03:12:21.009</title>
2912 <link>http://www.canon.com/iwBS/5DWF506/Data/
  MITUMORI-20051017_031221.009.zip</link>
2913 <description>Pre-Print:0x13127%nPrint-Page#1.0x13020
  0x13...</description>
  </item>
2914 <item rdf:about="http://www.canon.com/iwBS/5DWF506/LayoutError-
  MITUMORI-20051016_124358.120.html">
2915 <title>2005/10/16 12:43:58.120</title>
2916 <link>http://www.canon.com/iwBS/5DWF506/Data/MITUMORI-
  20051016_124358.120.zip</link>
2917 <description>Pre-Print:0x13127%nPrint-Page#1.0x13020
  0x13...</description>
  </item>
2918 <item rdf:about="http://www.canon.com/iwBS/5DWF506/LayoutError-
  MITUMORI-20051014_134552.385.html">
2919 <title>2005/10/14 13:45:52.385</title>
2920 <link>http://www.canon.com/iwBS/5DWF506/Data/MITUMORI-20051014
  134552.385.zip</link>
2921 <description>Pre-Print:nPrint-Page#1.0x13020 0x13126...</description>
  </item>
  <!--item要素のリスト-->
</rdf:RDF>

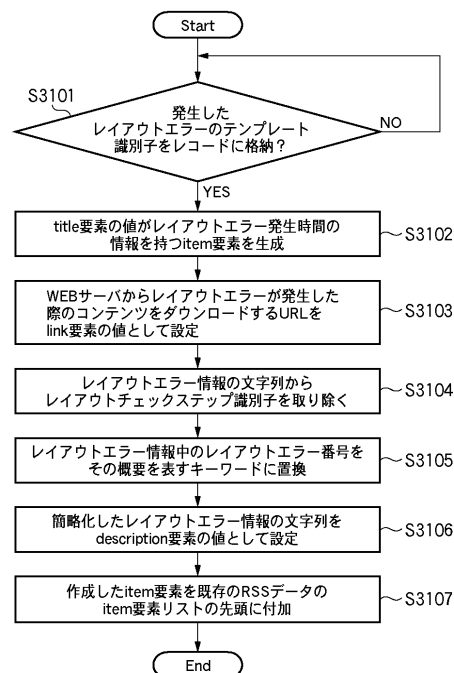
```

LayoutErrorList-MITUMORI.rss

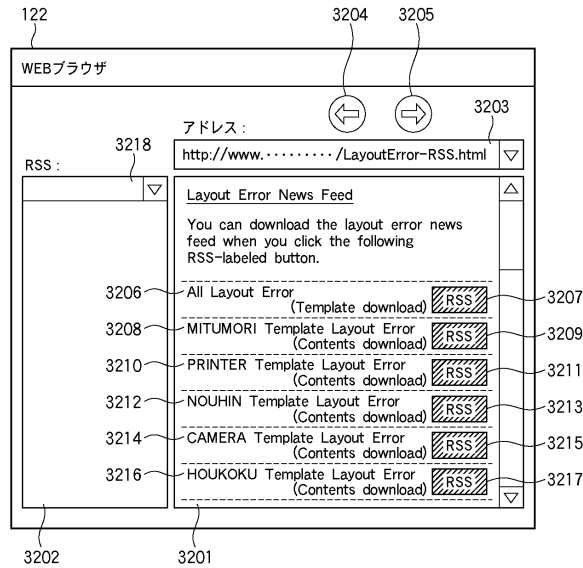
【図 30】



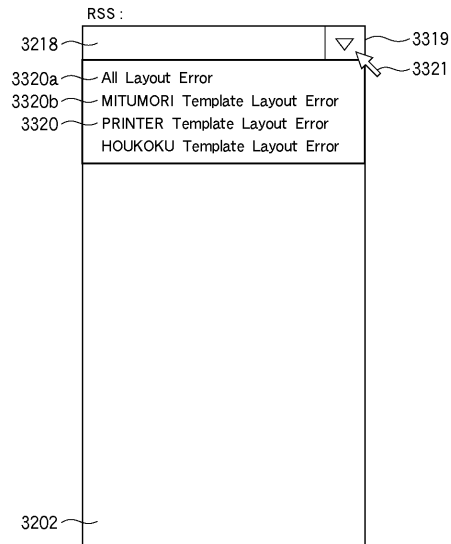
【図 31】



