

(19)日本国特許庁(JP)

(12)特許公報(B2)

(11)特許番号  
特許第7121313号  
(P7121313)

(45)発行日 令和4年8月18日(2022.8.18)

(24)登録日 令和4年8月9日(2022.8.9)

(51)国際特許分類 F I  
G 0 6 F 8/38 (2018.01) G 0 6 F 8/38

請求項の数 16 (全26頁)

|           |                              |          |   |
|-----------|------------------------------|----------|---|
| (21)出願番号  | 特願2020-187779(P2020-187779)  | (73)特許権者 | 390002761<br>キャノンマーケティングジャパン株式会社<br>東京都港区港南2丁目16番6号 |
| (22)出願日   | 令和2年11月11日(2020.11.11)       | (73)特許権者 | 592135203<br>キャノンITソリューションズ株式会社<br>東京都港区港南2丁目16番6号  |
| (62)分割の表示 | 特願2017-81667(P2017-81667)の分割 | (74)代理人  | 100189751<br>弁理士 木村 友輔                              |
| 原出願日      | 平成29年4月17日(2017.4.17)        | (72)発明者  | 鈴木 智子<br>東京都品川区東品川2丁目4番11号<br>キャノンITソリューションズ株式会社内   |
| (65)公開番号  | 特開2021-15644(P2021-15644A)   | (72)発明者  | 柴本 文洋<br>東京都品川区東品川2丁目4番11号<br>最終頁に続く                |
| (43)公開日   | 令和3年2月12日(2021.2.12)         |          |   |
| 審査請求日     | 令和2年12月11日(2020.12.11)       |          |   |

(54)【発明の名称】 情報処理装置とその処理方法及びプログラム

(57)【特許請求の範囲】

【請求項1】

特定のプログラムで表示される画面の遷移関係を特定する特定手段と、

所定の指示を受け付けたことに応じて、前記特定手段で特定した遷移関係に基づき、前記特定のプログラムで表示される複数の画面のうち、前記複数の画面のそれぞれを基準として、遷移回数が所定数を超える遷移元と遷移先の画面の情報を含むことなく、遷移回数が前記所定数までの遷移元と遷移先の少なくとも一方の画面の情報を含む遷移図をそれぞれ生成するように制御する制御手段と、

前記制御手段による制御で生成された前記遷移図を表示するように制御する表示制御手段と

を有することを特徴とする情報処理装置。

【請求項2】

前記所定の指示は、ユーザによる指示であることを特徴とする請求項1に記載の情報処理装置。

【請求項3】

前記所定の指示は、前記特定のプログラムを生成する指示であることを特徴とする請求項1または2に記載の情報処理装置。

【請求項4】

前記特定のプログラムを生成する生成手段をさらに有し、

前記制御手段は、前記生成手段により生成される特定のプログラムで表示される複数の

画面のそれぞれを基準とする遷移図をそれぞれ生成するように制御することを特徴とする請求項 1 乃至 3 のいずれか 1 項に記載の情報処理装置。

【請求項 5】

前記特定手段は、前記特定のプログラムを生成するための定義情報に基づいて前記遷移関係を特定することを特徴とする請求項 1 乃至 4 のいずれか 1 項に記載の情報処理装置。

【請求項 6】

前記特定手段は、各画面に対応する入出力定義に従って、各画面の遷移関係を特定することを特徴とする請求項 1 乃至 5 のいずれか 1 項に記載の情報処理装置。

【請求項 7】

前記特定手段は、前記入出力定義に含まれるアクション情報に従って、各画面の遷移関係を特定することを特徴とする請求項 6 に記載の情報処理装置。

10

【請求項 8】

前記特定手段は、前記アクション情報に定義される次入出力情報に従って、各画面の遷移関係を特定することを特徴とする請求項 7 に記載の情報処理装置。

【請求項 9】

前記特定のプログラムは、Webシステムで実行するプログラムであることを特徴とする請求項 1 乃至 8 のいずれか 1 項に記載の情報処理装置。

【請求項 10】

前記複数の画面は、前記Webシステムで用いるクライアント装置で表示される画面であることを特徴とする請求項 9 に記載の情報処理装置。

20

【請求項 11】

前記表示制御手段は、前記特定の画面ごとの仕様を表示するドキュメントを生成し、生成された前記ドキュメントと共に、前記遷移図を表示するよう制御することを特徴とする請求項 1 乃至 10 のいずれか 1 項に記載の情報処理装置。

【請求項 12】

前記ドキュメントには、前記特定の画面の画面イメージ、画面項目一覧の情報を含むことを特徴とする請求項 11 に記載の情報処理装置。

【請求項 13】

前記所定数は、前記特定のプログラムの開発者によって予め入力された情報に基づく数であることを特徴とする請求項 1 乃至 12 のいずれか 1 項に記載の情報処理装置。

30

【請求項 14】

前記所定数が設定された設定ファイルを読み込む読み込み手段を更に有し、  
前記表示制御手段は、前記設定ファイルに設定された前記所定数に基づいて前記遷移図を表示するよう制御することを特徴とする請求項 1 乃至 13 のいずれか 1 項に記載の情報処理装置。

【請求項 15】

特定のプログラムで表示される画面の遷移関係を特定する特定ステップと、  
所定の指示を受け付けたことに応じて、前記特定ステップで特定した遷移関係に基づき、前記特定のプログラムで表示される複数の画面のうち、前記複数の画面のそれぞれを基準として、遷移回数が所定数を超える遷移元と遷移先の画面の情報を含むことなく、遷移回数が前記所定数までの遷移元と遷移先の少なくとも一方の画面の情報を含む遷移図をそれぞれ生成するように制御する制御ステップと、  
前記制御ステップによる制御で生成された前記遷移図を表示するよう制御する表示制御ステップと

40

を情報処理装置に処理させる処理方法。

【請求項 16】

コンピュータを、請求項 1 乃至 14 のいずれか 1 項に記載された情報処理装置の各手段として機能させるためのプログラム。

【発明の詳細な説明】

【技術分野】

50

## 【 0 0 0 1 】

プログラム生成する情報処理装置とその処理方法及びプログラムに関する。

## 【 背景技術 】

## 【 0 0 0 2 】

近年、企業内では、クライアント端末に特別なアプリケーションをインストールすることなく業務が行えるように、業務用アプリケーションのWeb化が急速に進んでいる。そしてこのような業務アプリケーションの開発現場においては、開発スキル不足や要員不足を解決するために、Webアプリケーションをプログラミングレスで容易に作成できる開発ツールが用いられている。このような開発ツールを用いると、プログラミング言語の知識を有していなくても、業務・設計ノウハウを活用して基本設計情報を定義するだけで、Webアプリケーションを自動生成することができる。

10

## 【 0 0 0 3 】

ところで、このようなWebアプリケーションは複数の画面を用意し、画面から別の画面へと遷移していくように設計されるが、全体の業務の流れを把握するために画面遷移図を作成することがある。

## 【 0 0 0 4 】

このような画面遷移図を手作業で作成することは煩雑であるため、自動で画面遷移図を作成することのできる仕組みが求められている。

## 【 0 0 0 5 】

特に、Webアプリケーションの開発時には画面遷移の様子が開発途中で変更になることが多く、開発中においても画面遷移がどのような状態となっているのかを視覚的に認識することができる画面遷移図を生成する技術が、例えば特許文献1に開示されている。

20

## 【 先行技術文献 】

## 【 特許文献 】

## 【 0 0 0 6 】

【 文献 】 特開 2 0 1 6 - 1 7 3 7 2 1 号 公 報

## 【 発明の概要 】

## 【 発明が解決しようとする課題 】

## 【 0 0 0 7 】

ところで、Webアプリケーションでは多くの画面を有しているため、画面遷移図を表示すると複雑な図となってしまう見づらい環境となってしまう。

30

## 【 0 0 0 8 】

また、Webアプリケーションを実行した場合、利用ユーザは画面を介して操作しているため、その画面を基準とした画面遷移図が望ましいが、特許文献1の技術では、所定の遷移数以上の画面を中心に画面遷移図を表示するため、Webアプリケーションの実行時の環境には適さないという課題があった。

## 【 0 0 0 9 】

そこで、本発明の目的は、より適した画面遷移図を生成する仕組みを提供することである。

## 【 課題を解決するための手段 】

40

## 【 0 0 1 0 】

特定のプログラムで表示される画面の遷移関係を特定する特定手段と、所定の指示を受け付けたことに応じて、前記特定手段で特定した遷移関係に基づき、前記特定のプログラムで表示される複数の画面のうち、前記複数の画面のそれぞれを基準として、遷移回数が所定数を超える遷移元と遷移先の画面の情報を含むことなく、遷移回数が前記所定数までの遷移元と遷移先の少なくとも一方の画面の情報を含む遷移図をそれぞれ生成するように制御する制御手段と、前記制御手段による制御で生成された前記遷移図を表示するように制御する表示制御手段とを有することを特徴とする。

## 【 発明の効果 】

## 【 0 0 1 1 】

50

本発明によれば、より適した画面遷移図を生成することができる。

【図面の簡単な説明】

【0012】

【図1】本実施形態におけるシステム構成を示す図である。

【図2】本実施形態における各装置に適用可能なハードウェア構成を示す図である。

【図3】プログラム開発装置101における処理概要を示す図である。

【図4】プログラム開発装置101の処理後の処理概要を示す図である。

【図5】Webアプリケーションおよび画面仕様ドキュメント生成処理全体の一例を示すフローチャートである。

【図6】定義の構造の一例を示す図である。

10

【図7】画面仕様ドキュメントの生成処理の一例を示すフローチャートである。

【図8】検索情報作成処理の一例を示すフローチャートである。

【図9】画面一覧情報作成処理の一例を示すフローチャートである。

【図10】画面仕様情報作成処理の一例を示すフローチャートである。

【図11】画面項目仕様情報作成処理の一例を示すフローチャートである。

【図12】画面遷移図作成処理の一例を示すフローチャートである。

【図13】アプリケーション設定の一例を示す図である。

【図14】入出力定義の一例を示す図である。

【図15】入出力レイアウト定義画面の一例を示す図である。

【図16】階層レベルの設定情報の一例を示す図である。

20

【図17】画面遷移図と遷移図のデータの一例を示す図である。

【図18】アイコンが操作された際の画面遷移図の一例を示す図である。

【図19】画面仕様ドキュメントの一例を示す図である。

【図20】画面仕様ドキュメントにおける表示制御処理の一例を示すフローチャートである。

【図21】本実施形態における、特徴を示す機能構成の一例を示す図である。

【発明を実施するための形態】

【0013】

以下、本発明の実施の形態を、図面を参照して詳細に説明する。

【0014】

30

図1は、本発明に係わるプログラム開発装置（開発者がWebアプリケーション生成のために使用する情報処理装置）、プログラム開発サーバ、データベースサーバ、アプリケーションクライアント（クライアント装置）、アプリケーションサーバの構成の一例を示すシステム構成図である（情報処理システム）。

