

(19) 日本国特許庁(JP)

(12) 特許公報(B2)

(11) 特許番号

特許第4444634号  
(P4444634)

(45) 発行日 平成22年3月31日(2010.3.31)

(24) 登録日 平成22年1月22日(2010.1.22)

(51) Int.Cl. F I  
G O 6 F 12/00 (2006.01) G O 6 F 12/00 5 1 3 J

請求項の数 7 (全 24 頁)

<p>(21) 出願番号 特願2003-383415 (P2003-383415)                  (22) 出願日 平成15年11月13日(2003.11.13)                  (65) 公開番号 特開2005-148976 (P2005-148976A)                  (43) 公開日 平成17年6月9日(2005.6.9)                  審査請求日 平成18年10月27日(2006.10.27)</p>	<p>(73) 特許権者 301015956                  キヤノンソフトウェア株式会社                  東京都港区三田3丁目9番6号                  (74) 代理人 100145827                  弁理士 水垣 親房                  (72) 発明者 中川 裕之                  東京都港区三田3丁目9番7号 キヤノン                  ソフトウェア株式会社内                  審査官 田川 泰宏</p>
---	--

最終頁に続く

(54) 【発明の名称】 プログラム生成装置およびプログラム生成方法およびプログラムおよび記録媒体

(57) 【特許請求の範囲】

【請求項1】

複数のデータベースのテーブルにアクセス可能な機能を有するウェブアプリケーションプログラムを生成するプログラム生成装置であって、

データベースのデータベース名およびテーブル名の組合せを含む情報によりテーブルをそれぞれ定義する複数のテーブル情報を記憶する第1の記憶手段と、

前記複数のテーブル情報によりそれぞれ定義された複数のテーブルを1つの画面で切り替え可能なグループとして定義したグループ情報を記憶する第2の記憶手段と、

前記データベース毎に、データベースへ接続する際に用いるドライバ名、接続先、及び認証情報を含む接続情報を記憶する第3の記憶手段と、

前記グループ情報を用いて、前記グループ情報でグループとして定義された前記複数のテーブルのいずれかを接続の切り替え先として選択可能な操作画面をウェブブラウザで表示するためのプログラムコードを、生成する第1の生成手段と、

ウェブブラウザに表示される前記操作画面の操作により接続の切り替え先として選択されたテーブルの情報を前記ウェブブラウザから受け取った場合、前記選択されたテーブルに対応するデータベースの前記接続情報を用いて、前記選択されたテーブルに対応するデータベースへ接続を切り替えて前記選択されたテーブルにアクセスするためのプログラムコードを、生成する第2の生成手段と、

を有することを特徴とするプログラム生成装置。

【請求項2】

10

20

前記テーブル情報で定義されたテーブルは、同一のデータベースもしくは異なるデータベース上に配置されるものであることを特徴とする請求項 1 記載のプログラム生成装置。

【請求項 3】

前記操作画面は、前記操作画面で接続の切り替え先として選択されているテーブルに対して検索条件を指定して検索指示を行うための検索指示部を有し、

前記第 2 の生成手段により生成されるプログラムコードは、「前記操作画面の操作により前記操作画面で接続の切り替え先として選択されているテーブルに対する前記検索指示をウェブブラウザから受け取った場合、前記選択されているテーブルに対応するデータベースに接続されているか判定し、前記選択されているテーブルに対応するデータベースに接続されていない場合は、前記選択されているテーブルに対応するデータベースの前記接続情報を用いて前記データベースに接続して前記検索指示で指定された検索条件での検索処理を実行し、一方、前記選択されているテーブルに対応するデータベースに接続されている場合は、そのまま前記検索指示で指定された検索条件での検索処理を実行するためのプログラムコード」を含むことを特徴とする請求項 2 記載のプログラム生成装置。

10

【請求項 4】

前記操作画面は、前記操作画面で選択されているテーブルに対して更新データを指定して更新指示を行うための更新指示部を有し、

前記第 2 の生成手段により生成されるプログラムコードは、「前記操作画面の操作により前記操作画面で選択されているテーブルに対する前記更新指示をウェブブラウザから受け取ると、前記選択されているテーブルに対応するデータベースに接続されているか判定し、前記選択されているテーブルに対応するデータベースに接続されていない場合は、前記選択されているテーブルに対応するデータベースの前記接続情報を用いて前記データベースに接続して前記更新データでの更新処理を実行し、一方、前記で選択されているテーブルに対応するデータベースに接続されている場合は、そのまま前記更新指示で指示された更新データでの更新処理を実行するためのプログラムコード」を含むことを特徴とする請求項 2 又は 3 記載のプログラム生成装置。

20

【請求項 5】

複数のデータベースのテーブルにアクセス可能な機能を有するウェブアプリケーションプログラムを生成するコンピュータにおけるプログラム生成方法であって、

前記コンピュータの記憶手段に、データベースのデータベース名およびテーブル名の組合せを含む情報によりテーブルをそれぞれ定義する複数のテーブル情報、前記複数のテーブル情報によりそれぞれ定義された複数のテーブルを 1 つの画面で切り替え可能なグループとして定義したグループ情報、前記データベース毎に、データベースへ接続する際に用いるドライバ名、接続先、及び認証情報を含む接続情報を、それぞれ記憶させておき、

30

前記コンピュータが、前記グループ情報を用いて、前記グループ情報でグループとして定義された前記複数のテーブルのいずれかを接続の切り替え先として選択可能な操作画面をウェブブラウザで表示するためのプログラムコードを、生成する第 1 の生成ステップと

前記コンピュータが、ウェブブラウザに表示される前記操作画面の操作により接続の切り替え先として選択されたテーブルの情報を前記ウェブブラウザから受け取った場合、前記選択されたテーブルに対応するデータベースの前記接続情報を用いて、前記選択されたテーブルに対応するデータベースへ接続を切り替えて前記選択されたテーブルにアクセスするためのプログラムコードを、生成する第 2 の生成ステップと、  
を有することを特徴とするプログラム生成方法。

40

【請求項 6】

請求項 5 に記載されたプログラム生成方法をコンピュータに実行させるためのプログラム。

【請求項 7】

請求項 5 に記載されたプログラム生成方法をコンピュータに実行させるためのプログラムをコンピュータが読み取り可能に記録した記録媒体。

50

## 【発明の詳細な説明】

## 【技術分野】

## 【0001】

本発明は、ウェブ（Web）アプリケーション開発支援システムにおいて、ユーザが動的にデータベース（DB）の切り替えが可能なウェブアプリケーションプログラムの一連のプログラムコードを業務仕様という枠組みの中から自動生成可能なプログラム生成装置およびプログラム生成方法およびプログラムおよび記録媒体に関するものである。

## 【背景技術】

## 【0002】

従来、リレーショナルデータベースシステムなど、複数のテーブルに分かれて格納されているデータベース（DB）システムの場合、DBを構成する複数の関連あるテーブルに対してデータ入力を行わなければならなかった。

## 【0003】

また、膨大なデータを扱い、しかも適切に所望のデータにアクセスするためには、DB固有のデータ構造を意識する必要があった。

## 【0004】

この問題に対し特開平6-68151号公報（特許文献1）の「データベースアクセス方法および装置」では、DBのデータ構造を論理的に構築したスキーマを管理し表示するブラウザ手段により、DBアクセス手段を備える方法を提案している。この方法により、ユーザはDBの構造を意識することなく、効率よくDBの操作が行えるようになった。

## 【特許文献1】特開平6-68151号公報

## 【発明の開示】

## 【発明が解決しようとする課題】

## 【0005】

しかしながら、上記特許文献1では、テーブルの切り替えが行えるのは単一のDBシステム内のリレーションが張られているテーブルのみとなっており、分散型DBに対するアクセス手段については何ら考慮がなされていない。

## 【0006】

そのため、対象となるDBシステム以外のDBにアクセスしようとする場合、そのDB用のアプリケーションを一から開発しなくてはならないという問題点があった。

## 【0007】

本発明は、上記の問題点を解決するためになされたもので、本発明の目的は、分散したデータベースに対する接続をユーザの選択により動的に切り替え、該データベースのテーブルへのアクセス処理が可能なウェブアプリケーションプログラム、又は同一のデータベース内の異なるテーブルをユーザの選択により動的に切り替えてアクセス処理が可能なウェブアプリケーションプログラムを業務仕様の枠組みから選択的に自動生成でき、生産性と保守性に優れたウェブアプリケーションプログラムの領域を広げることができる優れたプログラム生成環境を構築できるプログラム生成装置およびプログラム生成方法およびプログラムおよび記録媒体を提供することである。

