

(19) 日本国特許庁(JP)

(12) 特 許 公 報(B2)

(11) 特許番号

特許第4902282号  
(P4902282)

(45) 発行日 平成24年3月21日 (2012.3.21)

(24) 登録日 平成24年1月13日 (2012.1.13)

(51) Int.Cl. F I  
**G 0 6 F 9/445 (2006.01)** G O 6 F 9/06 6 1 O A  
 G O 6 F 9/06 6 1 O Q

請求項の数 20 (全 39 頁)

<p>(21) 出願番号 特願2006-192042 (P2006-192042)                  (22) 出願日 平成18年7月12日 (2006.7.12)                  (65) 公開番号 特開2008-21111 (P2008-21111A)                  (43) 公開日 平成20年1月31日 (2008.1.31)                  審査請求日 平成21年1月27日 (2009.1.27)</p>	<p>(73) 特許権者 000005108                  株式会社日立製作所                  東京都千代田区丸の内一丁目6番6号                  (74) 代理人 100064414                  弁理士 磯野 道造                  (72) 発明者 南松 利博                  神奈川県横浜市戸塚区戸塚町5030番地                  株式会社日立製作所                  ソフトウェア事業部内                  審査官 林 毅</p>
--	--

最終頁に続く

(54) 【発明の名称】 業務システム構成変更方法、管理コンピュータ、および、業務システム構成変更方法のプログラム

(57) 【特許請求の範囲】

【請求項1】

複数の業務プログラムを1台以上の業務コンピュータが使用することによって実現される業務システムにおける前記業務プログラムの構成を、前記業務コンピュータと通信可能な管理コンピュータがスクリプトを用いて変更する業務システム構成変更方法であって、前記管理コンピュータは、処理部と記憶部を備え、

前記記憶部は、

前記業務システムにおける前記業務プログラムの構成を示す業務システム構成情報と、それぞれの前記業務コンピュータごとに、配備可能なOSの情報を関連付けた第1の情報と、

それぞれの前記業務プログラムごとに、その動作のために必要な他の業務プログラムである前提プログラムの情報を関連付けた第2の情報と、

前記スクリプトを構成し、少なくとも、実行する対象の業務コンピュータ名をパラメタとして入力可能な複数のスクリプト部品と、

それぞれのスクリプトテンプレートが前記スクリプト部品の代表的な組み合わせおよび実行順序を示す複数のスクリプトテンプレートと、を記憶し、

前記処理部は、前記業務プログラムを配備する場合、前記第1の情報を参照して、当該業務プログラムが動作するためのOSを配備可能な前記業務コンピュータを選択し、

前記第1の情報を参照して、当該業務プログラムが動作するためのOSを配備可能な前記業務コンピュータを選択し、

前記第2の情報を参照して、当該業務プログラムの動作のために必要な前記前提プログラムの情報を取得し、

前記業務システム構成情報において、前記業務プログラムと前記前提プログラムについての、前記業務システムの構成の変更前と変更後に関する差分をとり、その差分の変更の実行に必要な前記スクリプト部品を複数取得し、複数の前記スクリプトテンプレートのうちの1つ以上を用いて、当該取得した複数のスクリプト部品を組み合わせることで前記スクリプトを生成し、

当該スクリプトを用いて前記業務プログラムを配備することを特徴とする業務システム構成変更方法。

【請求項2】

前記処理部は、  
すでに配備されている前記業務プログラムのいずれかを削除する場合、  
その業務プログラムの削除に必要なスクリプト部品を取得することで削除用のスクリプトを生成し、

当該削除用のスクリプトを用いて前記業務プログラムを削除することを特徴とする請求項1に記載の業務システム構成変更方法。

【請求項3】

前記処理部は、  
前記管理コンピュータに接続された監視コンピュータの入力画面からオペレータによって入力された業務プログラムの配備指示に基づいて、前記業務プログラムを配備することを特徴とする請求項1に記載の業務システム構成変更方法。