【0015】

プログラム開発装置101は、開発者の操作に従って画面レイアウトおよびデータベース検索指示などを定義する。プログラム開発装置101単体では、開発者の入力受付を行い、後述するプログラム開発サーバ102に実際のプログラム生成処理、アプリケーション生成処理をさせてもよいし、プログラム開発装置101単体でプログラム生成、アプリケーション生成まで処理してもよい。

40

【0016】

なお、この実施形態においては、プログラム開発装置101で生成するアプリケーションはWebアプリケーションとしたが、これに限定するものではなく、携帯電話・スマートフォン・タブレットなどの情報処理装置で動作するアプリケーションや組み込みソフトウェアなど、Web技術による通信を利用したアプリケーションでなくてもよい。

【0017】

プログラム開発サーバ102a～102b（情報処理装置）は、プログラム開発装置101により入力された開発者の指示に従って、プログラムを開発する。プログラム開発サーバ102aはLANなどのネットワーク106内に配置されてもよいし、プログラム開発サーバ102bはインターネット上やクラウド上に配置されてもよい。

50

## 【0018】

データベースサーバ103a~103b(情報処理装置)は、開発されたアプリケーションが使用するデータベースであり、また本発明では開発時にも動作確認などのために利用してもよい。例えば、開発者が利用するためにデータベースサーバ103は、プログラム開発装置101と同一の装置で構成されていてもよいし、LANなどのネットワーク106内に配置されてもよい(データベースサーバ103a)。またインターネット上やクラウド上に配置されてもよい(データベースサーバ103b)。また、プログラム開発装置101が、プログラム開発サーバ102と協調する場合には、プログラム開発サーバ102とデータベースサーバ103が同一の装置内に構成されていてもよい。

## 【0019】

アプリケーションサーバ105a~105b(情報処理装置)は、プログラム開発装置101で開発されたアプリケーションを実行する。LANなどのネットワーク106内に配置されてもよい(アプリケーションサーバ105a)し、またインターネット上やクラウド上に配置されてもよい(アプリケーションサーバ105b)。また、ネットワーク106、インターネット、クラウド上のデータベースサーバ103と接続して動作する可能である。

## 【0020】

アプリケーションクライアント104a~104b(情報処理装置)は、アプリケーションサーバ105と協調してプログラム開発装置101で開発したアプリケーションプログラムを動作させる、ユーザの入力端末である。LANなどのネットワーク106内に配置されてもよい(アプリケーションクライアント104a)し、またインターネット上やクラウド上に配置されてもよい(アプリケーションクライアント104b)。携帯端末などの情報処理装置であってもよい。

## 【0021】

図2は、本発明に係わるプログラム開発装置101、プログラム開発サーバ102、データベースサーバ103、アプリケーションクライアント104、アプリケーションサーバ105として適用可能な各ハードウェア構成の一例を示すブロック図である。

## 【0022】

図2において、CPU201は、システムバス204に接続される各デバイスを統括的に制御する。

## 【0023】

また、ROM203あるいは外部メモリ211には、CPU201の制御プログラムであるオペレーティングシステム(OS)や、各サーバ、クライアント、装置など情報処理装置の後述する各種機能を実現するためのプログラムが記憶されている。

## 【0024】

RAM202は、CPU201の主メモリ、ワークエリア、一時待避領域等として機能する。

## 【0025】

入力コントローラ205は、入力部209からの入力を制御する。この入力部209としては、情報処理装置では、キーボード、マウス等のポインティングデバイスが挙げられる。

## 【0026】

出力コントローラ206は、出力部210の表示を制御する。この出力部210としては、例えば、CRTや液晶ディスプレイ等が挙げられる。

## 【0027】

外部メモリコントローラ207は、ブートプログラム、各種のアプリケーション、フォントデータ、ユーザーファイル、編集ファイル、プリンタドライバ等を記憶する外部メモリ211へのアクセスを制御する。加えて、各サーバ、クライアント、装置等の各種機能を実現するための各種テーブル、パラメータが記憶されている。この外部メモリ211としては、ハードディスク(HD)やフレキシブルディスク(FD)、PCMCIAカード

10

20

30

40

50

スロットにアダプタを介して接続されるコンパクトフラッシュ（登録商標）、スマートメディア等が挙げられる。

【0028】

通信I/Fコントローラ208は、ネットワークを介して外部機器との通信制御処理を実行する。

【0029】

本発明を実現するためのプログラム212は外部メモリ211に記録されており、必要に応じてRAM202にロードされることによりCPU201によって実行されるものである。

【0030】

次に本実施形態における基本の処理について、図3、図4の構成図を用いて説明する。なお、本実施形態における特徴となる処理の詳細については、図5、図7～図12、図20のフローチャートを用いて説明する（特に図12）。

【0031】

図3は、プログラム開発装置101の処理概要を示す構成図である。プログラム開発装置101は、Webアプリケーションを開発する開発者が定義した定義ファイルをもとにWebアプリケーションを自動生成することを目的とした装置である。プログラム開発装置101は、リポジトリ定義部401、Webアプリケーション生成部416、画面仕様ドキュメント生成部420により構成される。また、前述に加え、階層レベル設定415、画面遷移図生成部439を本実施形態における特徴として備えている。

【0032】

リポジトリ定義部401は、アプリケーション定義402、入出力定義403、データモデル定義404、ビジネスプロセス定義405、データベース定義406、ドキュメント用定義407、画面HTMLデータ408、ドキュメントフィルタ定義409、表紙ファイル410、アプリケーションリソースファイル411、ドキュメントリソースファイル412、ヘッダ/フッタファイル413、関連文書414、階層レベル設定415を備える。これらのファイルは、Webアプリケーション開発ツールを介して開発者によって入力され、作成される。すなわち、リポジトリ定義部401は、プログラムへ引数として入力される項目を定義する入力定義情報と、プログラムから処理結果として出力する項目を定義する出力定義情報とを入出力定義情報として管理している。

【0033】

次に、Webアプリケーション生成部416について説明する。Webアプリケーション生成部416で生成された情報は、外部メモリ211に記憶される。

【0034】

Webアプリケーション生成部416はリポジトリ定義解析部417によってリポジトリ定義部401を解析し、Webアプリケーションコード生成部418を介し、ソースコードコンパイル部419にてコンパイル済Javaコード444とHTML/JSP/Javascript445からなるWebアプリケーション443を生成する（javaは登録商標）。

【0035】

リポジトリ定義解析部417は、リポジトリ定義部401からデータベース定義406、アプリケーション定義402、データモデル定義404、入出力定義403、ビジネスプロセス定義405を読み込み解析するものとする。

【0036】

すなわち、Webアプリケーション生成部416は、Webアプリケーションに用いられるプログラムを生成するルールを記憶する手段の一例である。

【0037】

Webアプリケーション生成部416は、外部メモリ211に記憶されているコード生成ルールと、リポジトリ定義解析部417によって解析された内容とを用いて、Webアプリケーションコード生成部418でソースコード（JSP）を生成する。このソースコ

10

20

30

40

50

ードを取り込みソースコードコンパイル部 4 1 9 でコンパイルされたファイルを生成する。そしてソースコードコンパイル部 4 1 9 がコンパイルされたファイルをアプリケーションサーバへデプロイする。なお、ソースコード、コンパイルされたファイル、デプロイされたデータを総称してプログラムデータともいう。

【 0 0 3 8 】

より詳細には、生成されるソースコードは、大きくサーバ用とクライアント用に分かれている。サーバ用のソースコードをコンパイルしコンパイルされたファイルが生成され、クライアント用のソースコードはコンパイルされずファイルとしてプログラム開発装置 1 0 1 上に置く。このファイルを用いてアプリケーションサーバ 1 0 5 にデプロイする。

【 0 0 3 9 】

また、サーバ用は J a v a プログラム ( j a v a は登録商標 ) が生成され、生成された直後にコンパイルされて c l a s s ファイルになり、その c l a s s ファイルがデプロイされる。クライアント用は J S P / J a v a S c r i p t / C S S が生成され、それらはコンパイルされず、そのままデプロイされる。

【 0 0 4 0 】

デプロイされたモジュールが W e b アプリケーション 4 4 3 である。具体的には、デプロイすると「 . w a r 」の拡張子を有するファイルがアプリケーションサーバに配置される。すなわち、 W e b アプリケーション生成部 4 1 6 は、 W e b アプリケーションに用いられるプログラムを生成するルールを記憶している。例えば、予め決められたルールに従って、ソースコードが生成される。

【 0 0 4 1 】

次に、画面仕様ドキュメント生成部 4 2 0 について説明する。画面仕様ドキュメント生成部 4 2 0 で生成された情報は、外部メモリ 2 1 1 に記憶される。

【 0 0 4 2 】

画面仕様ドキュメント生成部 4 2 0 は、リポジトリ定義解析部 4 2 1 によってリポジトリ定義部 4 0 1 を解析する。画面仕様ドキュメント生成部 4 2 0 は、画面一覧生成部 4 2 2、画面仕様生成部 4 2 3、画面 H T M L 生成部 4 2 4、画面項目仕様生成部 4 2 5、検索機能生成部 4 2 6、関連文書生成部 4 2 8、表紙生成部 4 2 9、ヘッダ/フッタ生成部 4 3 0、画面遷移図生成部 4 3 9 を介して H T M L / C S S / J a v a S c r i p t 4 4 9 と画像ファイル 4 5 0 からなる画面仕様ドキュメント 4 4 8 を生成する ( j a v a は登録商標 )。

【 0 0 4 3 】

画面遷移図生成部 4 3 9 は、画面遷移図に必要なデータを抽出する画面遷移データ抽出部 4 4 0 と、画面ブロック生成部 4 4 1 と、画面遷移図レイアウト構成部 4 4 2 を有している。