## 【課題を解決するための手段】

## 【0008】

本発明は、複数のデータベースのテーブルにアクセス可能な機能を有するウェブアプリケーションプログラムを生成するプログラム生成装置であって、データベースのデータベース名およびテーブル名の組合せを含む情報によりテーブルをそれぞれ定義する複数のテーブル情報を記憶する第1の記憶手段と、前記複数のテーブル情報によりそれぞれ定義された複数のテーブルを1つの画面で切り替え可能なグループとして定義したグループ情報を記憶する第2の記憶手段と、前記データベース毎に、データベースへ接続する際に用いるドライバ名、接続先、及び認証情報を含む接続情報を記憶する第3の記憶手段と、前記グループ情報を用いて、前記グループ情報でグループとして定義された前記複数のテーブルのいずれかを接続の切り替え先として選択可能な操作画面をウェブブラウザで表示する

10

20

30

40

50

ためのプログラムコードを、生成する第1の生成手段と、ウェブブラウザに表示される前記操作画面の操作により接続の切り替え先として選択されたテーブルの情報を前記ウェブブラウザから受け取った場合、前記選択されたテーブルに対応するデータベースの前記接続情報を用いて、前記選択されたテーブルに対応するデータベースへ接続を切り替えて前記選択されたテーブルにアクセスするためのプログラムコードを、生成する第2の生成手段とを有することを特徴とする。

【発明の効果】

【0016】

本発明によれば、分散したデータベースに対する接続をユーザの選択により動的に切り替え、該データベースのテーブルへのアクセス処理が可能なウェブアプリケーションプログラム、又は同一のデータベース内の異なるテーブルをユーザの選択により動的に切り替えてアクセス処理が可能なウェブアプリケーションプログラムを業務仕様の枠組みから選択的に自動生成でき、生産性と保守性に優れたウェブアプリケーションプログラムの領域を広げることができる優れたプログラム生成環境を構築できる等の効果を奏する。

10

【発明を実施するための最良の形態】

【0017】

以下、図を参照して本発明のアプリケーション開発システムについて説明する。

【0018】

図1は、本発明の一実施形態を示すプログラム生成装置の構成を示すブロック図である。

20

【0019】

図において、101は入力装置で、キーボードやマウス等のポインティングデバイス等に相当し、業務仕様情報（データ項目のデータベースへの配置とデータ項目間の関係、即ちあるデータがどのような条件のもとにどのようなデータから求められるかを示す情報等）の入力を行うことが可能である。

【0020】

102は表示装置で、CRT、LCD等で構成され、業務仕様情報の入力画面や、該入力画面から入力された業務仕様情報、生成されたプログラムコード等を表示可能である。104はハードディスク等の外部記憶装置で、入力された業務仕様（データベース上の複数のテーブルをそれぞれ定義した定義情報）を記憶するための入出力テーブル109、入力されたデータベースへ接続するための接続情報を記憶するログイン情報110、入力された生成するウェブアプリケーションのアプリケーション名111、生成されたプログラムコード（JSPコード112、サーブレット（Servlet）コード113、Beanコード114）等を記憶するものである。

30

【0021】

103は処理装置で、CPU、ROM、RAM等から構成され、CPUがROM、外部記憶装置104又はその他の記録媒体に格納されるプログラムをRAM上にロードして実行することにより実現されるものであり、画面制御部105、コード生成部107を備える。

【0022】

画面制御部105は、入力装置101から業務仕様情報の入力を行うために表示装置102の画面（不図示）を制御する業務仕様記述部116を有する。

40

【0023】

コード生成部107は、入力装置101から入力された又は外部記憶装置104から読み出された業務仕様情報に基づいてプログラムコード（JSPコード112、サーブレット（Servlet）コード113、Beanコード114）を生成し、外部記憶装置104に記憶させる。

【0024】

108は全体制御部で、画面制御部105、コード生成部107等の各部の制御を行う。

50

## 【 0 0 2 5 】

図 2 は、図 1 に示したコード生成部 1 0 7 によって生成したサーブレット、J S P、および B e a n を利用したウェブアプリケーションシステムの全体構成の一例を示す図である。

## 【 0 0 2 6 】

コード生成部 1 0 7 によって生成されたサーブレット、J S P、および B e a n は、それぞれ実行可能な状態で、サーバ 3 0 4 側のアプリケーションサーバ 3 0 6 上に配置される。

## 【 0 0 2 7 】

この W e b アプリケーションシステムは、クライアント 3 0 1 からウェブブラウザ 3 0 2 によって利用される。ウェブブラウザ 3 0 2 から J S P 3 0 9 にアクセスすることにより、H T M L ページ 3 0 3 を生成し、クライアント 3 0 1 側に返送する。そして、ウェブブラウザ 3 0 2 上には、J S P 3 0 9 から送信される H T M L ページ 3 0 3 が表示される。

10

## 【 0 0 2 8 】

また、W e b ブラウザ 3 0 2 からのリクエストは、H T T P 経由でサーバ 3 0 4 側に送信される。サーバ 3 0 4 には H T T P サーバ 3 0 5 とアプリケーションサーバ 3 0 6 があり、アプリケーションサーバ 3 0 6 にはサーブレットや J S P の実行環境が含まれている。

## 【 0 0 2 9 】

サーバ 3 0 4 側にリクエストが送信されると、リクエストに応じて該当するサーブレット 3 0 7 が起動される。サーブレット 3 0 7 は、対応する B e a n 3 0 8 に処理を要求する。さらに、B e a n 3 0 8 は、サーブレット 3 0 7 からの要求に応じて D B アクセスオブジェクト 3 1 0 a , D B アクセスオブジェクト 3 1 0 b を利用して内部データベース 3 1 1 a , 外部データベース 3 1 1 b にアクセスし、その処理結果を B e a n 3 0 8 内に保持する。

20

## 【 0 0 3 0 】

次に、サーブレット 3 0 7 から J S P 3 0 9 に B e a n 3 0 8 を渡し、J S P 3 0 9 を呼び出す。J S P 3 0 9 は、B e a n 3 0 8 で保持する処理結果を参照して、H T M L ページを生成し、クライアント 3 0 1 側に返送する。クライアント 3 0 1 側に返送された H T M L ページは、W e b ブラウザ 3 0 2 上に表示される。

30

## 【 0 0 3 1 】

サーブレット 3 0 7 は、W e b ブラウザ 3 0 2 からのリクエストの受け付け、B e a n 3 0 8 への処理要求、および J S P 3 0 9 の呼び出しという、それぞれの間をつなぎ、制御する役割を持つ。J S P 3 0 9 は H T M L ページを出力するための表示に関することを受け持ち、B e a n 3 0 8 はデータベース 3 1 1 a , 3 1 1 b へのアクセス等の業務に関する処理を受け持つ。

## 【 0 0 3 2 】

なお、B e a n 3 0 8 や J S P 3 0 9 を利用せずにサーブレット 3 0 7 のみで（又は、サーブレット 3 0 7 や B e a n 3 0 8 を利用せずに J S P 3 0 9 のみで）、W e b ブラウザ 3 0 2 からのリクエストを処理することは可能であるが、B e a n 3 0 8 や J S P 3 0 9 を利用してそれぞれの役割を明確に分離することで、機能分担を簡潔に把握することができる。

40

## 【 0 0 3 3 】

なお、B e a n 3 0 8 は、内部データベースアクセス用の B e a n と外部データベースアクセス用の B e a n とに分けて実装してもよい。

## 【 0 0 3 4 】

また、内部データベースが複数あってもよいし、外部データベースが複数あってもよい。

## 【 0 0 3 5 】

50

図3は、本発明のプログラム生成装置における第1の制御処理手順の一例を示すフローチャートであり、図1に示した処理装置103の全体制御部108の動作に対応する。なお、S101～S105は各ステップを示す。

【0036】

図1に示した処理装置103は、コード生成する業務仕様(入出力テーブル)の入力を行う入力モードと、入力された業務仕様(入出力テーブル)からプログラムコードを生成する生成モードとを有する。

【0037】

まず、全体制御部108が起動されると、ステップS101において、入力装置101からオペレータの指示により選択されたモードが生成モードか入力モードかを判定し、入力モードが選択されたと判定した場合には、ステップS102に進み、全体制御部108は、画面制御部105を起動する。

10

【0038】

起動された画面制御部105は、ステップS103において、業務仕様記述部116により業務仕様入力画面(不図示)を表示させ、オペレータからの業務仕様の一形態である入出力テーブル(詳細は図4に示す)への項目入力、ログイン情報(詳細は図5に示す)の入力、生成するウェブアプリケーションのアプリケーション名111の入力等を受け付け、外部記憶装置104に格納し(入出力テーブル109、ログイン情報110、アプリケーション名111)、処理を終了する。

【0039】

20

一方、ステップS101において、入力装置101からオペレータの指示により選択されたモードが生成モードであると判定した場合には、ステップS104に進み、全体制御部108は、外部記憶装置104に格納された入出力テーブルを読み出し、該読み出した入出力テーブル内の業務仕様情報に基づいてコード生成を行う。

【0040】

そして、ステップS105において、全体制御部108は、生成されたコードを、オペレータの指示により表示装置102に表示したり、外部記憶装置104に書き込む出力処理を行う。