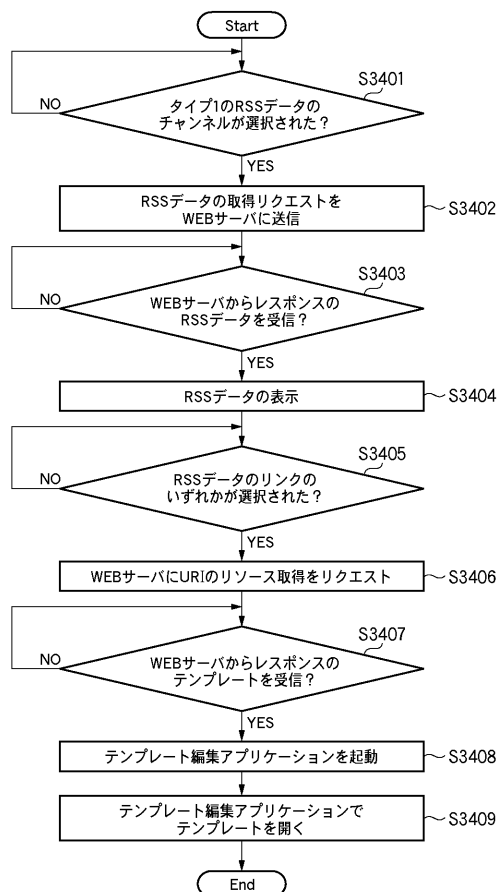
【図 3 2】



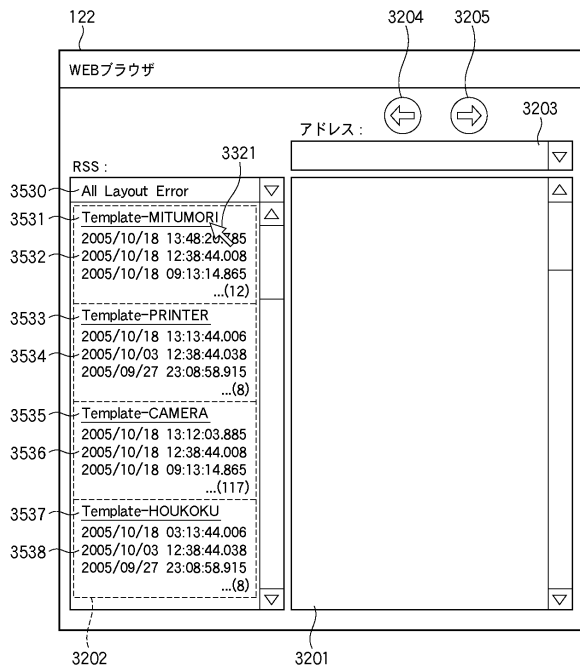
【図 3 3】



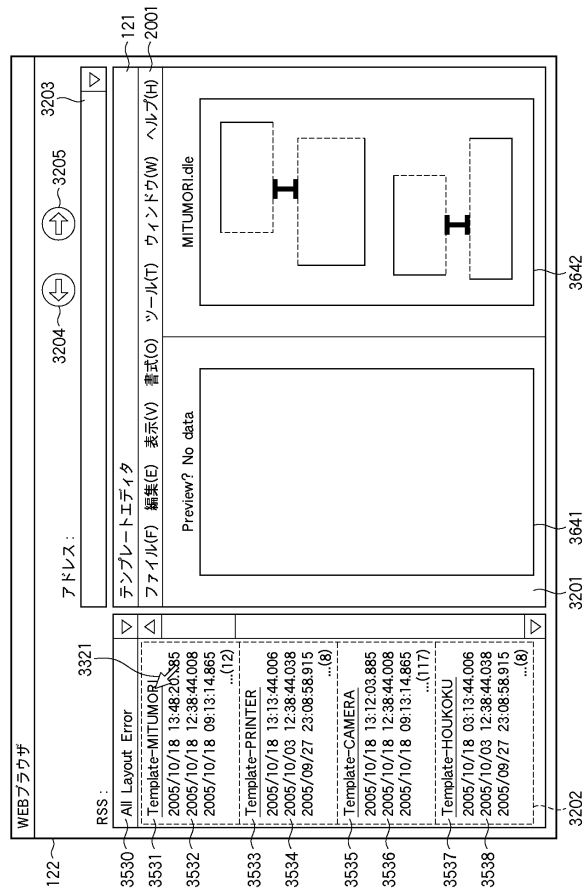
【図 3 4】



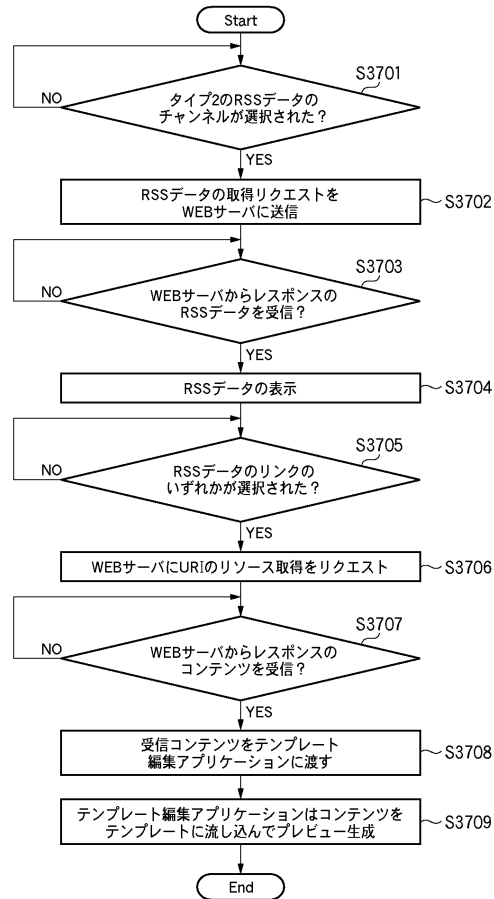
【図 3 5】



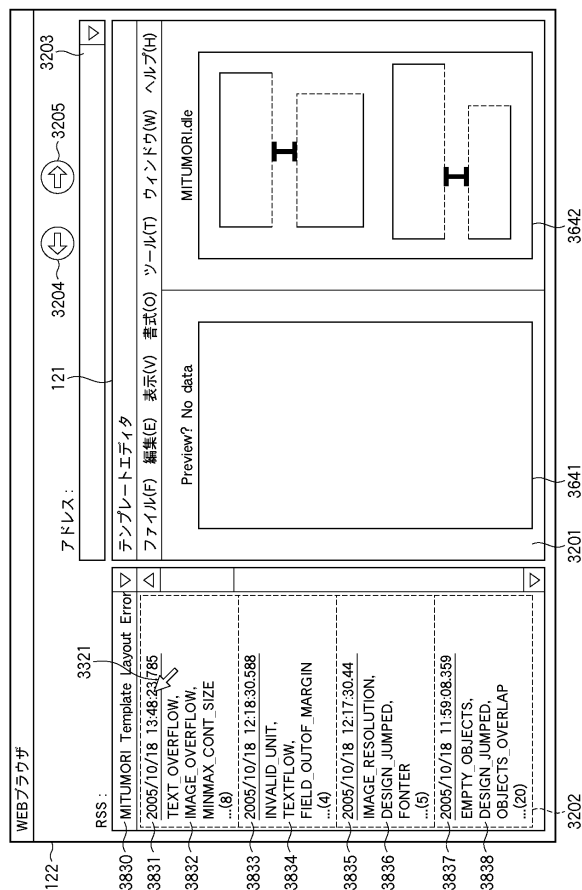
【図 36】



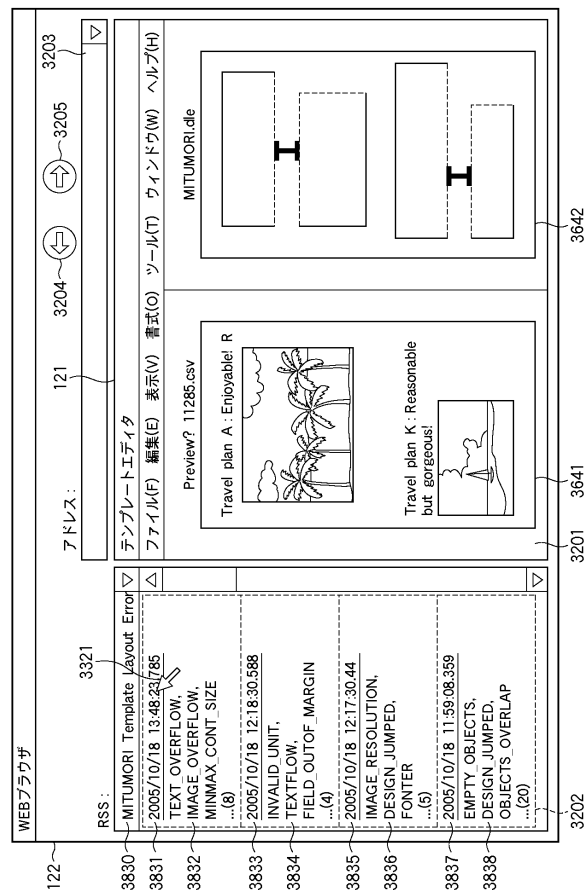
【図 37】



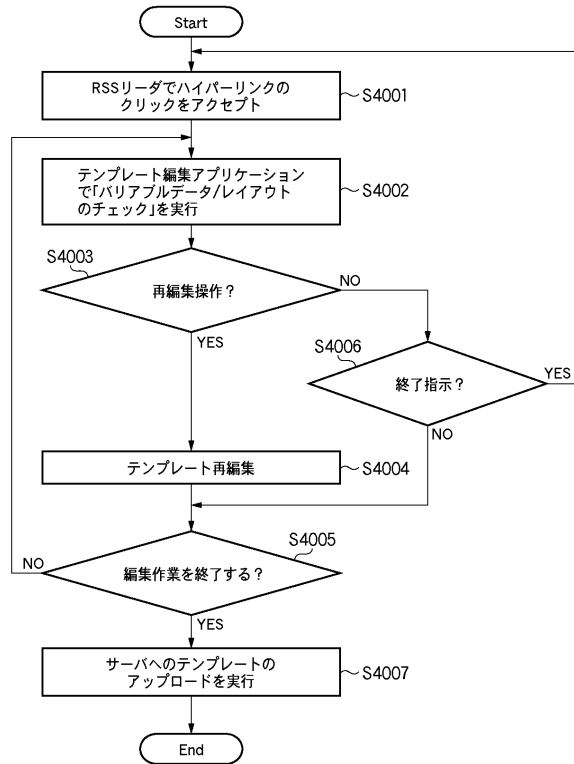
【図 38】