【 0 0 4 4 】

その際、画面遷移図レイアウト構成部 4 4 2 は、出力フィルタ管理部 4 3 1 を介してドキュメントエリア設定管理部 4 3 3 のエリア出力設定 4 3 4 とエリアサイズ設定 4 3 5 によって画面一覧生成部 4 2 2、画面仕様生成部 4 2 3、画面 H T M L 生成部 4 2 4、画面項目仕様生成部 4 2 5、検索機能生成部 4 2 6 へレイアウトを反映し、さらにドキュメントテーマ管理部を介して、組み込みデザイン情報 4 3 7 やユーザ作成デザイン情報 4 3 8 を用いて、ドキュメント全体の装飾を画面仕様ドキュメント 4 4 8 へ反映する。

【 0 0 4 5 】

また、言語別ドキュメント置換処理部 4 3 2 は表紙生成部 4 2 9、ヘッダ/フッタ生成部 4 3 0 へ言語別のファイルを設定する。

【 0 0 4 6 】

すなわち、画面仕様ドキュメント生成部 4 2 0 は、入出力定義 4 0 3 を用いて、 W e b アプリケーションの定義 ( 画面 ) を説明するドキュメントを生成する。

【 0 0 4 7 】

図 4 は、アプリケーションサーバ 1 0 5 と W e b サーバ 1 0 7 とアプリケーションクラ

10

20

30

40

50

クライアント104の処理概要を示す構成図である。

【0048】

アプリケーションサーバ105は、Webアプリケーション生成部416で生成されたファイルをデプロイし、Webアプリケーション443を実行するサーバである。Webアプリケーション443は、コンパイル済Javaコード444と、HTML/JSP/JavaScript445のファイルを含むものである(javaは登録商標)。また、データベース446と連携している。

【0049】

Webサーバ107は、画面仕様ドキュメント生成部420で生成された画面仕様ドキュメント448を記憶し、アプリケーションクライアント104からの要求に応じて、画面仕様ドキュメントをアプリケーションクライアント104送信する。

10

【0050】

画面仕様ドキュメントのアプリケーションクライアント104の表示例が図19の1900である。

【0051】

なお、本実施形態では、アプリケーションサーバ105とWebサーバ107とを別筐体としたが、同一筐体で実行させる構成であってもよい。

【0052】

図5は、Webアプリケーションおよび画面仕様ドキュメント生成処理全体のフローチャートの一例を示す図である。なお、以下のフローチャートの各ステップは、各装置のCPU201が実行する。

20

【0053】

ステップS500において、プログラム開発装置101は、Webアプリケーションの自動生成のための定義の入力を受ける。具体的には、開発するWebアプリケーション(プロジェクト)として、図13のアプリケーション設定1300でアプリケーションのコードや名前、アプリケーションタイプなどを設定する。

【0054】

このWebアプリケーション(プロジェクト)に対して、データモデル定義、入出力定義、ビジネスプロセス定義を設定する。これらの定義は、Webアプリケーション自動生成する設定画面でユーザが任意に設定するものとするが、予め生成された定義を読み込む構成であってもよい。

30

【0055】

また、所属入出力一覧1301で、どの入出力定義を所属(利用)するかを設定する。

【0056】

入出力定義は、図14と図15で設定したデータを用いて定義されるものである。また、画面仕様ドキュメントで表示する画面遷移図階層レベルについても、事前に設定しておくものとし、図16の1600の設定情報を記憶する。本実施形態では、1601で示すように階層は1階層の設定となっている。

【0057】

ステップS501において、プログラム開発装置101は、ユーザによるWebアプリケーション生成指示を受け付けると、外部メモリ211に記憶されているリポジトリ定義部401のアプリケーション定義402をRAM202に読み込む。

40

【0058】

ステップS502において、プログラム開発装置101は、データモデル定義404を読み込む。データモデル定義404は、リポジトリ定義部401として登録されている。データモデル定義404は、データモデル設定画面(不図示)でデータベース上の項目をマッピングして登録されるものである。一般的には、1テーブル1データモデルが作成される。データモデル定義の作成はWebアプリケーションの開発において既知の技術であるため説明を省略する。

【0059】

50

ステップ S 5 0 3 において、プログラム開発装置 1 0 1 は、入出力定義 4 0 3 を読み込む。入出力定義 4 0 3 は、リポジトリ定義部 4 0 1 として登録されている。

【 0 0 6 0 】

入出力定義 4 0 3 の生成について説明する。上述したように入出力定義 4 0 3 の設定自体はステップ S 5 0 0 など事前に行われる。まず、図 1 5 の入出力レイアウト定義画面 1 5 0 0 上で、画面エディタ上にオブジェクト（フィールドやアクションボタン）をドラッグアンドドロップ操作等により配置する。

【 0 0 6 1 】

また、アクションボタンに対して、図 1 5 の 1 5 0 1 で条件を設定する。1 5 0 2 でアクションボタンを押下した際の遷移先の画面（次入出力の定義）を設定する。

10

【 0 0 6 2 】

図 1 5 に対応する入出力定義の例が、図 1 4 の 1 4 0 3 である。図 1 4 の 1 4 0 3 では、アクションボタンに対応する定義のほか、入出力レイアウト定義画面 1 5 0 0 上で設定したテキストフィールドの定義が別のレコードに表示されている。

【 0 0 6 3 】

画面ごとの入出力定義が 1 4 0 0 ~ 1 4 0 3 を示している。なお、次入出力への設定は画面の定義以外に、インポートやエクスポートの処理にかかる定義を設定することも可能である。

【 0 0 6 4 】

1 4 0 4 で、入出力定義に対して、「（アット）N O D O C」の識別情報を設定することで、個別に画面仕様ドキュメントが生成されないように制御することが可能となっている。画面仕様ドキュメントは生成されないが、画面遷移図のオブジェクトとしては表示されるものとする。「（アット）N O D O C」については、以下「N O D O C」として説明する。

20

【 0 0 6 5 】

本実施形態における設定された定義の構造を示す図が図 6 の 6 0 0 であり、1 4 0 0 ~ 1 4 0 3 に対応する構造が、6 0 1 である。

【 0 0 6 6 】

アクションボタンに対応する定義については、項目タイプ、項目コード、名前、次入出力などの値を有している。次入出力については、アクションにより次に移る画面（定義）が設定されている。本実施形態では、特に項目タイプによって画面遷移図が色分けされ、次入出力によって遷移先が決定され画面遷移図が生成される。

30

【 0 0 6 7 】

ステップ S 5 0 4 において、プログラム開発装置 1 0 1 は、ビジネスプロセス定義 4 0 5 を読み込む。ビジネスプロセス定義 4 0 5 は、リポジトリ定義部 4 0 1 として登録されている。ビジネスプロセス定義 4 0 5 についてステップ S 5 0 0 で事前設定されているものである。また、ビジネスプロセス定義 4 0 5 は、受け付けたデータの処理の流れを定義したものである。例えば、入力したデータの重複をチェックして、登録するといったロジックを定義したものである。ビジネスプロセス定義についても、Web アプリケーションの開発において既知の技術であるため説明を省略する。

40

【 0 0 6 8 】

ステップ S 5 0 5 において、プログラム開発装置 1 0 1 は、外部メモリ 2 1 1 に記憶されているリポジトリ定義部 4 0 1 のデータベース定義 4 0 6 を R A M 2 0 2 に読み込む。データベース定義 4 0 6 は予め生成されているものとする。

【 0 0 6 9 】

ステップ S 5 0 6 で、プログラム開発装置 1 0 1 の Web アプリケーション生成部 4 1 6 が、読み込みしたリポジトリ定義部 4 0 1 の各定義・各ファイルから情報を取得し、Web アプリケーションのソースコードを生成する（例えば、Web アプリケーション 4 4 3）。具体的には、「入出力定義」「データモデル定義」「ビジネスプロセス定義」に基づき生成する。なお、生成されたソースコードにはプログラミング言語が記載されたファ

50

イルHTML、JSP、JavaScript（登録商標）等のWebアプリケーションの提供に利用されるファイルも含まれる。

【0070】

ステップS507で、プログラム開発装置101の画面仕様ドキュメント生成部420が、画面仕様ドキュメント448を生成する。リポジトリ定義解析部421は、読み込んだ定義を解析したうえでメモリに記憶しておき、各生成部から適宜参照される。また、レイアウト構成部427は、各生成部の作成したドキュメントの各構成要素をレイアウトして、画面仕様ドキュメントを構成する。

【0071】

画面仕様ドキュメントの生成処理については、図7で詳述するが、生成した画面仕様ドキュメントのイメージ図は図19の1900である。生成された画面仕様ドキュメントは、対応する入出力定義のコード（例えば、PRODUCT\_REGISTER\_IOなど）に対応付けて保存される。

10

【0072】

ここで、図19の画面仕様ドキュメントの説明を行う。

【0073】

1901は、検索機能を示している。画面仕様ドキュメントから、指定された文字列を含む入出力を検索して表示することが可能である。

【0074】

1902は、画面仕様ドキュメントを表示可能な一覧である。1301で指定した入出力の一覧に対応して表示する。ユーザの操作（クリック等）で、その入出力の画面仕様ドキュメントを右ペインに表示する。

20

【0075】

1903は、画面仕様エリアである。入出力定義で設定した属性を表示する。

【0076】

1904は、画面HTMLエリアである。入出力定義の画面イメージをHTMLで表示する。

【0077】

1905は、画面項目仕様エリアである。画面に対応する入出力項目の属性を一覧表示する。1904の「1」～「6」に対応して属性が表示されている。

30

【0078】

1906は、画面遷移図エリアである。ドキュメント表示している入出力の遷移元、遷移先を画面遷移図で表示する。この画面遷移図の生成は図12で詳述する。

【0079】

1907は、関連文書エリアである。関連文書へのリンクを表示する。

【0080】

図7は、画面仕様ドキュメント生成のフローチャートの一例を示す図である。なお、以下のフローチャートの各ステップは、装置のCPU201が実行する。

【0081】

ステップS701は、出力フィルタ管理部431が、ドキュメントフィルタ定義409を読み込む。出力フィルタ管理部431はドキュメントフィルタ情報をメモリに記憶し、その情報は各種生成部から適宜参照される。

40

【0082】

ドキュメントフィルタとは、例えば、図19の1900に表示する属性（項目）や、入出力タイプ（IO、MENU、DIALOG）を決定するものであり、開発者が任意に設定できるものである。これによって、図19の1900に表示される情報がフィルタされる。

【0083】

ステップS702は、ヘッダ/フッタ生成部430が、ヘッダ/フッタファイル413の有無を確認する。ヘッダ/フッタファイル413があった場合、ステップS703で、

50

ヘッダ/フッタ生成部 430 が、ヘッダ/フッタを作成する。ここでいうヘッダ/フッタとは、図 19 の 1900 で不図示となっており、例えば、1900 に加えてヘッダ部とフッタ部を JSP で記述できるものである。ヘッダ部は例えば企業ロゴなどを表示するような部分である。また、フッタ部は例えば、コピーライトを表示するような部分である。