【0041】

以下、図4を参照して、本発明のプログラム生成装置における業務仕様情報(入出力テーブル)について説明する。図示しない入力画面から入力される業務仕様情報は、内部的には図4に示すような入出力テーブルの形式で保存される。

30

【0042】

図4は、本発明のプログラム生成装置における業務仕様情報(図1に示した入出力テーブル109)について説明する模式図であり、ウェブアプリケーションにおけるDBの動的な切り替え機能とそれを表示するための画面の定義を記述した業務仕様を入出力テーブルの形式で示したものに对应する。

【0043】

列1は、DBシステム名を示したものであり、DBシステムを利用するときには識別用として用いる。

40

【0044】

列2は、データファイル番号を示したものであり、データファイルを構成するデータ項目として列3のデータファイル項目枝番がある。

【0045】

列4はデータファイルの名称であり、列5はデータファイル項目枝番に対応する各データ項目の名称である。

【0046】

列6は、その項目の種別を示したIO区分である。IO区分はそれぞれ、ファイルであることを示す「F」、表示項目であることを示す「D」、複数のテーブルをまとめて、一つの画面で制御できる機能があることを示す「G1」と「G2」がある。この制御につい

50

ては別途下記で説明する。

【0047】

即ち、列6のI O区分で「F」が指定されている場合には、その行のデータはデータ項目を定義し、「D」が指定されている場合には、その行のデータは画面表示項目を定義し、「G1」、「G2」が指定されている場合には、その行のデータはDB・テーブルの切り替え機能を定義するものとなる。

【0048】

列7は、各項目の桁数を表している。列8は、表示方法を示しており、「S」の場合は検索用の項目として表示され、「H」の場合は検索結果を表示する項目として扱われる。また、検索用にも結果用にも使用される項目は「S, H」として定義する。列9は、項目の項番を示している。

10

【0049】

図5は、図1に示したログイン情報を説明するための模式図である。

【0050】

図において、601はDBシステム名で、生成されるアプリケーションで接続するDBシステムの名称に対応する。602はJDBCドライバで、DBシステムに接続する際に用いるドライバである。603はURL, 604はユーザ名, 605はパスワードで、DBシステムに接続する際に用いるURL, ユーザ名, パスワードに対応する。

【0051】

なお、入出力テーブル109, ログイン情報110, アプリケーション名111は、業務仕様記述部116により表示される業務仕様入力画面(不図示)から入力する構成であってもよいし、その他の任意のアプリケーションにより生成されるテキストファイル, CSVファイル, その他のファイル形式のファイルであってもよい。

20

【0052】

以下、図6～図8を参照して、DBの動的な切り替えについて説明する。

【0053】

図6～図8は、本発明のプログラム生成装置により生成されるアプリケーションによるDB接続切り替えを説明するためのDB構成図であり、特に図6は外部(リモート)分散DB構成図に対応し、図7は内部(ローカル)同一DB構成図に対応し、図8は内部(ローカル)DB及び外部(リモート)DBによる分散DB構成図に対応する。

30

【0054】

DBの動的な切り替えとは、一つの生成された画面において検索や更新の対象となるDB・テーブルを画面上の操作で選択することで、対象DB・テーブルを切り替えることができる機能を示している。

【0055】

本発明では、図6～図8に示すように、切り替え対象のDB・テーブルを仮想的に一つのテーブルとして制御する手段をとる(図6～図8に示す仮想DB401)。

【0056】

具体的には、予め定義してあるログイン情報110(JDBCドライバ, URL, ユーザ名, パスワード)を入出力テーブル109の列6に「G2」が定義されている行(図4の例では「行2」, 「行3」)の列3の値(図4の例では「ClinDB/ClinicM」, 「DrudB/DrugstoreM」)と照らし合わせて取得し、その情報をアプリケーションに付加する。アプリケーション生成後に画面上からユーザがテーブルを選択すれば、上記情報を利用してDBシステム・テーブルへのアクセスを行うことが可能となっている。

40

【0057】

この切り替えの際、図6に示すような異なる外部DBシステム同士の切り替えの場合、及び図8に示すような外部DBシステムと内部のDBシステムとを切り替える場合(即ち、分散型DBの場合)は、DBシステムそのものの切り替えを生成されたアプリケーション内で行うこととなる。

50

## 【0058】

また、図7に示すような同一DBシステム内での切り替えの場合は、上記の外部DBの切り替えの場合と同様にして、DBシステム内のテーブルをその都度切り替えてセレクトすることを生成されたアプリケーション内で行うこととなる。

## 【0059】

このような制御方法により、生成されたアプリケーションのユーザは、DBシステムの物理的な構成や接続を意識することなく切り替えが行えるようになる。

## 【0060】

例えば、図6に示すように、外部のDBシステムに接続されたテーブル同士を交互に切り替えたり、図7に示すように、内部のDB内のテーブルを切り替えたり、図8に示すように、内部と外部のDBシステムを組み合わせたものでも、そのことを意識することなく切り替えることが可能となる。

10

## 【0061】

切り替えの対象となるDB・テーブルは、図4で示した入出力テーブル109の列6で「G2」が指定されている行(図4の例では「行2」「行3」)の列3の値(図4の例では「CLIDB/ClinicM」,「DrudB/DrugstoreM」)となる。この値の記述ルールは、DBシステム名/テーブル名のように「/」記号で区切って表す。

## 【0062】

このDB切り替え機能を使用すれば、例えばユーザが薬を検索したいといった場合には、CLIDB(例えば内部DBシステム)に接続されている病院テーブル(ClinicM)とDrudB(例えば外部ネットワークに接続されている外部DBシステム)に接続されている薬局テーブルを、ユーザが操作画面(後述する図9に示す)上から選択することで、DBの切り替えを行いその検索対象をリアルタイムに切り替えることが可能となる。

20

## 【0063】

このように、複数のDB・テーブルをまとめる仮想的なテーブルの生成は、上記の図4に示した入出力テーブル109の「行1」～「行3」の定義を読み込んで行われる。

## 【0064】

まず、入出力テーブル109において、列6のIO区分が「G1」の行(図4の例では「行1」)をセレクトする。この行は、対象となるDB・テーブルを仮想的に一つにまとめるための項目となる。対象となるDB・テーブルは、IO区分に「G2」が指定されている行2,行3に記述されている。

30

## 【0065】

IO区分に「G2」が指定された行2,行3の列3のデータファイル項目枝番はDB/テーブル名を示している。なお、生成されたアプリケーションの操作画面に表示されるテーブル名称(後述する図9の201a,201b)は列5で指定されているデータファイル項目名(図4の例では、「病院」,「薬局」)となる。

## 【0066】

図9は、本発明のプログラム生成装置により生成されるウェブアプリケーションにおける操作画面の一例を示す模式図である。

40

## 【0067】

上述したような生成方法により生成された操作画面は、図9の左上にあるようなプルダウン式の選択メニュー201と切り替え検索ボタン202が自動的に生成され、選択メニュー201の中に、201a,201bに示すように、図4の列6で「G2」が定義された行のデータファイル項目名がテーブル名称として表示される。この選択メニュー201でテーブル名称(201a又は201b)を選択し、切り替え検索ボタン202を押下すれば、DB・テーブルの接続を切り替え、選択したテーブルが検索・更新の対象となる。

## 【0068】

なお、検索のための処理は、検索条件入力欄203に値を入力し、検索ボタン204を

50



指示することで行われる。

【0069】

また、更新のための処理は、検索結果表示入力欄205で値を変更入力し、更新ボタン206を指示することで行われる。

【0070】

さらに、検索条件入力欄203に値を入力した状態で、切り替え検索ボタン202を指示すると、DB・テーブルの接続を切り替え、該切り替えられたDB・テーブルに対して、該検索条件入力欄203に入力されている検索条件での検索処理が実行される。これにより、同一の検索条件で、DB・テーブルを切り替えて順次検索することが可能である。

【0071】

以下、図10～図14のフローチャートを参照して、図1に示したコード生成部107によるコード生成処理について説明する。

【0072】

図10は、本発明の第2の制御処理手順の一例を示すフローチャートであり、図1に示したコード生成部107によるコード生成処理に対応する。なお、S201～S207は各ステップを示す。

【0073】

まず、ステップS201において、外部記憶装置104に記憶された入出力テーブル109から1行(1データ項目)を読み取り、ステップS202において、列6のIO区分の値を取得し、ステップS203において、取得したIO区分の値が「G1」であるか否かをチェックする。

【0074】

ステップS203で、列6のIO区分の値に「G1」が定義されていないと判定した場合には、そのままステップS205に進む。

【0075】

一方、ステップS203で、列6のIO区分の値に「G1」が定義されていると判定した場合には、ステップS204において、入出力テーブル109から列6のIO区分の値が「G2」となる行を全て取得してグループ化し、列2,列3,列4の情報をグループの「DB情報」としてRAM内のワークエリアに書き込む。