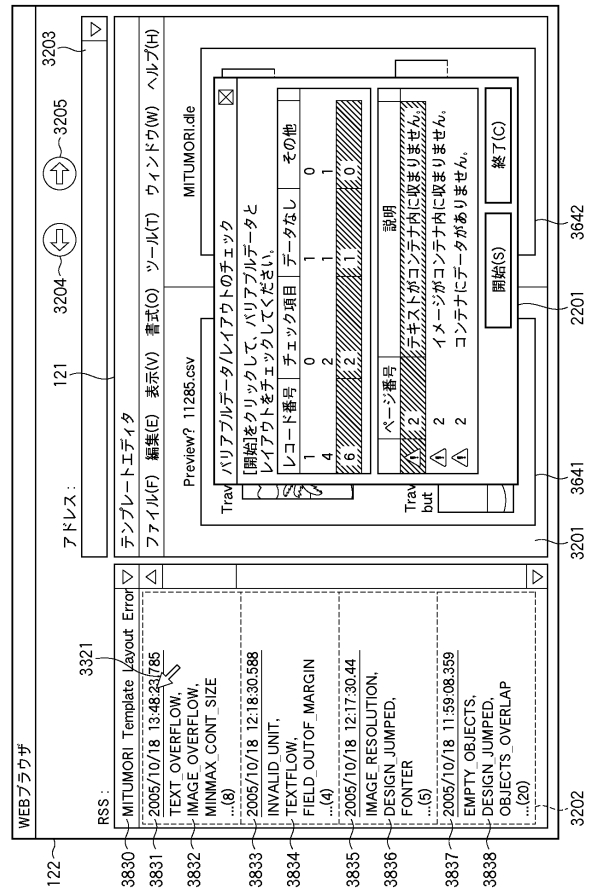
【図 39】



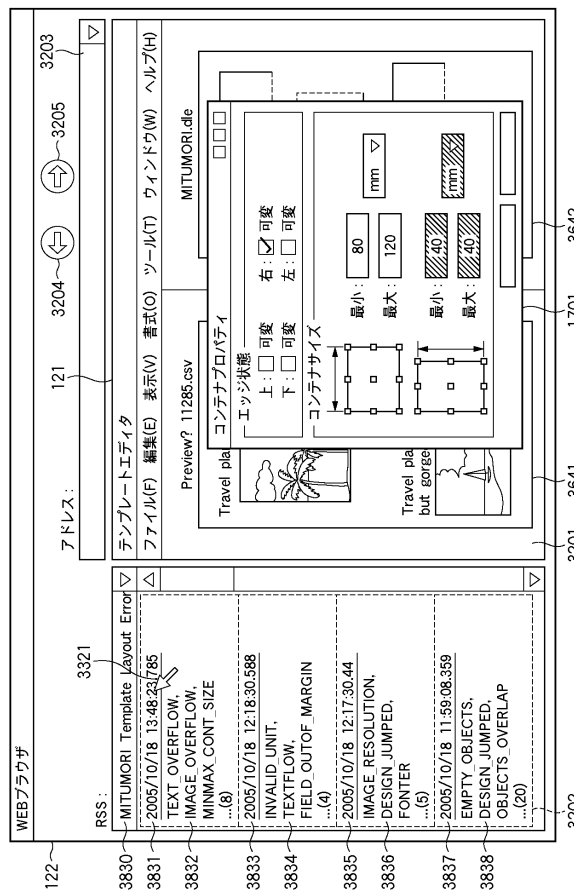
【図 40】



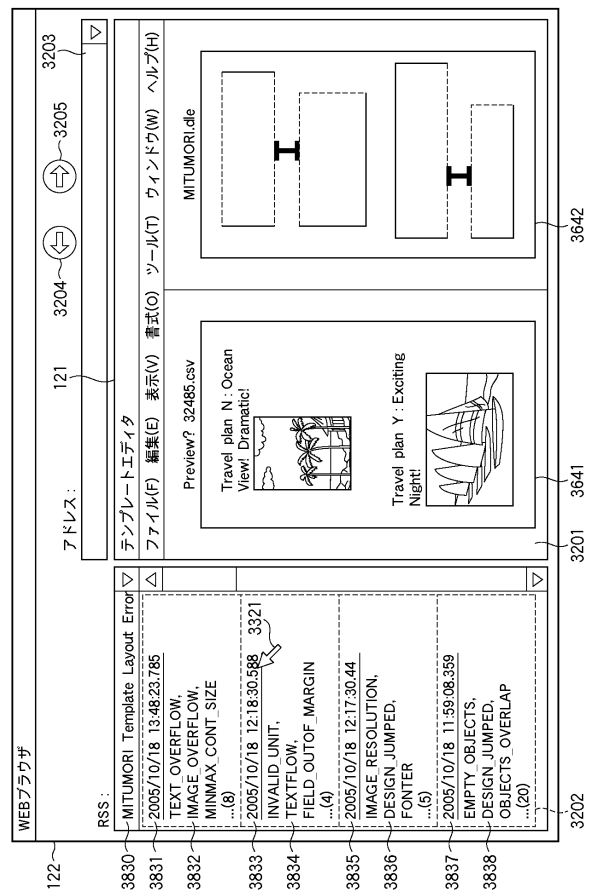
【図 41】



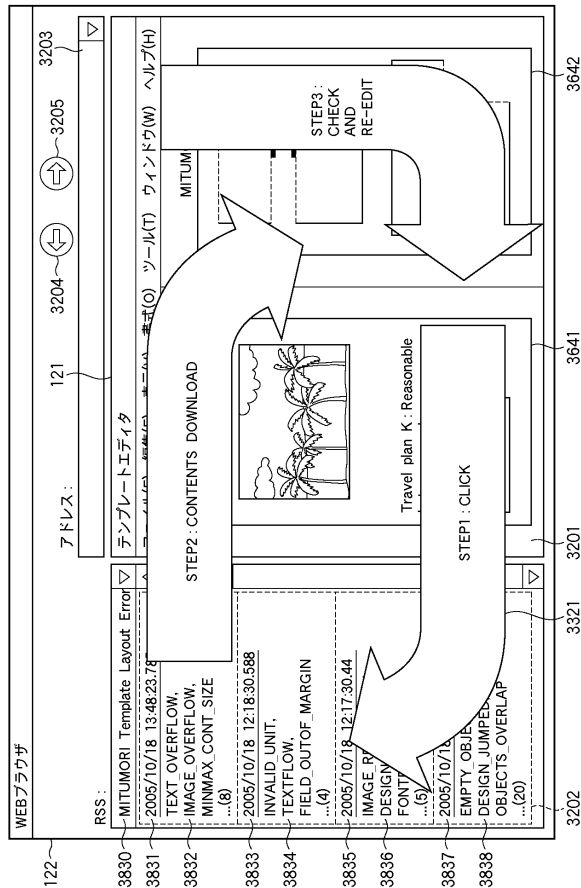
【図 42】



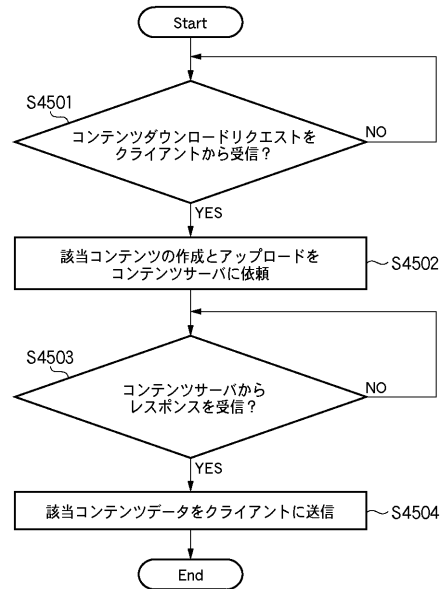
【図 43】



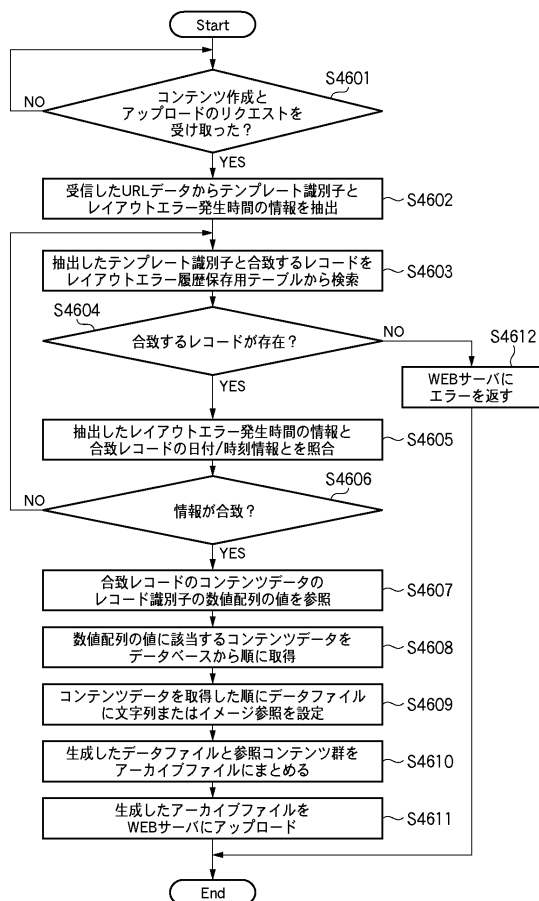
【図 44】



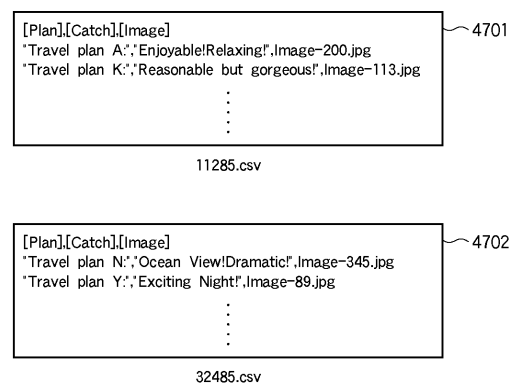
【図 45】



【図 46】



【図 47】



フロントページの続き

審査官 西出 隆二

- (56)参考文献 特開2005-122728(JP,A)
特開2006-085402(JP,A)
特開2006-074226(JP,A)
特開2006-065677(JP,A)
特開2006-074225(JP,A)
レッドクルーズ株式会社, RSSメディアビジネス最前線 電子メール、ウェブに続く第三のインターネットメディア, iNTERNET magazine make innovation with technology! No.127, 日本, 株式会社インプレス, 2005年 8月 1日, 第127号, 44~71頁

- (58)調査した分野(Int.Cl., DB名)
G06T 11/60
G06F 17/21
G06Q 50/00