【0084】

ステップ S704 は、検索機能生成部 426 が、検索情報の作成を行う。詳細については、図 8 のフローチャートで説明する。

【0085】

ステップ S705 は、画面一覧生成部 422 が、画面一覧の作成を行う。詳細については、図 9 のフローチャートで説明する。

10

【0086】

入出力の定義分、ステップ S706 から S719 までの処理を繰り返す。なお、本実施形態では、図 14 に示すように入出力定義が 3 つ（メインメニュー、商品一覧、商品登録）があるため、3 回繰り返すものとする。

【0087】

ステップ S707 は、画面仕様ドキュメント生成部 420 が、入出力定義に「NODOC」指定があるかどうかを確認する。指定があった場合、その入出力定義に対する画面仕様ドキュメントは出力せず、次の入出力定義の処理に移る。

【0088】

「NODOC」の指定は、1404 で指定されているもので、各入出力定義情報に対してメモリで管理している。1404 は、入出力定義に「NODOC」を指定した画面イメージ図である。

20

【0089】

ステップ S708 は、画面仕様ドキュメント生成部 420 が、出力フィルタ管理部に問い合わせ、入出力定義の入出力タイプ（MENU や IO など）が出力対象と設定されているかどうかを確認する。出力対象とするかは事前に開発者が設定しているものとする。

【0090】

出力対象とされていなかった場合、その入出力定義の画面仕様ドキュメントは出力対象とせず、次の入出力の処理に移る。

【0091】

ステップ S709 は、画面一覧生成部 422 が、ステップ S705 で作成した情報を元に、画面一覧を出力する。具体的には、1902 に表示される HTML 情報を出力するものである。

30

【0092】

ステップ S710 は、画面仕様生成部 423 が、画面仕様を作成・出力する。画面仕様作成・出力処理については、図 10 のフローチャートで説明する。

【0093】

ステップ S711 は、画面 HTML 生成部 424 が、画面 HTML を作成・出力する。画面 HTML の各表示部品には、画面 HTML データ 408 のデータを組み込む。この画面 HTML は、入出力レイアウト定義画面 1500 でレイアウトした画面に相当するデータであり、各項目に対して番号を付与したものである。

40

【0094】

ステップ S712 は、画面項目仕様生成部 425 が、画面項目仕様を作成・出力する。画面項目仕様作成・出力処理については、図 11 のフローチャートで説明する。

【0095】

ステップ S713 は、関連文書生成部 428 が、関連文書 414 が設定されているかどうかを確認する。設定されている場合には、ステップ S714 へ処理を移し、設定されていなかった場合は、ステップ S715 へ処理を移す。

【0096】

ステップ S714 は、関連文書生成部 428 が、関連文書のリンク情報を作成して H T

50

MLファイルへ出力する。

【0097】

ステップS715は、画面遷移図生成部439が、画面遷移図を作成・出力する。画面遷移図作成・出力処理については、図12のフローチャートで説明する。

【0098】

ステップS716は、ヘッダ/フッタ生成部430が、ステップS702にてヘッダ/フッタが作成されているか否かを判定する。作成されていた場合、ステップS717へ処理を移し、作成されていない場合にはステップS718へ処理を移す。

【0099】

ステップS717は、ヘッダ/フッタ生成部430が、作成したヘッダ/フッタをHTMLファイルへ出力する。

10

【0100】

ステップS718は、検索機能生成部426が、ステップS704で作成した情報を元に、HTMLファイルへ出力する。

【0101】

ステップS720は、表紙生成部429が、表紙ファイル410が設定されているかを判定する。表紙ファイル410が設定されている場合には、ステップS721へ処理を移す。表紙ファイル410が設定されていない場合には、処理を終了する。

【0102】

表紙ファイル410は、1908のエリアに表示するトップ画面を示すファイルである。このトップ画面がない場合には、画面一覧における最初の画面(定義)に対応する画面仕様ドキュメントが表示される。例えば、メインメニュー。

20

【0103】

図8は、検索情報作成のフローチャートの一例を示す図である。なお、以下のフローチャートの各ステップは、装置のCPU201が実行する。

【0104】

入出力定義の数分、検索機能生成部426が、ステップS801からステップS805までの処理を繰り返す。

【0105】

ステップS802は、入出力定義に「NODOC」指定があるか否かを判定する。「NODOC」指定がある場合には、ステップS804へ処理を移し、「NODOC」指定がない場合には、ステップS803へ処理を移す。「NODOC」指定がある場合には、その入出力の検索情報は作成せず、次の入出力定義の処理に移る。

30

【0106】

ステップS803で、出力フィルタ管理部431に問い合わせ、入出力定義の入出力タイプ(1400参照)が出力対象と設定されているか否かを判定する。出力対象とされていない場合、その入出力の検索情報は作成せず、次の入出力の処理に移る。

【0107】

ステップS804で、検索情報を作成する。検索情報とは、画面仕様ドキュメント内を検索するための文字列を含む情報である。例えば、入出力定義の設定値であって、1901で入力されたキーワードを検索するためのテキストベースのデータである。作成された検索情報は外部メモリに記憶する。

40

【0108】

図9は、画面一覧情報作成のフローチャートの一例を示す図である。なお、以下のフローチャートの各ステップは、装置のCPU201が実行する。

【0109】

入出力定義の数分、画面一覧生成部422が、ステップS901からステップS905までの処理を繰り返す。

【0110】

ステップS902は、入出力定義に「NODOC」指定があるか否かを判定する。「N

50

「NODOC」指定がある場合には、ステップS905へ処理を移し、「NODOC」指定がない場合には、ステップS903へ処理を移す。「NODOC」指定がある場合には、その入出力定義に関する画面一覧情報(1901に相当)は作成せず、次の入出力定義の処理に移る。

【0111】

ステップS903は、出力フィルタ管理部431に問い合わせ、入出力定義の入出力タイプ(1400参照)が出力対象と設定されているか否かを判定する。出力対象とされていない場合、その入出力の検索情報は作成せず、次の入出力の処理に移る。

【0112】

ステップS904で、入出力定義に対応する画面一覧情報を作成する。具体的には、入出力定義のコードと名前(1400参照)をもとに、部品を生成する。この部品には、生成される画面仕様ドキュメントのリンクを付与され、部品を押下すると対象の画面仕様ドキュメントを取得して表示することが可能となる。

10

【0113】

図10は、画面仕様情報作成のフローチャートの一例を示す図である。なお、以下のフローチャートの各ステップは、装置のCPU201が実行する。

【0114】

入出力定義の属性の数分、画面仕様生成部423が、ステップS1001からステップS1004までの処理を繰り返す。

【0115】

ステップS1002は、出力フィルタ管理部431に問い合わせ、入出力定義の属性が出力対象と設定されているか否かを判定する。出力対象とされていなかった場合、その入出力属性情報は作成せず、次の入出力属性の処理に移る。出力対象とされていた場合、ステップS1003へ処理を移す。

20

【0116】

ステップS1003は、入出力属性情報(1903に相当)を作成する。

【0117】

ステップS1005は、画面仕様生成部423が、入出力属性情報を元に、画面仕様をHTMLファイルへ出力する。

【0118】

図11は、画面項目仕様情報作成のフローチャートの一例を示す図である。なお、以下のフローチャートの各ステップは、装置のCPU201が実行する。

30

【0119】

入出力定義の項目の数分、画面項目仕様生成部425が、ステップS1101からステップS1110までの処理を繰り返す。

【0120】

ステップS1102は、入出力項目に「NODOC」指定があるか否かを判定する。指定があった場合、その入出力項目の画面項目仕様情報は作成せず、次の入出力項目の処理に移る。指定がなかった場合ステップS1103へ処理を移す。

【0121】

ステップS1103は、入出力項目が非表示項目か否かを判定する。非表示項目であった場合にはステップS1104へ処理を移す。

40

【0122】

ステップS1104は、出力フィルタ管理部431に問い合わせ、非表示項目が出力対象と設定されているか否かを判定する。出力対象とされていなかった場合、その入出力項目の画面項目仕様情報は作成せず、次の入出力項目の処理に移す。出力対象とされている場合、ステップS1105へ処理を移す。

【0123】

ステップS1105は、出力フィルタ管理部431に問い合わせ、入出力項目の入出力項目タイプが出力対象と設定されているか否かを判定する。出力対象とされていなかった

50

場合、その入出力項目の画面項目仕様情報は作成せず、次の入出力項目の処理に移る。出力対象とされている場合、ステップ S 1 1 0 6 へ処理を移す。

【 0 1 2 4 】

入出力項目の入出力項目属性の数分、ステップ S 1 1 0 6 からステップ S 1 1 0 9 までの処理を繰り返す。

【 0 1 2 5 】

ステップ S 1 1 0 7 は、出力フィルタ管理部 4 3 1 に問い合わせ、入出力項目の属性が出力対象と設定されているか否かを判定する。出力対象とされていなかった場合、その入出力項目属性情報は作成せず、次の入出力属性の処理に移る。

出力対象とされていた場合、ステップ S 1 1 0 8 へ処理を移す。

10

【 0 1 2 6 】

ステップ S 1 1 0 8 は、画面項目仕様情報を作成する。

【 0 1 2 7 】

ステップ S 1 1 1 1 は、画面項目仕様生成部 4 2 5 が、画面項目仕様情報を元に、画面項目仕様を HTML ファイルへ出力する ( 1 9 0 5 に相当 ) 。

【 0 1 2 8 】

図 1 2 は、画面遷移図作成のフローチャートの一例を示す図である。なお、以下のフローチャートの各ステップは、装置の CPU 2 0 1 が実行する。

【 0 1 2 9 】

入出力定義の定義数分、画面遷移データ抽出部 4 4 0 が、ステップ S 1 2 0 1 からステップ S 1 2 0 7 までの処理を繰り返す。本実施形態では、入出力定義が 3 つあるため、3 回繰り返す。

20

【 0 1 3 0 】

ステップ S 1 2 0 2 は、入出力定義から入出力の項目コードと名前 ( 図 1 4 参照 ) と画面データ ( 図 1 5 の 1 5 0 3 に相当 ) を抽出する。