【0076】

次に、ステップS205において、画面表示項目,ファイル項目情報の取得処理(図11)を行い、ステップS206に進む。

【0077】

ステップS206において、入出力テーブル109内の全行(全データ項目)について、ステップS201～S205の処理を終了したか否かを判定し、まだ終了していないと判定した場合には、ステップS201の処理に戻る。

【0078】

一方、ステップS206で、入出力テーブル109内の全行(全データ項目)について、ステップS201～S205の処理を終了したと判定した場合には、ステップS207の処理に進む。

【0079】

次に、ステップS207において、JSP,サーブレット,Bean生成処理を行い、処理を終了し、図2のフローチャートに戻る。

【0080】

図11は、本発明の第3の制御処理手順の一例を示すフローチャートであり、図10に示したステップS205の画面表示項目,ファイル項目情報の取得処理に対応する。なお、S301～S304は各ステップを示す。

【0081】

まず、ステップS301において、列6のIO区分の値が「D」であるか否かをチェックする。

10

20

30

40

50

## 【0082】

ステップS301で、列6のIO区分の値に「D」が定義されていると判定した場合には、「画面表示項目指定情報」と判定して、ステップS302に進み、図10のステップS201で入出力テーブル109から読み出した1行のデータ(列2~列8)を「画面表示項目指定情報」として取得してRAM内のワークエリアに書き込み、ステップS303に進む。

## 【0083】

一方、ステップS301で、列6のIO区分の値に「D」が定義されていないと判定した場合には、そのままステップS303に進む。

## 【0084】

ステップS303で、列6のIO区分の値に「F」が定義されていると判定した場合には、ファイル項目定義情報と判定して、ステップS304に進み、図10のステップS201で入出力テーブル109から読み出した1行のデータ(列1~列7)を「ファイル項目定義情報」として取得してRAM内のワークエリアに書き込む。

## 【0085】

一方、ステップS303で、列5のIO区分の値に「F」が定義されていないと判定した場合には、そのまま処理を終了し、図10のフローチャートにリターンする。

## 【0086】

図12は、本発明の第4の制御処理手順の一例を示すフローチャートであり、図10のステップS207に示したJSP, サブレット, Bean生成処理に対応する。なお、S401~S403は各ステップを示す。

## 【0087】

まず、ステップS401において、外部記憶装置104に格納されたログイン情報110, アプリケーション名111, RAM内のワークエリアに記憶されたDB情報, ファイル項目情報に基づいて、各データベースへの接続, データ検索, データ更新を行うBean114を生成する(詳細は図13に示す)。

## 【0088】

次に、ステップS402において、外部記憶装置104に格納されたアプリケーション名111, RAMに記憶されたDB情報, 画面用情報に基づいて、ウェブブラウザからの入力データを取得してBean114, JSP112を呼び出しウェブアプリケーション全体を制御するサブレット113を生成する(詳細は図14に示す)。

## 【0089】

次に、ステップS403において、外部記憶装置104に格納されたアプリケーション名111, RAMに記憶されたDB情報, 画面用情報, ファイル項目情報に基づいて、図9に示した操作画面のウェブブラウザへの表示を制御するJSP112を生成し(詳細は図15に示す)、処理を終了して、図10のフローチャートに戻る。

## 【0090】

図13は、本発明の第5の制御処理手順の一例を示すフローチャートであり、図12のステップS401に示したBean生成処理に対応する。なお、S501~S505は各ステップを示す。

## 【0091】

まず、ステップS501において、Beanファイルを生成する。なお、Beanファイル名は、例えば、図示しない入力画面より入力されたアプリケーション名に文字列「Bean」を付加し拡張子を「java(登録商標)」としたBeanファイルとする。また、このとき、作成したBeanファイル内の先頭に、「パッケージ名(例えば、アプリケーション名に文字列「Bean」を付加したもの)の宣言、「java(登録商標).sql」パッケージのインポート、Java(登録商標)Beansを利用するために必要なインタフェースの実装等を行うコード、データベース接続オブジェクトの宣言、SQLステートメントオブジェクトの宣言、結果セットオブジェクトの宣言等を行うコード」を格納しておく。

10

20

30

40

50

## 【0092】

次に、ステップS502において、外部記憶装置104に格納されたログイン情報110，RAMに記憶されたDB情報に基づいて「各DBに対する各DB接続用メソッドに対応する各コード」をBeanファイル内に生成する。

## 【0093】

詳細には、「ログイン情報110内のDBシステム名601に対応するJDBCドライバ602を用いてドライバクラスをロードし、さらに、ログイン情報110内のDBシステム名601に対応するURL603，ユーザ名604，パスワード605を用いてデータベースへ接続する処理を行い、さらに、SQLステートメントオブジェクトを生成する処理を行う各DB毎のDB接続用メソッドに対応するコード」をBeanファイル内に生成する。

10

## 【0094】

次に、ステップS503において、RAMに記憶されたDB情報に基づいて「各DBに対する各DB切断用メソッドに対応する各コード」をBeanファイル内に生成する。

## 【0095】

詳細には、「SQLステートメントオブジェクトとデータベース接続オブジェクトをクローズする処理を行う各DB毎のDB切断用メソッドに対応するコード」をBeanファイル内に生成する。

## 【0096】

次に、ステップS504において、RAMに記憶されたDB情報，ファイル項目情報に基づいて「各テーブルに対する各データ検索用メソッドに対応するコード」をBeanファイル内に生成する。

20

## 【0097】

詳細には、「データファイル番号をテーブル名、該データファイル番号内のデータ項目枝番をテーブル内の項目名とし、セットされた検索条件によりテーブルデータを検索するSQL文を生成し、該SQL文のクエリを実行し検索結果を取得する各テーブル毎のデータ検索用メソッドに対応するコード」をBeanファイル内に生成する。

## 【0098】

次に、ステップS505において、RAMに記憶されたDB情報，ファイル項目情報に基づいて「各テーブルに対する各データ更新用メソッド」をBeanファイル内に生成する。

30

## 【0099】

詳細には、「データファイル番号をテーブル名、該データファイル番号内のデータ項目枝番をテーブル内の項目名とし、セットされた更新データによりテーブルデータを更新するSQL文を生成し、該SQL文のクエリを実行する各テーブル毎のデータ更新用メソッドに対応するコード」をBeanファイル内に生成する。

## 【0100】

そして、処理を終了し、図12のフローチャートにリターンする。

## 【0101】

図14は、本発明の第6の制御処理手順の一例を示すフローチャートであり、図1に示したコード生成部107によるサブレット作成処理（図12のステップS402）に対応する。なお、S601～S607は各ステップを示す。

40

## 【0102】

まず、ステップS601において、サブレットファイルを生成する。このサブレットファイルのファイル名は、例えば、アプリケーション名に文字列「Servlet」を付加し拡張子を「java（登録商標）」とする。なお、このとき、サブレットファイル内の先頭に、「使用するBeanパッケージ及びその他必要なパッケージをインポートするコード、アクセス修飾子、クラスの定義、クラス（例えば、HttpServletクラス）の継承、ウェブブラウザ（HTMLフォーム）から送信されたパラメータ（送信情報）を受け取るためのメソッドを示すコード」を格納しておく。

50

## 【0103】

次に、ステップS602において、RAMに記憶された画面表示項目情報に基づいて「ウェブブラウザからの送信情報を受け取るコード」をサブレットファイル内に生成する。

## 【0104】

次に、ステップS603において、RAMに記憶されたDB情報に基づいて「切り替え検索が指示された場合、Beanのメソッドを用いて、テーブル切り替え検索処理を行うコード」をサブレットファイル内に生成する。

## 【0105】

詳細には、「ウェブブラウザからの送信情報に基づいてユーザからの指示に基づいて切り替え検索が指示された（ユーザにより切り替え検索ボタン202が指示された）と判定した場合、さらに、切り替え先のテーブルが現在接続されているDBか判定し、現在接続されているDBでない場合、現在接続されているDBのDB切断用メソッドにより切断し、切り替え先のテーブルのDBのDB接続用メソッドにより切り替え先のテーブルのDBに接続し、接続したDBの情報を所定の記憶領域（セッションオブジェクト）に保存し、切り替え先のテーブルのデータ検索用メソッドを用いて、ウェブブラウザから取得した検索条件での検索処理を実行し、一方、切り替え先のテーブルが現在接続されているDBの場合には、切り替え先のテーブルのデータ検索用メソッドを用いて、ウェブブラウザから取得した検索条件での検索処理を実行するコード」をサブレットファイル内に生成する。

10

20

## 【0106】

次に、ステップS604において、RAMに記憶されたDB情報に基づいて「検索が指示された場合、Beanのメソッドを用いて、データ検索処理を行うコード」をサブレットファイル内に生成する。