【 0 1 3 1 】

ステップ S 1 2 0 3 は、入出力定義からアクション定義を全て抽出しメモリに保存する。アクション定義とは、項目タイプで「 A アクション」となっている定義である。なお、ユーザの操作等によりイベントが発生したときに自動でアクション ( 検索、登録、削除、キャンセル ) が実行されるものが、アクション定義として設定がされている。

30

【 0 1 3 2 】

ステップ S 1 2 0 4 からステップ S 1 2 0 6 までの処理をアクション定義の数分繰り返す。

【 0 1 3 3 】

ステップ S 1 2 0 5 は、アクション定義から遷移データを抽出し、メモリに保存する。遷移データとは、図 1 4 にある次入出力の設定値である。1 4 1 3 を参照すると、商品登録画面で登録ボタンが押下されると、商品一覧画面に遷移することを特定できる情報が保存される。また、1 4 1 1 と 1 4 1 2 を参照すると、商品登録画面は、メインメニューと商品一覧画面から遷移することが特定できる情報が保存される。すなわち、入出力定義に含まれるアクション情報に従って、遷移先の入出力定義を特定する特定処理の一例を示すものである。

40

【 0 1 3 4 】

ステップ S 1 2 0 8 は、階層レベル設定ファイル 1 6 0 0 を読み込む。本実施形態では、1 6 0 1 に示す通り階層レベルは「 1 」となっている。

【 0 1 3 5 】

ステップ S 1 2 0 9 は、ステップ S 1 2 0 2 とステップ S 1 2 0 5 で保存したメモリから今回の処理対象の画面 ( 例えば、商品登録画面の定義 ) のデータとステップ S 1 2 0 8 で 1 6 0 0 のような設定ファイルから読み込んだ階層レベル設定分の遷移データを読み出す。

【 0 1 3 6 】

50

画面遷移データ抽出部 4 4 0 で抽出した対象画面と遷移画面分のデータを、画面ブロック生成部 4 4 1 が、ステップ S 1 2 1 0 からステップ S 1 2 1 4 までの処理を繰り返す。この処理は遷移図で用いる遷移元と遷移先となるデータ部品を生成する処理である。

【 0 1 3 7 】

ステップ S 1 2 1 1 は、画面遷移する入出力をデータ部品（ブロック）に変換する。データ部品に変換する際には、1 4 0 0 のコードと名前の値を含めたデータ部品（矩形オブジェクト）を生成する。なお、データ部品を変換する（生成する）場合には、1 4 0 0 の入出力タイプが識別できるように色分けする。色情報はタイプによってあらかじめ定義されているものとするが、開発者が任意に設定してもよい。

10

【 0 1 3 8 】

また、生成している画面仕様ドキュメントに対応する入出力定義が基準となり、この基準となる入出力定義と、遷移先又は遷移元の入出力定義に対応するオブジェクトを生成すると処理と言い換えることも可能である。

【 0 1 3 9 】

ステップ S 1 2 1 2 は、入出力定義に「NODOC」指定があるか否かを判定する。指定がある場合には、その入出力定義の画面仕様ドキュメントへのリンクの作成は行わず、次の入出力の処理に移る。指定がない場合には、ステップ S 1 2 1 3 に処理を移す。

【 0 1 4 0 】

ステップ S 1 2 1 3 は、生成したデータ部品に、入出力定義に対応して生成された画面仕様ドキュメントへのリンクを作成する。画面仕様ドキュメントのファイル名などあらかじめルールに従って生成されるため、このタイミングでリンクを生成することが可能となっている。なお、リンクの作成は、図 7 の処理が終了したタイミング（画面仕様ドキュメントが完全に生成されたタイミング）で行ってもよい。

20

【 0 1 4 1 】

「NODOC」指定がある場合には、画面仕様ドキュメントは生成されていないが、遷移元や遷移先としてはユーザに認識させることが望ましいため画面遷移図上では確認できるようにしている。

【 0 1 4 2 】

ステップ S 1 2 2 3 は、アクション情報を生成する。具体的には、遷移元の画面（入出力定義）で用いられているアクション情報（1 5 0 1）を収集する。このアクション情報は、1 7 0 3 のアイコンの押下によって表示される情報となる。画面遷移図における遷移元又は遷移先にかかるアクション情報を取得する取得処理の一例を示すものである。

30

【 0 1 4 3 】

ここで、アクション情報で同じ画面に遷移する場合（同じ次入出力がある場合）、画面遷移図でアクション情報をまとめて表示できるように管理する。

【 0 1 4 4 】

ステップ S 1 2 1 5 は、画面遷移図レイアウト構成部 4 4 2 がデータ部品を描画する位置を設定する。具体的には、商品登録画面（商品登録の入出力定義）に対する画面仕様ドキュメントを生成している場合には、商品登録画面のデータ部品を中央になるように位置を決定する（HTML 5 の Canvas 上の位置）。図 1 7 を参照すると、1 7 0 5 の位置である。Canvas を示す情報が、1 7 0 1 である。すなわち、基準となる入出力定義に対応するオブジェクトをユーザが認識しやすい形態で配置する処理の一例を示すものである。

40

【 0 1 4 5 】

ステップ S 1 2 1 6 は、データ部品を遷移元と遷移先のグループに分類する。図 1 7 を参照すると、1 7 0 2 のように、対象画面とそれ以外の画面データを遷移元、遷移先グループに分類した画面遷移情報を作成する。1 7 0 6 が遷移元の情報で、1 7 0 7 が遷移先の情報となっている。

【 0 1 4 6 】

50

ステップ S 1 2 1 6 で分類したグループ分、ステップ S 1 2 1 7 からステップ S 1 2 2 1 までの処理を繰り返す。

【 0 1 4 7 】

ステップ S 1 2 1 8 は、各グループのデータ部品を描画する位置（HTML5のCanvas上の位置）を設定する。描画する位置は、ステップ S 1 2 1 5 の決定した位置の左の位置に遷移元のデータ部品の位置を、右の位置に遷移先のデータ部品の位置を設定する。同じグループに複数のデータ部品がある場合には、縦に位置するように設定する。描画する位置は所定のルールに従って決定される。

【 0 1 4 8 】

ステップ S 1 2 1 9 は、各データ部品間を結ぶ遷移を表す遷移矢印を描画する位置を設定する。

10

【 0 1 4 9 】

ステップ S 1 2 2 0 は、アクション情報を表示するためのアイコンを描画する位置を設定する。すなわち、アクション情報を表示させるオブジェクトを、画面遷移図における遷移にかかる位置に配置する処理の一例を示すものである。

【 0 1 5 0 】

すなわち、ステップ S 1 2 1 0 ~ ステップ S 1 2 2 1 の処理は、特定処理により特定された遷移先の入出力定義に従って、基準となる入出力定義における遷移構成を決定する決定処理の一例を示すものである。

【 0 1 5 1 】

20

ステップ S 1 2 2 2 は、ステップ S 1 2 1 5 からステップ S 1 2 2 1 で設定した画面遷移図描画のための設定を基に、キャンバス上に画面遷移図を描画する。すなわち、決定処理により決定された遷移構成に従って、生成されるプログラムで用いられる指定された画面それぞれに対応する画面遷移図を生成する画面遷移図生成処理の一例を示すものである。また、アクション情報を任意に表示可能に、画面遷移図を生成する処理である。

【 0 1 5 2 】

ここで画面遷移図を描画した際のイメージについて説明する。図 1 7 は画面遷移図と遷移図のデータを示す図である。

【 0 1 5 3 】

1 7 0 0 は、画面仕様ドキュメント出力の対象となる入出力定義（PRODUCT\_REGISTER\_IO）の画面遷移図のイメージ図である。1 7 0 1 は、画面遷移図部分のHTMLである。1 7 0 2 は、画面遷移情報である。

30

1 7 0 3 は、画面遷移図に表示しているアイコンをクリックしたときに表示するアイコンである。1 7 0 4 は、1 4 1 3 の次入出力の設定値から決定された遷移先のデータ部品である。1 7 0 5 は、画面仕様ドキュメント出力の対象となる入出力定義（PRODUCT\_REGISTER\_IO）のデータ部品を示し、キャンバス上の中央に配置されている。

【 0 1 5 4 】

また、図 1 7 の 1 7 0 3 のアクション情報のアイコンが操作された際の画面遷移図を図 1 8 に示す。図 1 8 は、アイコンが操作された際の画面遷移図のイメージ図である。1 8 0 0 はアクション情報表示例を示す図であり、1 8 0 1 がアイコンを押下した際に表示される商品一覧のアクション情報例である。また 1 8 0 1 は、1 4 1 2 のアクションに関する情報となっている。

40

【 0 1 5 5 】

この 1 8 0 1 に表示するアクションに関する情報は、ステップ S 1 2 2 2 で遷移図を描画する際に、メモリから取得する。すなわち、画面遷移図における遷移元又は遷移先にかかるアクション情報を取得する取得処理の一例である。

【 0 1 5 6 】

また、1 8 0 1 に表示するアクションに関する情報は、同一の遷移先に対するアクション情報が複数ある場合には、当該複数のアクション情報をまとめて表示可能に、画面遷移図に表示する。

50

## 【0157】

このように、指定された画面（入出力定義）を基準として、その画面それぞれに適した画面遷移図を容易に生成することができる。画面仕様ドキュメント単位で適した画面遷移図が生成することができるため、ユーザが確認している画面を基準に、どのようなアクションによってどのような画面遷移がされるのかを容易に把握させることが可能となる。

## 【0158】

図20は、画面仕様ドキュメントにおける表示制御のフローチャートの一例を示す図である。なお、以下のフローチャートの各ステップは、装置のCPU201が実行する。

## 【0159】

アプリケーションクライアント104から、Webサーバ107へアクセスして、画面仕様ドキュメント（HTML）を取得する。取得した画面仕様ドキュメントを表示部に表示する。上述したように、画面仕様ドキュメントの表示例が図19である。

10

## 【0160】

ステップS2001は、画面一覧1902を介してユーザからの操作を受け付ける。具体的には、表示したい入出力定義の画面仕様ドキュメントのクリックを受け付ける。

## 【0161】

ステップS2002は、選択された入出力定義の画面仕様ドキュメントを、画面一覧1902のリンクに従って、Webサーバ107から取得して表示する（1908）。なお、リンクには、画面に対応するコード（例えば、PRODUCT\_REGISTER\_IO）が含まれており、このコードに対応する画面仕様ドキュメントを取得するものとする。

20

## 【0162】

ステップS2003は、画面仕様ドキュメントの画面遷移図19において、アクション情報のアイコン1703が選択されたか否かを判定する。アイコン1703が選択された場合は、ステップS2004へ処理を移す。アイコン1703が選択されていない場合は、ステップS2005へ処理を移す。

## 【0163】

ステップS2004は、アクション情報を表示する。アクション情報の表示例が1801である。このアクション情報もWebサーバ107から取得される。

## 【0164】

ステップS2005は、データ部品がクリックされたか否かを判定する。具体的には、1704などのデータ部品が選択されたか否かを判定する。データ部品が選択された場合にはステップS2006へ処理を移す。データ部品が選択されていない場合には終了する（操作待ちとなる）。

30

## 【0165】

ステップS2006は、選択されたデータ部品が有している、コードを取得する。例えば、1707の「PRODUCT\_LIST\_IO」を取得する。

## 【0166】

ステップS2007は、取得したコードをもとに、Webサーバに対象の画面仕様ドキュメントを要求し、コードに対応した画面仕様ドキュメントを取得する。また、この取得した画面仕様ドキュメントを表示する。

40

## 【0167】

なお、本実施形態では、Webアプリケーションを生成し構築するシステムとして説明したが、XML形式でやり取りするWebサービスとして動作するシステムであってもよい。

## 【0168】

次に、図21を用いて、本実施形態における、特徴を示す機能構成の一例について説明する。

## 【0169】

なお、図21の機能構成は、プログラム開発装置101の機能構成であるが、プログラム開発が可能な装置であればサーバであってもよい。すなわち、クラウド環境で開発でき

50

るシステムを利用する場合、サーバの機能構成に置き換えることも可能である。すなわち、Webシステムで実行するプログラムを生成する情報処理装置と言い換えることが可能である。また、Webシステムで用いるクライアント装置で表示する画面に対応する入出力定義に従って、Webシステムで実行するプログラムを生成する情報処理装置である。

【0170】

特定部2101は、入出力定義に含まれるアクション情報に従って、遷移先の入出力定義を特定する機能部である。

【0171】

決定部2102は、特定部2101により特定された遷移先の入出力定義に従って、基準となる入出力定義における遷移構成を決定する機能部である。

10

【0172】

画面遷移図生成部2103は、決定部2102により決定された遷移構成に従って、生成されるプログラムで用いられる指定された画面それぞれに対応する画面遷移図を生成する機能部である。

【0173】

取得部2104は、画面遷移図における遷移元又は遷移先にかかるアクション情報を取得する機能部である。

【0174】

また、画面遷移図生成部2103は、取得部2104により取得されたアクション情報を任意に表示可能に、画面遷移図を生成する機能部である。

20

【0175】

また、画面遷移図生成部2103は、前記入出力定義の複数のアクション情報を含み、同一の遷移先に対するアクション情報が複数ある場合には、当該複数のアクション情報をまとめて表示可能に、画面遷移図を生成する機能部である。

【0176】

また、画面遷移図生成部2103は、前記アクション情報を表示させるオブジェクトを、画面遷移図における遷移にかかる位置に配置する機能部である。

【0177】

ドキュメント生成部2105は、前記入出力定義に従って、入出力定義の仕様を表示するドキュメントを生成する機能部である。

30

【0178】

また、画面遷移図生成部2103は、ドキュメント生成部2105で生成されたドキュメントに対応して画面遷移図を生成する機能部である。

【0179】

また、画面遷移図生成部2103は、基準となる入出力定義と、遷移先又は遷移元の入出力定義に対応するオブジェクトを生成する機能部である。

【0180】

前記オブジェクトは、ドキュメント生成部2105は、により生成されたドキュメントとリンクするように生成されるもので、画面遷移図生成部2103により作成される。当該オブジェクトは前記入出力定義のタイプに従って識別可能に生成される。

40

【0181】

前記基準となる入出力定義は、ドキュメント生成部2105でドキュメントを生成する際に用いられる入出力定義であって、画面遷移図生成部2103は、前記基準となる入出力定義に対応するオブジェクトをユーザが認識しやすい形態で配置した画面遷移図を生成する。

【0182】

以上、本実施形態の詳細な説明を終了する。

【0183】

本実施形態では、Webアプリケーションの実行環境で表示する画面ごとの画面遷移図を容易に生成することができる。

50

## 【 0 1 8 4 】

以上のように、前述した実施形態の機能を実現するプログラム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。

## 【 0 1 8 5 】

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

## 【 0 1 8 6 】

プログラム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리카ード、ROM、EEPROM、シリコンディスク等を用いることが出来る。

10

## 【 0 1 8 7 】

また、コンピュータが読み出したプログラムを実行することにより、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。

## 【 0 1 8 8 】

さらに、記録媒体から読み出されたプログラムが、コンピュータに挿入された機能拡張ボードやコンピュータに接続された機能拡張ユニットに備わるメモリに書き込まれた後、そのプログラム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。

20

## 【 0 1 8 9 】

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、ひとつの機器から成る装置に適用しても良い。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。

30

## 【 0 1 9 0 】

上記プログラムの形態は、オブジェクトコード、インタプリタにより実行されるプログラムコード、OS（オペレーティングシステム）に供給されるスクリプトデータ等の形態から成ってもよい。

## 【 0 1 9 1 】

さらに、本発明を達成するための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。なお、上述した各実施形態およびその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。

40

## 【符号の説明】

## 【 0 1 9 2 】

- 1 0 1 プログラム開発装置
- 1 0 2 プログラム開発サーバ
- 1 0 3 データベースサーバ
- 1 0 4 アプリケーションクライアント
- 1 0 5 アプリケーションサーバ
- 1 0 6 ネットワーク
- 1 0 7 Webサーバ
- 2 0 1 CPU