## 【0107】

詳細には、「ウェブブラウザからの送信情報に基づいてユーザからの指示に基づいて検索が指示された（ユーザにより検索ボタン204が指示された）と判定した場合、さらに、現在のテーブルのDBが接続されているか判定し、現在のテーブルのDBが接続されていない場合は、現在のテーブルのDBに対応するDB接続用メソッドにより現在のテーブルのDBに接続し、接続したDBの情報を所定の記憶領域（セッションオブジェクト）に保存し、現在のテーブルのデータ検索用メソッドを用いて、ウェブブラウザから取得した検索条件での検索処理を実行し、一方、現在のテーブルのDBが接続されている場合は、そのまま、現在のテーブルのデータ検索用メソッドを用いて、ウェブブラウザから取得した検索条件での検索処理を実行するコード」をサブレットファイル内に生成する。

30

## 【0108】

次に、ステップS605において、RAMに記憶されたDB情報に基づいて「更新が指示された場合、Beanのメソッドを用いて、データ更新処理を行うコード」をサブレットファイル内に生成する。

## 【0109】

詳細には、「ウェブブラウザからの送信情報に基づいてユーザからの指示に基づいて更新が指示された（ユーザにより更新ボタン206が指示された）と判定した場合、さらに、現在のテーブルのDBが接続されているか判定し、現在のテーブルのDBが接続されていない場合は、現在のテーブルのDBに対応するDB接続用メソッドにより現在のテーブルのDBに接続し、接続したDBの情報を所定の記憶領域（セッションオブジェクト）に保存し、現在のテーブルのデータ更新用メソッドを用いて、ウェブブラウザから取得した更新データの更新処理を実行し、一方、現在のテーブルのDBが接続されている場合は、そのまま、現在のテーブルのウェブブラウザから取得した更新データの更新処理を実行するコード」をサブレットファイル内に生成する。

40

## 【0110】

次に、ステップS606において、「Beanを所定の記憶領域（例えば、セッション

50

オブジェクト)に保存するコード」をサーブレットファイル内に生成する。

【0111】

次に、ステップS607において、「JSPを呼び出すコード」をサーブレットファイル内に生成し、処理を終了して図12のフローチャートにリターンする。

【0112】

図15は、本発明の第7の制御処理手順の一例を示すフローチャートであり、図1に示したコード生成部107によるJSP作成処理(図12のステップS403)に対応する。なお、S701~S710は各ステップを示す。

【0113】

まず、ステップS701において、JSPファイルを生成する。このJSPファイルのファイル名は、例えば、アプリケーション名に拡張子を「jsp」とする。なお、このJSPファイル内には、「JSPページのMIMEタイプとエンコードするキャラクタセットを設定するコード、当該データファイルに対応するBeanをインスタンスするコード、htmlのタグである「<html><head><title>アプリケーション名</title>」等のコード」を生成し格納しておくものとする。

10

【0114】

次に、ステップS702において、「所定の記憶領域(セッションオブジェクト)からBeanを取得するコード」をJSPファイル内に生成する。

【0115】

次に、htmlのタグである「</head><body>」等を格納した後、ステップS703において、「フォームのパラメータが送信されたときに、起動するサーブレットを指定するコード」を生成する。詳細には、htmlのタグである「<form method="post" action=サーブレットのurl>」等のコードを格納する。

20

【0116】

次に、ステップS704において、RAMに記憶されたDB情報に基づいて「選択メニュー201を生成するコード」をJSPファイル内に生成する。

【0117】

次に、ステップS705において、「フォームのパラメータを送信する機能を備える切り替え検索ボタン202を生成するコード」をJSPファイル内に生成する。

30

【0118】

次に、ステップS706において、RAMに記憶された画面表示項目情報(特に「S」「S,H」のある項目)に基づいて「検索条件入力欄203を生成するコード」をJSPファイル内に生成する。なお、この検索条件入力欄203を生成するコードは、画面表示項目情報内で指定される各項目の桁数に応じて各項目の入力欄を生成するにコードとする。

【0119】

次に、ステップS707において、「フォームのパラメータを送信する機能を備える検索ボタン202を生成するコード」をJSPファイル内に生成する。

【0120】

40

次に、ステップS708において、RAMに記憶された画面表示項目情報(特に「S,H」「H」のある項目)に基づいて「検索結果表示入力欄205を生成するコード」をJSPファイル内に生成する。なお、この検索結果表示入力欄205を生成するコードは、画面表示項目情報内で指定される各項目の桁数に応じて各項目の表示入力欄を生成するにコードとする。

【0121】

次に、ステップS709において、「フォームのパラメータを送信する機能を備える更新ボタン206を生成するコード」をJSPファイル内に生成する。

【0122】

次に、ステップS710において、RAM内に格納された画面表示項目情報に基づいて

50

「Bean内に検索条件情報が格納されている場合、該検索条件情報を検索条件入力欄203にセットし、また、Bean内に検索結果情報が格納されている場合、該検索結果情報を検索結果表示入力欄205にセットするコード」をJSPファイル内に生成する。

【0123】

最後に、htmlのタグである"`</form></body></html>`"等を格納し、処理を終了して図12のフローチャートにリターンする。なお、このJSP内に、Java(登録商標)Script等の各種スクリプトコード等も含むように構成してもよい。

【0124】

図4に示すように、テーブルの定義と画面構成を業務仕様の一部として定義することにより、DB接続切り替え、操作に関するコードと画面を自動生成することができる。これにより、業務仕様の変更などに柔軟に対応でき、設計・実装・管理を一元管理し高品質でスピーディーなウェブアプリケーションの開発が可能となる。

10

【0125】

以下、図4に示した入出力テーブル109の内容にそって、図10～図15に示した処理による、ユーザが動的にDB・テーブルを切り替え可能な操作画面(図9)の生成例について示す。

【0126】

図4に示した入出力テーブル109の行1では、列6に「G1」が定義されおり、同じ列6に「G2」が定義されている行2,行3の項目をまとめて仮想的に一つのテーブルとする処理が行われる。

20

【0127】

また、行2では、列6に「G2」が定義されていることから、列3で定義している「CliniDB/ClinicM」を画面上で選択可能なテーブルとして使用できることを示している。

【0128】

同様に、行3も「DruDB/DrugstoreM」が選択可能なテーブルとして定義されている。

【0129】

さらに、行4～行10では、列6に「D」が定義されているので、実際に画面の表示項目となることを示している。それぞれ列5のデータファイル項目名が画面に表示され、列7で指定した桁数分の入力項目が生成される。なお、生成される画面は検索用とその結果表示用という構成になっている。検索用(図9の検索条件入力欄203)には、列8で「S」又は「S,H」が指定されている行4～行7までの項目が表示され、結果表示用(図9の検索結果表示入力欄205)には「H」又は「S,H」が指定されている行6～行10の項目が表示される。

30

【0130】

また、行11～行24では、列6に「F」が定義されているので、ファイル定義であることを示している。このファイル定義は行2,行3で定義されている「病院」と「薬局」の二つのDB・テーブルについて定義しており、行11～行17が「病院」を示し、行18～行24が「薬局」を示している。

40

【0131】

そして、上記図4に示した入出力テーブル109に基づく図10～図15の処理の結果、実行時に図9のような操作画面を表示するウェブアプリケーションが生成される。

【0132】

まず、図4に示した入出力テーブル109の行1の定義より、この画面がDB・テーブルを動的に切り替えることが出来る機能がある画面ということになり、図9に示すようなDB・テーブルの選択メニュー201と切り替え確認のための切り替え検索ボタン202が図1に示したコード生成部107により自動的に生成される。

【0133】

50

選択メニュー201に表示されているテーブル201a, 201bは、図1の行2, 行3で定義されているDB・テーブルとなる。例えば、ここで「Clinic(病院)」201aを選択し、切り替え検索ボタン202を押下すれば検索の対象は図4に示した入出力テーブル109の行11~行17で定義されている「CLiDB」DBシステムの「CLiNICM」テーブルとなる。

【0134】

また、検索するためのキーを入力する項目は、図4に示した入出力テーブル109の行4~行7の項目となっている。

【0135】

これにより生成された検索条件入力欄203に必要な値を入力し、検索ボタン204を指示すると、指定されたテーブルを検索し、図9に示す検索結果表示入力欄205に結果一覧が表示される。

【0136】

この検索結果表示入力欄205は、図4に示した入出力テーブル109の行6~行10で定義された項目が表示される。なお、図4に示した入出力テーブル109で利用したアプリケーションにおけるDBの構成は図6に示すDB構成に対応する。

【0137】

なお、この図4に示した入出力テーブル109を用いた生成例は、異なる外部DBシステム間での切り替えに対応するが、図4に示した入出力テーブル109の定義を変更すれば、図6に示したような外部DBを切り替える機能以外にも、図8に示したような内部DB内のテーブルを切り替える機能や、図9に示したような内部と外部DBの切り替えを行う機能を付加することが可能である。

【0138】

図16は、本発明のプログラム生成装置における業務仕様情報(図1に示した入出力テーブル109)について説明する模式図であり、同一のDBシステム(内部DB又は外部DB)内でテーブル切り替えを行う場合に対応する。