50

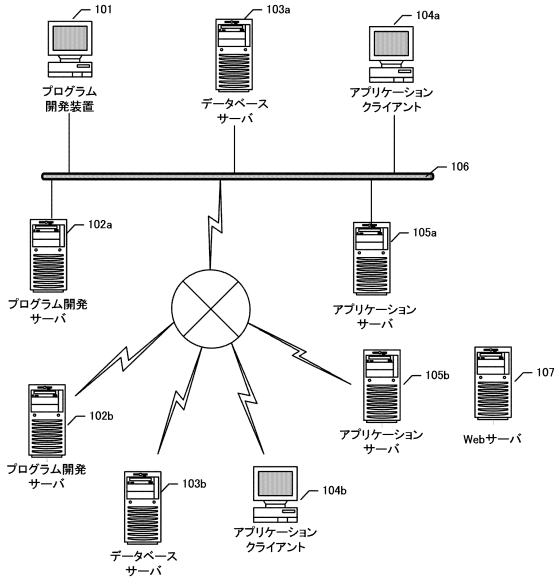
2 0 2 R A M

2 1 1 外部メモリ

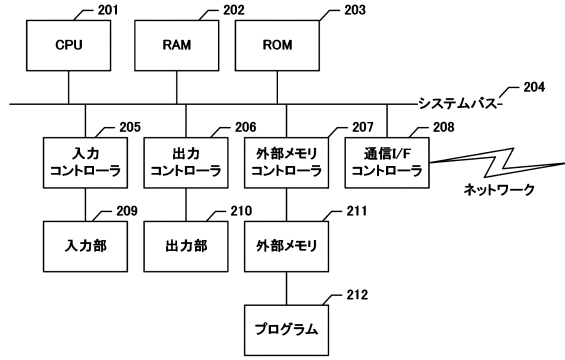
【 図 面 】

【 図 1 】

【 図 2 】



10



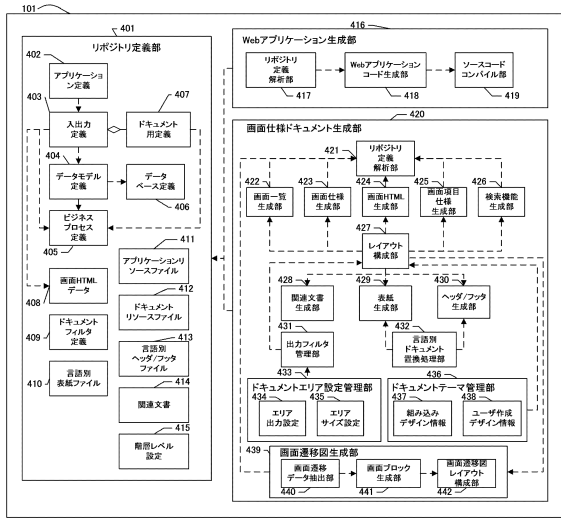
20

30

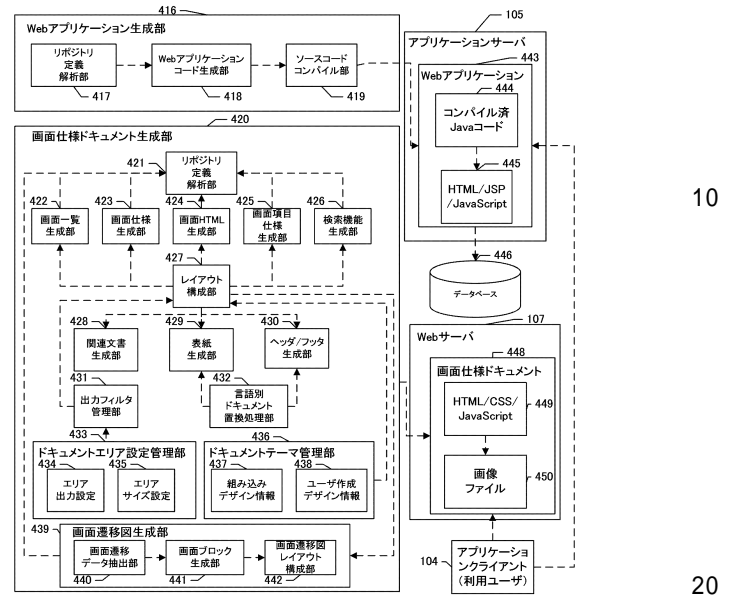
40

50

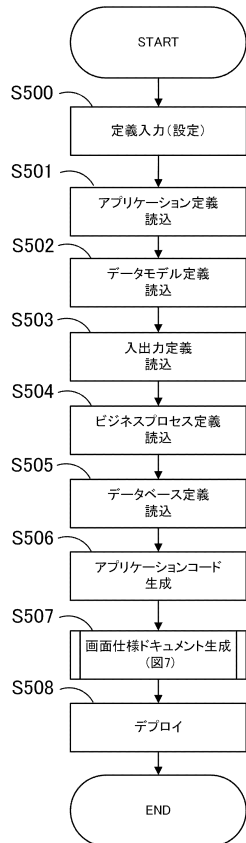
【図 3】



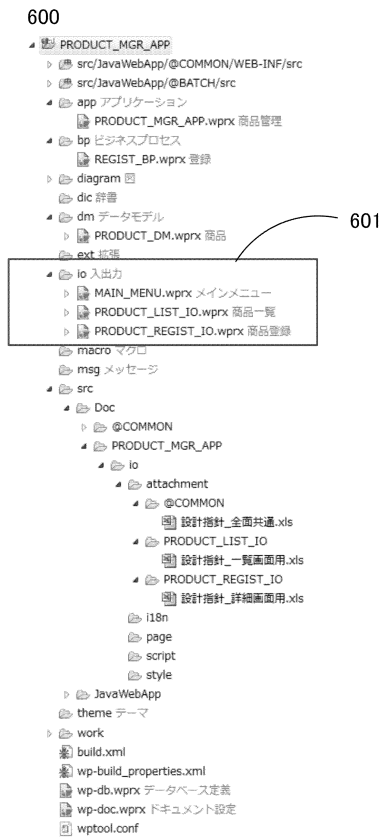
【図 4】



【図 5】



【図 6】



10

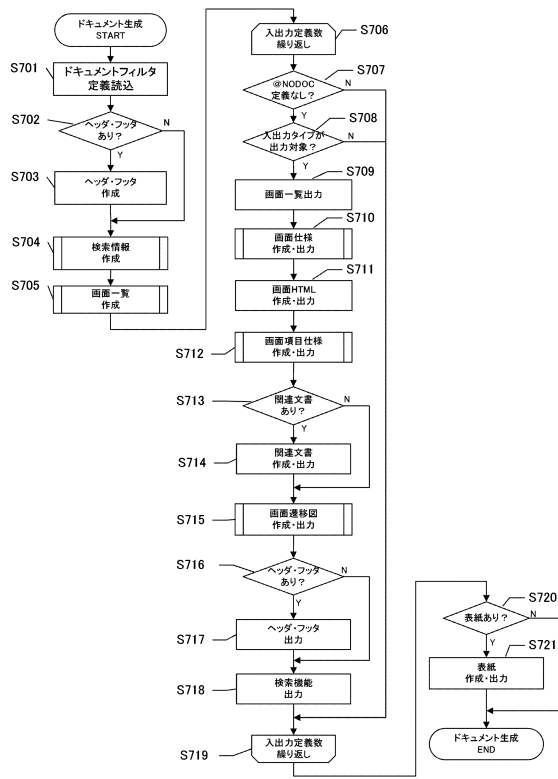
20

30

40

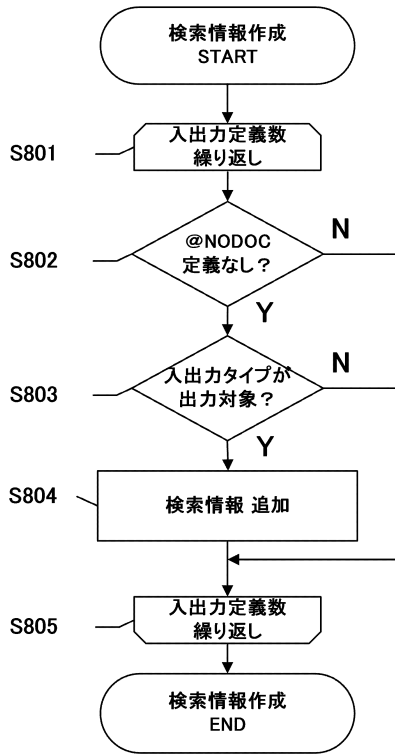
50

【 図 7 】



【 図 8 】

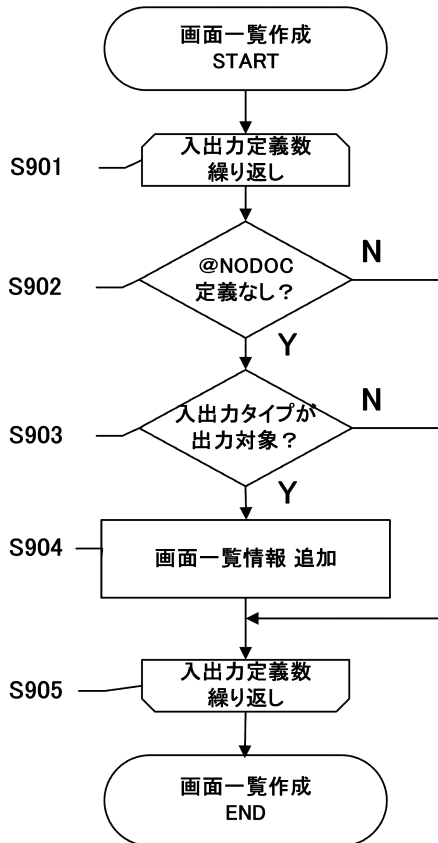
S800



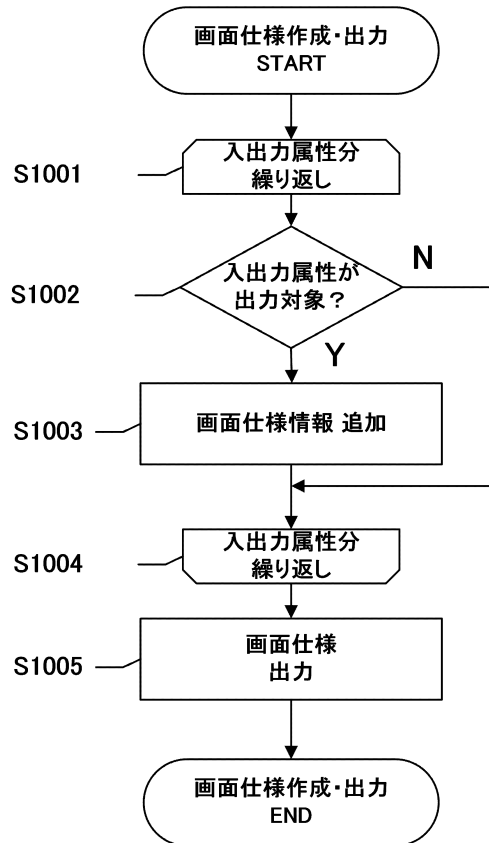
10

20

【 図 9 】



【 図 10 】

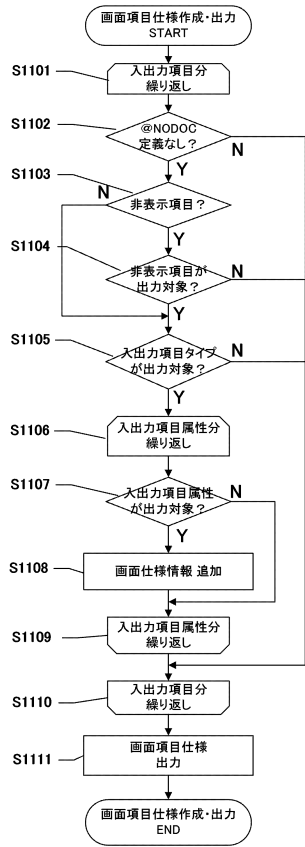


30

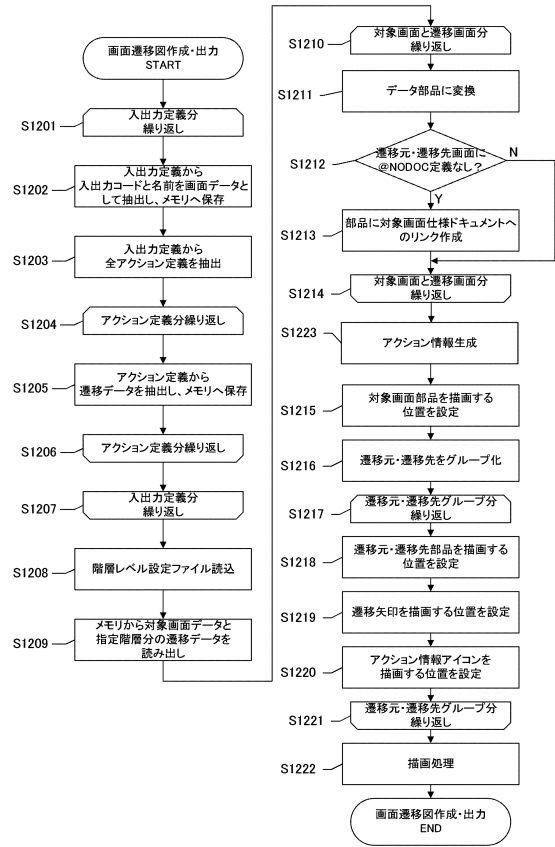
40

50

【図 1 1】



【図 1 2】



10

20

【図 1 3】

アプリケーション定義

(a) アプリケーション設定 1300

コード: PRODUCT\_MGR\_APP  
 名前: 商品管理  
 アプリケーションタイプ: WEB  
 初期入力コード: MAIN\_MENU メインメニュー

(b) 所属入出力一覧 1301

| 入出力一覧  | 所属入出力                  |
|--------|------------------------|
| 入出力コード | 入出力コード                 |
|        | MAIN_MENU メインメニュー      |
|        | PRODUCT_LIST_IO 商品一覧   |
|        | PRODUCT_REGIST_IO 商品登録 |

【図 1 4】

入出力定義例 1400

| コード               | 名前      | 入出力タイプ |
|-------------------|---------|--------|
| MAIN_MENU         | メインメニュー | MENU   |
| PRODUCT_LIST_IO   | 商品一覧    | IO     |
| PRODUCT_REGIST_IO | 商品登録    | IO     |

(a) メインメニュー 1401

| 項目タイプ  | 項目コード           | 名前       | 表示 | 必須 | レベル | 桁数 | 小数桁 | 次入出力                   |
|--------|-----------------|----------|----|----|-----|----|-----|------------------------|
| Aアクション | TO_PRODUCT_ID   | 商品登録ページへ | 表示 | 1  | 0   | -1 |     | PRODUCT_REGIST_IO 商品登録 |
| Aアクション | TO_PRODUCT_LIST | 商品一覧ページへ | 表示 | 1  | 0   | -1 |     | PRODUCT_LIST_IO 商品一覧   |

(b) 商品一覧 1402

| 項目タイプ  | 項目コード         | 名前   | 表示 | 必須 | レベル | 桁数 | 小数桁 | 次入出力 | データモジュール               | データモジュール項目コード     |
|--------|---------------|------|----|----|-----|----|-----|------|------------------------|-------------------|
| システム   | PRODUCT_ID    | 商品ID | 表示 | 1  | 0   | 0  |     |      |                        |                   |
| Aアクション | TO_PRODUCT_ID | 商品ID | 表示 | 2  | 0   | -1 |     |      | PRODUCT_REGIST_IO 商品登録 |                   |
| IO入出力  | PRODUCT_ID    | 商品ID | 表示 | 2  | 0   | -1 |     |      | PRODUCT_ID 商品ID        | PRODUCT_ID 商品ID   |
| IO入出力  | PRODUCT_NAME  | 商品名  | 表示 | 2  | 0   | -1 |     |      | PRODUCT_ID 商品ID        | PRODUCT_NAME 商品名  |
| IO入出力  | PRODUCT_PRICE | 価格   | 表示 | 2  | 0   | -1 |     |      | PRODUCT_ID 商品ID        | PRODUCT_PRICE 価格  |
| IO入出力  | PRODUCT_STOCK | 在庫数  | 表示 | 2  | 0   | -1 |     |      | PRODUCT_ID 商品ID        | PRODUCT_STOCK 在庫数 |
| Aアクション | UPDATE        | 更新   | 表示 | 1  | 0   | -1 |     |      | PRODUCT_LIST_IO 商品一覧   |                   |
| Aアクション | BACK          | 戻る   | 表示 | 1  | 0   | -1 |     |      |                        |                   |

(c) 商品登録 1403

| 項目タイプ  | 項目コード         | 名前   | 表示 | 必須 | レベル | 桁数 | 小数桁 | 次入出力 | データモジュール             | データモジュール項目コード     |
|--------|---------------|------|----|----|-----|----|-----|------|----------------------|-------------------|
| IO入出力  | PRODUCT_ID    | 商品ID | 表示 | 1  | 0   | -1 |     |      | PRODUCT_ID 商品ID      | PRODUCT_ID 商品ID   |
| IO入出力  | PRODUCT_NAME  | 商品名  | 表示 | 1  | 0   | -1 |     |      | PRODUCT_ID 商品ID      | PRODUCT_NAME 商品名  |
| IO入出力  | PRODUCT_PRICE | 価格   | 表示 | 1  | 0   | -1 |     |      | PRODUCT_ID 商品ID      | PRODUCT_PRICE 価格  |
| IO入出力  | PRODUCT_STOCK | 在庫数  | 表示 | 1  | 0   | -1 |     |      | PRODUCT_ID 商品ID      | PRODUCT_STOCK 在庫数 |
| Aアクション | REGIST_ACT    | 登録   | 表示 | 1  | 0   | -1 |     |      | PRODUCT_LIST_IO 商品一覧 |                   |
| Aアクション | BACK          | 戻る   | 表示 | 1  | 0   | -1 |     |      |                      |                   |

30

40

@NODOC 定義例 1404

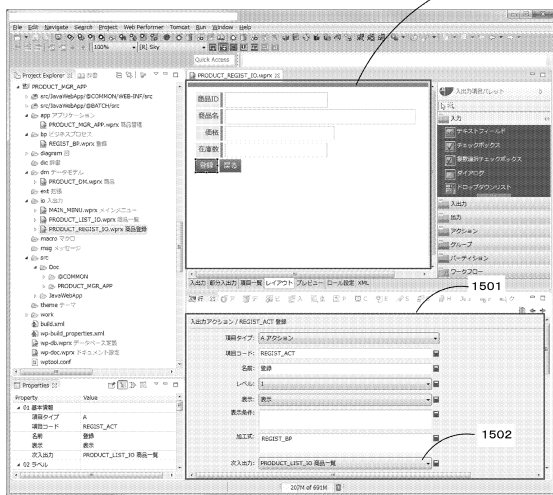
仕様: @NODOC

50

【 図 1 5 】

1500

入出カレイアウト定義例(PRODUCT REGIST IO)



【 図 1 6 】

1600

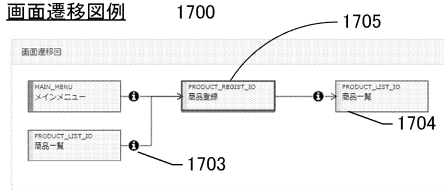
画面遷移階層レベル設定

#画面遷移階層レベル設定  
#画面仕様ドキュメントに表示する画面遷移図の階層レベルを設定することができます。  
#pageflow.level=2  
pageflow.level=1

1601

【 図 1 7 】

画面遷移図例



画面遷移図エリア(HTML) 1701

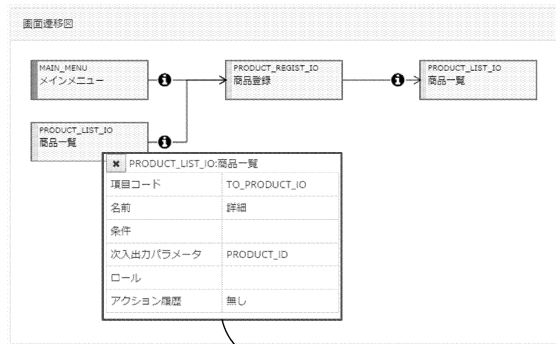
```
<div class="wp-pageflow-canvas-area">
  <canvas id="wp-pageflow-canvas"></canvas>
</div>
```

画面遷移情報例 1702

```
var pageflowText=[code: "PRODUCT_REGIST_IO",name: "商品登録",ioType:"IO",isWorkflow:false,exists:true,
in:[code: "MAIN_MENU",name: "メインメニュー",ioType:"MENU",isWorkflow:false,exists:true],
actions:[code: "TO_PRODUCT_IO",name: "商品登録ページへ",
role:"condition",parameter:"noHistory:""1"]],code: "PRODUCT_LIST_IO",name: "商品一覧",
ioType:"IO",isWorkflow:false,exists:true,
actions:[code: "TO_PRODUCT_IO",name: "詳細",role:"condition",parameter:"PRODUCT_ID","noHistory:""1"]],
out:[code: "PRODUCT_LIST_IO",name: "商品一覧",ioType:"IO",isWorkflow:false,exists:true],
actions:[code: "REGIST_ACT",name: "登録",role:"condition",parameter:"noHistory:""1"]]];
```

【 図 1 8 】

アクション情報表示 1800



1801

10

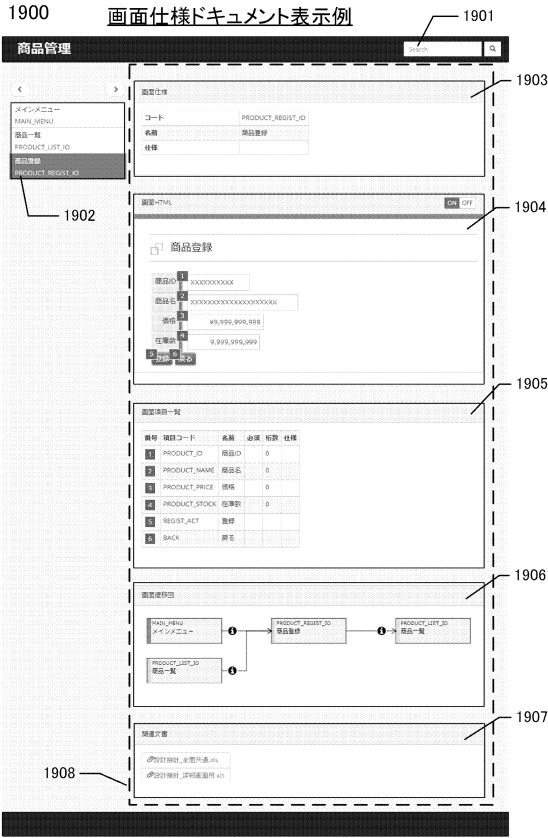
20

30

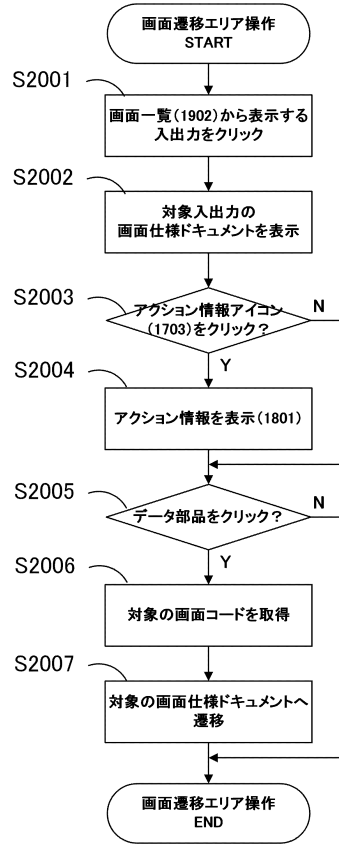
40

50

【図 19】



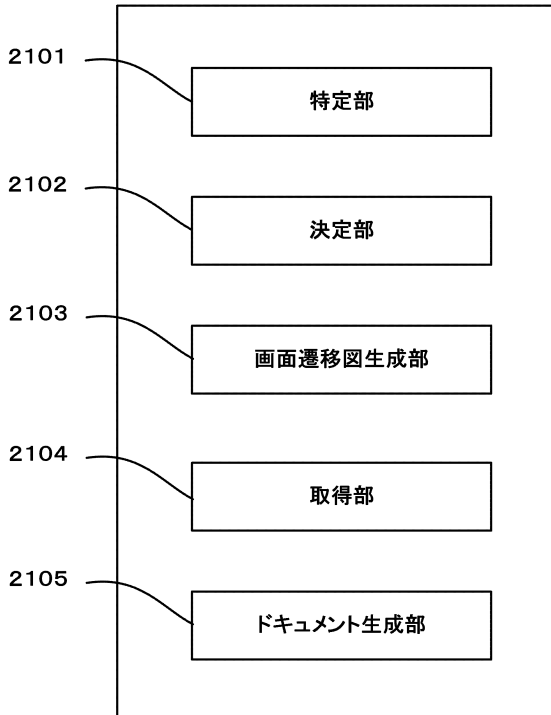
【図 20】



10

20

【図 21】



30

40

50

---

フロントページの続き

キヤノンITソリューションズ株式会社内

審査官 金木 陽一

- (56)参考文献 特開2012-221107(JP,A)  
特開2008-140408(JP,A)  
特開2016-173721(JP,A)  
特開2005-165934(JP,A)
- (58)調査した分野 (Int.Cl., DB名)  
G06F 8/30  
G06F 8/38