【0139】

例えば、入出力定義を図16に示した入出力テーブルのように行2, 行3の列3の値を「CLiDB/CLiNICM」, 「CLiDB/DrugstoreM」とし、「/」記号の前のDBシステム名称を同一の「CLiDB」として記述し、図16の行11~行24のファイル定義も列1のDBシステム名を「CLiDB」と同一とする。このように定義を変更した場合は、同一のDB(内部DBのみ又は外部DBのみ)の「CLiNICM」テーブルと「DrugstoreM」テーブルを交互に切り替えることになる。内部DBのみのDB構成は、図7に示した例に対応する。

【0140】

また、図4の行2の列3に示した「CLiDB」を外部DBとし、図4の行3の列3の「DrudB」を内部DBとする。なお、内部DB, 外部DBの区別は、図5に示したロケイン情報110のURL603で定義される。

【0141】

このようなDB構成の場合は、外部DB「CLiDB」のテーブル「CLiNICM」と内部DB「DrudB」のテーブル「DrugstoreM」を交互に切り替えることが可能となる。このような外部DB及び内部DBからなるDB構成は、図8に示した例に対応する。

【0142】

以上示したように、DBの接続方法(URL指定(ネットワーク接続でもローカルでもよい)等)とDBのテーブル構成を業務仕様を入力として、DBの選択画面プログラム(JSP112)を生成するステップと、DBへの接続(割り当て)処理プログラム(サーブレット113)を生成するステップと、DBへのアクセス処理プログラム(Bean114)を生成するステップとを備えることにより、ウェブアプリケーション開発支援システムにおいて、ネットワーク接続された複数のDBをユーザ選択により動的に切り替えて

10

20

30

40

50

アクセスする機能、又は、同一DB上の複数のテーブルをユーザ選択により動的に切り替えてアクセスできる機能を、業務仕様の枠組みの中で定義し、プログラムコードを自動生成することができる。

【0143】

従って、生成された操作画面上で対象となるDB上のテーブルへの割り当てとアクセス処理をユーザが動的に切り替えることができる機能を有するアプリケーションプログラムを、DB接続のための定義を業務仕様の枠組みの中で定義して自動生成することができ、生産性、保守性、管理性を著しく向上することができる。

【0144】

また、分散したDBに対してユーザの選択による動的に接続を切り替え、また該テーブルへのアクセス処理が自動生成可能となったことで、生産性と保守性に優れたWebアプリケーションプログラムの領域を広げることができる。

10

【0145】

同様な定義・生成方式の枠組みの中で、同一DBで異なるテーブルへの接続、アクセス処理を自動生成可能とした。

【0146】

また、上記実施形態中では、2つのDB・テーブルを例にして、該2つのDB・テーブルを切り替え可能なウェブアプリケーションの生成方法について説明したが、切り替え可能なDB・テーブルは2つに限られるものではなく、入出力テーブル109に切り替え可能な3つ以上のDB・テーブルを定義し、ログイン情報110に対応するDBの接続情報をそれぞれ定義することにより、3つ以上のDB・テーブルを切り替え可能なウェブアプリケーションを生成することができる。

20

【0147】

また、ログイン情報110に、DBシステム名、JDBCドライバ名、URL、ユーザ名、パスワードを記憶させる場合について説明したが、JDBCドライバ名、URL、ユーザ名、パスワードの代わりに、DBシステムのオブジェクト名を記憶させるように構成してもよい。その場合、DBシステムのオブジェクト名からDBシステムを検索して接続するようなDB接続用メソッドを生成するように構成する。

【0148】

また、図4、図5、図16の各種データの構成及びその内容はこれに限定されるものではなく、用途や目的に応じて、様々な構成や内容で構成されることは言うまでもない。

30

【0149】

以上、一実施形態について示したが、本発明は、例えば、システム、装置、方法、プログラムもしくは記録媒体等としての実施態様をとることが可能であり、具体的には、複数の機器から構成されるシステムに適用しても良いし、また、一つの機器からなる装置に適用しても良い。

【0150】

また、本発明のプログラム生成装置により生成されるプログラムコードを、ASP(Active Server Pages)、PHP(Hypertext Preprocessor)等のJava(登録商標)以外の他のプログラムによるウェブアプリケーションとすることで、本発明のプログラム生成装置を各種ウェブアプリケーションの生成にも適用可能である。

40

【0151】

さらに、上記実施形態では、入出力テーブル109で指定されるデータベース上の複数のテーブルをそれぞれ定義する複数の定義情報、及びログイン情報110で指定される前記データベースへ接続するための接続情報に基づいて「前記複数のテーブルを動的に切り替えてアクセス可能なプログラムコード」を生成するプログラム生成装置について説明したが、複数のテーブルを動的に切り替えてアクセスすることができるデータベースアクセス装置(方法)そのものも本発明に含まれるものである。

【0152】

50



さらに、上記各実施形態では、ウェブアプリケーションサーバ上で動作するウェブアプリケーションプログラムを生成するプログラム生成装置について説明したが、ミドルウェアとしてウェブアプリケーションサーバの代わりに、ノーツ（商品名）・アプリケーションサーバやドミノ（商品名）・アプリケーションサーバ等のグループウェアを利用したアプリケーションを生成するように構成してもよい。

【0153】

この場合、サブレットの代わりに、エージェントと呼ばれるJava（登録商標）で作成されたプログラムをプログラム生成装置で生成し、グループウェアのアプリケーションサーバに配置するものとする。また、JSPにより表示していた入出力画面は、フォームと呼ばれるコンテンツとして、プログラム生成装置で生成して、グループウェアのデータベースに登録しておく。また、Beanは上記実施形態と同一のものを使用可能である。

10

【0154】

また、本発明のプログラム生成装置が生成するアプリケーションは、PDA（携帯情報端末）や携帯電話といった携帯端末をクライアントとするアプリケーションとしてもよい。

【0155】

そして、クライアントとの間のプロトコルもHTTPのみではなく、WAP（Wireless Application Protocol）等とし、JSPにより生成されるコードも、HTMLに限られるものではなく、XML、WML等のその他の構造化言語とするような、プログラムを、本発明のプログラム生成装置が生成するように構成してもよい。

20

【0156】

また、上記実施形態では、プログラム生成装置が、データベースへのアクセスを行うためのBean、各データ項目の入出力画面を選択された画面パターンで表示するためのJSP、入出力画面からの入力情報を取得して該入力情報に基づいてBean、JSPの実行を制御してアプリケーション全体を統括制御するサブレットから構成されるウェブアプリケーションを生成する場合について説明したが、上記Bean、JSPの機能をも含んだサブレットのみ、上記Beanの機能を含んだサブレットとJSPのみ、又は、上記JSPの機能を含んだサブレットとBeanのみ、から構成されるウェブアプリケーションを生成するように構成してもよいことはいうまでもない。

30

【0157】

さらに、上記実施形態内に記載された各変形例を組み合わせた構成も全て本発明に含まれるものである。

【0158】

以下、図17に示すメモリマップを参照して本発明に係るプログラム生成装置で読み取り可能なデータ処理プログラムの構成について説明する。

【0159】

図17は、本発明に係るプログラム生成装置で読み取り（読み出し）可能な各種データ処理プログラムを格納する記録媒体（記憶媒体）のメモリマップを説明する図である。

40

【0160】

なお、特に図示しないが、記録媒体に記憶されるプログラム群を管理する情報、例えばバージョン情報、作成者等も記憶され、かつ、プログラム読み出し側のOS等に依存する情報、例えばプログラムを識別表示するアイコン等も記憶される場合もある。

【0161】

さらに、各種プログラムに従属するデータも上記ディレクトリに管理されている。また、インストールするプログラムやデータが圧縮されている場合に、解凍するプログラム等も記憶される場合もある。

【0162】

本実施形態における図3、図10、図11、図12、図13、図14、図15に示す機

50

能が外部からインストールされるプログラムによって、ホストコンピュータにより遂行されていてもよい。そして、その場合、CD-ROMやフラッシュメモリやFD等の記録媒体により、あるいはネットワークを介して外部の記録媒体から、プログラムを含む情報群を出力装置に供給される場合でも本発明は適用されるものである。

【0163】

以上のように、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（またはCPUやMPU）が記録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、本発明の目的が達成されることは言うまでもない。

【0164】

この場合、記録媒体から読み出されたプログラムコード自体が本発明の新規な機能を実現することになり、そのプログラムコードを記憶した記録媒体は本発明を構成することになる。

【0165】

プログラムコードを供給するための記録媒体としては、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、CD-ROM、CD-R、DVD-ROM、磁気テープ、不揮発性のメモリカード、ROM、EEPROM、シリコンディスク等を用いることができる。

【0166】

また、コンピュータが読み出したプログラムコードを実行することにより、前述した実施形態の機能が実現されるだけでなく、その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OS（オペレーティングシステム）等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ことは言うまでもない。

【0167】

さらに、記録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、そのプログラムコードの指示に基づき、その機能拡張ボードや機能拡張ユニットに備わるCPU等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ことは言うまでもない。

【0168】

また、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、1つの機器からなる装置に適用してもよい。また、本発明は、システムあるいは装置にプログラムを供給することによって達成される場合にも適用できることは言うまでもない。この場合、本発明を達成するためのソフトウェアによって表されるプログラムを格納した記録媒体を該システムあるいは装置に読み出すことによって、そのシステムあるいは装置が、本発明の効果を享受することが可能となる。

【0169】

さらに、本発明を達成するためのソフトウェアによって表されるプログラムをネットワーク上のサーバ、データベース等から通信プログラムによりダウンロードして読み出すことによって、そのシステムあるいは装置が、本発明の効果を享受することが可能となる。

【産業上の利用可能性】

【0170】

本発明のプログラム生成装置により生成されるプログラムコードを、ASP（Active Server Pages）、PHP（Hypertext Preprocessor）等のJava（登録商標）以外の他のプログラムによるアプリケーションとすることで、本発明のプログラム生成装置を各種アプリケーションの生成にも適用可能である。

【図面の簡単な説明】

【0171】

【図 1】本発明の一実施形態を示すプログラム生成装置の構成を示すブロック図である。

【図 2】図 1 に示したコード生成部によって生成したサーブレット、J S P、および B e a n を利用したウェブアプリケーションシステムの全体構成の一例を示す図である。

【図 3】本発明のプログラム生成装置における第 1 の制御処理手順の一例を示すフローチャートである。

【図 4】本発明のプログラム生成装置における業務仕様情報（図 1 に示した入出力テーブル）について説明する模式図である。

【図 5】図 1 に示したログイン情報を説明するための模式図である。

【図 6】本発明のプログラム生成装置により生成されるアプリケーションによる D B 接続切り替えを説明するための D B 構成（外部（リモート）分散 D B 構成）を示す図である。

10

【図 7】本発明のプログラム生成装置により生成されるアプリケーションによる D B 接続切り替えを説明するための D B 構成（内部（ローカル）同一 D B 構成）を示す図である。

【図 8】本発明のプログラム生成装置により生成されるアプリケーションによる D B 接続切り替えを説明するための D B 構成（内部（ローカル）D B 及び外部（リモート）D B による分散 D B 構成）を示す図である。

【図 9】本発明のプログラム生成装置により生成されるウェブアプリケーションにおける操作画面の一例を示す模式図である。

【図 10】本発明の第 2 の制御処理手順の一例を示すフローチャートである。

【図 11】本発明の第 3 の制御処理手順の一例を示すフローチャートである。

【図 12】本発明の第 4 の制御処理手順の一例を示すフローチャートである。

20

【図 13】本発明の第 5 の制御処理手順の一例を示すフローチャートである。

【図 14】本発明の第 6 の制御処理手順の一例を示すフローチャートである。

【図 15】本発明の第 7 の制御処理手順の一例を示すフローチャートである。

【図 16】本発明のプログラム生成装置における業務仕様情報（図 1 に示した入出力テーブル）について説明する模式図である。

【図 17】本発明に係るプログラム生成装置で読み取り（読み出し）可能な各種データ処理プログラムを格納する記録媒体（記憶媒体）のメモリマップを説明する図である。

【符号の説明】

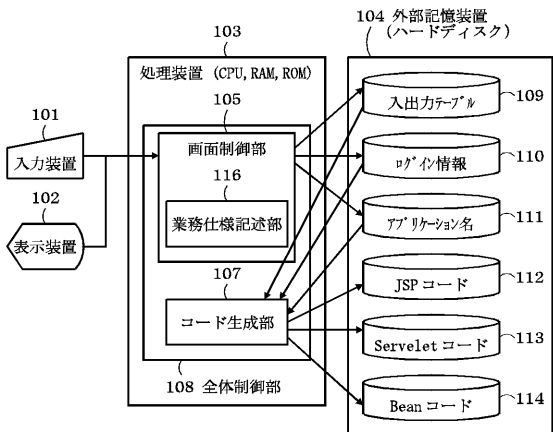
【 0 1 7 2 】

- 1 0 1 入力装置
- 1 0 2 表示装置
- 1 0 3 処理装置（C P U , R A M , R O M ）
- 1 0 4 外部記憶装置（ハードディスク）
- 1 0 5 画面制御部
- 1 1 6 業務仕様記述部
- 1 0 7 コード生成部
- 1 0 8 全体制御部
- 1 0 9 入出力テーブル
- 1 1 0 ログイン情報
- 1 1 1 アプリケーション名
- 1 1 2 J S P コード
- 1 1 3 S e r v l e t コード
- 1 1 4 B e a n コード

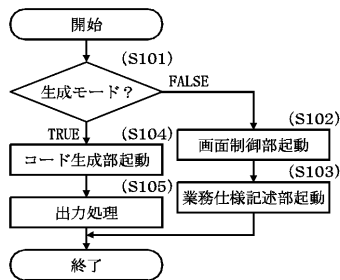
30

40

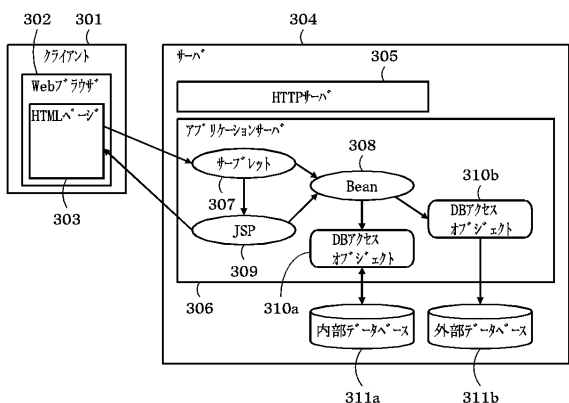
【図1】



【図3】



【図2】



【図4】

入力テーブル名一覧

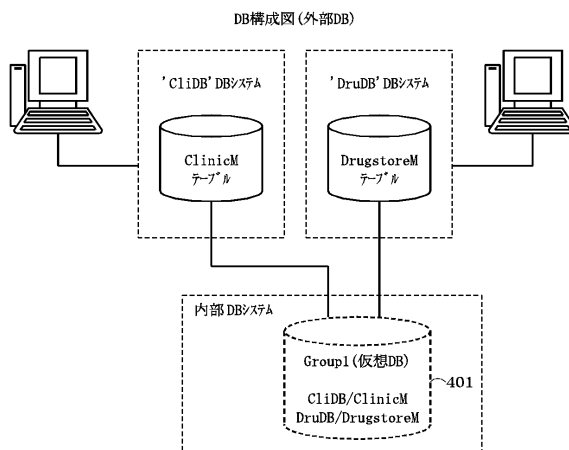
列1	列2	列3	列4	列5	列6	列7	列8	列9
データファイル番号	データファイル項目枝番	データファイル名	データファイル項目名	IO区分	桁	表示	項番	
行1	DaiGroup	Group1			G1			1
行2	Group1	ClidB/ ClinicM		病院	G2			2
行3	Group1	DruDB/ DrugstoreM		薬局	G2			3
行4	DaiGroup	KenCode		県コード	D	3	S	4
行5	DaiGroup	Sisetu		施設名(カナ)	D	30	S	5
行6	DaiGroup	SisetumeiK		施設名(漢字)	D	20	S,H	6
行7	DaiGroup	Jyusyo		住所	D	30	S,H	7
行8	DaiGroup	Tel		電話番号	D	15	H	8
行9	DaiGroup	Zip		郵便番号	D	7	H	9
行10	DaiGroup	DsfCode		DSFコード	D	3	H	10
行11	ClidB	ClinicM	KenCode	病院	県コード	F	3	11
行12	ClidB	ClinicM	Sisetumei	病院	施設名(カナ)	F	30	12
行13	ClidB	ClinicM	SisetumeiK	病院	施設名(漢字)	F	20	13
行14	ClidB	ClinicM	Jyusyo	病院	住所	F	30	14
行15	ClidB	ClinicM	Tel	病院	電話番号	F	15	15
行16	ClidB	ClinicM	Zip	病院	郵便番号	F	7	16
行17	ClidB	ClinicM	DsfCode	病院	DSFコード	F	3	17
行18	DruDB	DrugstoreM	KenCode	薬局	県コード	F	3	18
行19	DruDB	DrugstoreM	Sisetumei	薬局	施設名(カナ)	F	30	19
行20	DruDB	DrugstoreM	SisetumeiK	薬局	施設名(漢字)	F	20	20
行21	DruDB	DrugstoreM	Jyusyo	薬局	住所	F	30	21
行22	DruDB	DrugstoreM	Tel	薬局	電話番号	F	15	22
行23	DruDB	DrugstoreM	Zip	薬局	郵便番号	F	7	23
行24	DruDB	DrugstoreM	DsfCode	薬局	DSFコード	F	3	24

【図5】

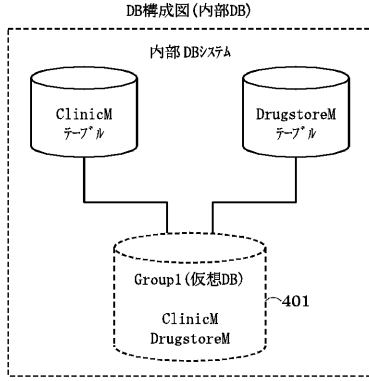
601	602	603	604	605
DBシステム名	JDBCドライバ	URL	ユーザー名	パスワード
ClidB	.....Driver	jdbc :..... ClidB	xxx	xxxx00
DruDB	.....Driver	jdbc :..... DruDB	yyyyy	yy999

110

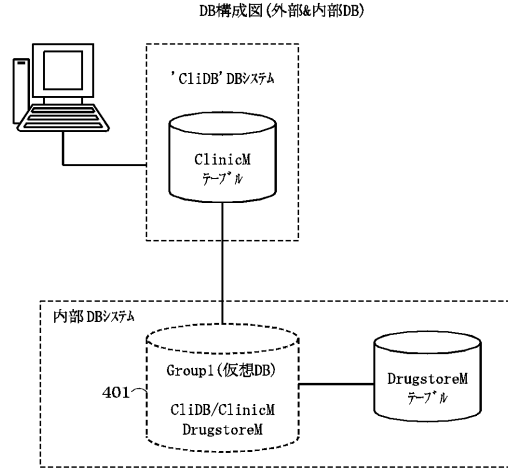
【図6】



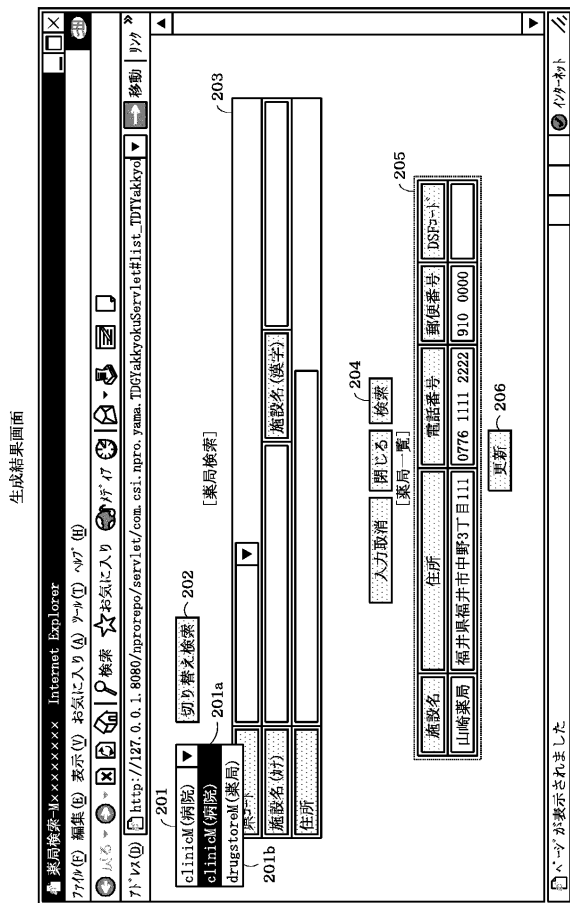
【図7】



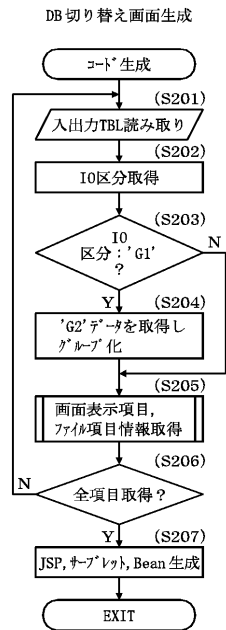
【図8】



【図9】

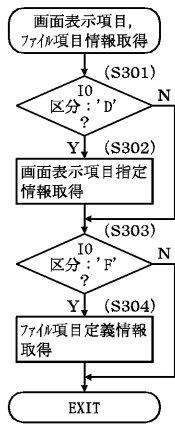


【図10】



【図11】

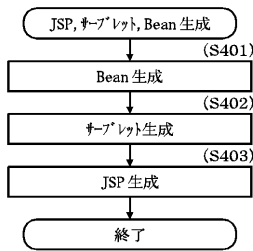
DB切り替え画面生成



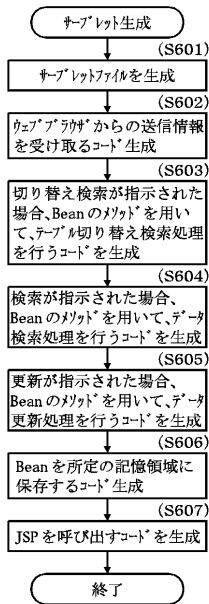
【図13】



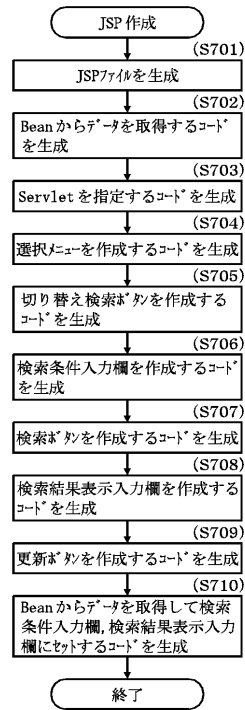
【図12】



【図14】



【図15】



【図16】

入出力テーブルイメージ(同一DBのみ(内部同一DBのみ又は外部同一DBのみ))

	列1	列2	列3	列4	列5	列6	列7	列8	列9
	データファイル番号	データファイル項目枝番	データファイル名	データファイル項目名	IO区分	桁	表示		項番
行1	DaiGroup	Group1			G1				1
行2	Group1	CLiDB/ ClinicM		病院	G2				2
行3	Group1	CLiDB/ DrugstoreM		薬局	G2				3
行4	DaiGroup	KenCode		県コード	D	3	S		4
行5	DaiGroup	Sisetu		施設名(カナ)	D	30	S		5
行6	DaiGroup	SisetumeiK		施設名(漢字)	D	20	S,H		6
行7	DaiGroup	Jyusyo		住所	D	30	S,H		7
行8	DaiGroup	Tel		電話番号	D	15	H		8
行9	DaiGroup	Zip		郵便番号	D	7	H		9
行10	DaiGroup	DsfCode		DSFコード	D	3	H		10
行11	CLiDB	ClinicM	KenCode	病院	県コード	F	3		11
行12	CLiDB	ClinicM	Sisetumei	病院	施設名(カナ)	F	30		12
行13	CLiDB	ClinicM	SisetumeiK	病院	施設名(漢字)	F	20		13
行14	CLiDB	ClinicM	Jyusyo	病院	住所	F	30		14
行15	CLiDB	ClinicM	Tel	病院	電話番号	F	15		15
行16	CLiDB	ClinicM	Zip	病院	郵便番号	F	7		16
行17	CLiDB	ClinicM	DsfCode	病院	DSFコード	F	3		17
行18	CLiDB	DrugstoreM	KenCode	薬局	県コード	F	3		18
行19	CLiDB	DrugstoreM	Sisetumei	薬局	施設名(カナ)	F	30		19
行20	CLiDB	DrugstoreM	SisetumeiK	薬局	施設名(漢字)	F	20		20
行21	CLiDB	DrugstoreM	Jyusyo	薬局	住所	F	30		21
行22	CLiDB	DrugstoreM	Tel	薬局	電話番号	F	15		22
行23	CLiDB	DrugstoreM	Zip	薬局	郵便番号	F	7		23
行24	CLiDB	DrugstoreM	DsfCode	薬局	DSFコード	F	3		24

【図17】

FD/CD-ROM等の記憶媒体

ディレクトリ情報	
第1のデータ処理プログラム	図2に示すフローチャートのステップに対応するプログラムコード群
第2のデータ処理プログラム	図10に示すフローチャートのステップに対応するプログラムコード群
第3のデータ処理プログラム	図11に示すフローチャートのステップに対応するプログラムコード群
第4のデータ処理プログラム	図12に示すフローチャートのステップに対応するプログラムコード群
第5のデータ処理プログラム	図13に示すフローチャートのステップに対応するプログラムコード群
第6のデータ処理プログラム	図14に示すフローチャートのステップに対応するプログラムコード群
第7のデータ処理プログラム	図15に示すフローチャートのステップに対応するプログラムコード群

記憶媒体のメモリマップ

---

フロントページの続き

- (56)参考文献 新井 正広, J a v a 統合開発環境のいま フリーから商用まで総チェック 商用編, J A V A  
P R E S S , 日本, (株)技術評論社, 2 0 0 1年 2月10日, 第16巻, p.48-55  
B S I のデータマート短期構築ソリューション~短期間でのデータ活用術, ネットワークコンピ  
ューティング, 日本, 株式会社リックテレコム, 1 9 9 9年 7月 1日, 第11巻, 第7号, p.10  
1-106

- (58)調査した分野(Int.Cl., DB名)  
G 0 6 F 1 2 / 0 0