【請求項4】

前記処理部は、  
前記管理コンピュータに接続された監視コンピュータの入力画面からオペレータによって入力された業務プログラムの移動指示に基づき、  
請求項1に記載の業務システム構成変更方法による業務プログラムの配備と、請求項2に記載の業務システム構成変更方法による業務プログラムの削除を組み合わせることで、前記業務プログラムを移動することを特徴とする業務システム構成変更方法。

【請求項5】

前記処理部は、  
前記業務システム構成情報を参照し、前記スクリプトテンプレートを構成する前記スクリプト部品のうち不要分を削除し、  
その不要なスクリプト部品を削除したスクリプトテンプレートを複数集めることで前記スクリプトを生成することを特徴とする請求項1に記載の業務システム構成変更方法。

【請求項6】

前記処理部は、  
前記記憶部に記憶された変更前および変更後の前記業務システム構成情報を参照して、その取得したスクリプト部品の各パラメタを設定することを特徴とする請求項1に記載の業務システム構成変更方法。

【請求項7】

前記処理部は、  
前記第1の情報を参照して、当該業務プログラムが動作するためのOSを配備可能な前記業務コンピュータを選択する場合、

該当する前記業務コンピュータが複数あるときは、  
当該業務プログラムが動作するために必要なOS修正用のパッチファイルを最も多く有する前記業務コンピュータを選択することを特徴とする請求項1に記載の業務システム構成変更方法。

【請求項8】

前記記憶部は、  
前記スクリプト部品ごとに、前記業務コンピュータのリポートが必要な場合はリポート

10

20

30

40

50

フラグを記憶し、

前記処理部は、

前記スクリプトを用いて前記業務プログラムや前記前提プログラムを配備するときに、前記スクリプト部品を実行した場合、そのスクリプト部品が前記リポートフラグをともなっていたとき、あるいは、前記リポートフラグをともなっていないが前記業務コンピュータにリポート要求が出ていたときに、その業務コンピュータをリポートすることを特徴とする請求項 1 に記載の業務システム構成変更方法。

【請求項 9】

前記第 1 の情報は、OS 情報管理テーブルであることを特徴とする請求項 1 に記載の業務システム構成変更方法。

10

【請求項 10】

前記第 2 の情報は、

前記業務プログラムが動作するために必要な OS 修正用のパッチファイルの情報、前記前提プログラムの情報、および、前記業務プログラムが連携するプログラムの情報を定義したプログラム構成管理テーブルであることを特徴とする請求項 1 に記載の業務システム構成変更方法。

【請求項 11】

前記第 2 の情報は、

前記業務コンピュータの OS の情報、OS 修正用のパッチファイルの情報、および、配備済みのプログラムの情報を管理する物理構成情報管理テーブルあることを特徴とする請求項 1 に記載の業務システム構成変更方法。

20

【請求項 12】

1 台以上の業務コンピュータと通信可能で、複数の業務プログラムを前記 1 台以上の業務コンピュータが使用することによって実現される業務システムにおける前記業務プログラムの構成を、スクリプトを用いて変更する管理コンピュータであって、

処理部と記憶部を備え、

前記記憶部は、

前記業務システムにおける前記業務プログラムの構成を示す業務システム構成情報と、それぞれの前記業務コンピュータごとに、配備可能な OS の情報を関連付けた第 1 の情報と、

30

それぞれの前記業務プログラムごとに、その動作のために必要な他の業務プログラムである前提プログラムの情報を関連付けた第 2 の情報と、

前記スクリプトを構成し、少なくとも、実行する対象の業務コンピュータ名をパラメタとして入力可能な複数のスクリプト部品と、

それぞれのスクリプトテンプレートが前記スクリプト部品の代表的な組み合わせおよび実行順序を示す複数のスクリプトテンプレートと、を記憶し、

前記処理部は、

前記業務プログラムを配備する場合、

前記第 1 の情報を参照して、当該業務プログラムが動作するための OS を配備可能な前記業務コンピュータを選択し、

40

前記第 2 の情報を参照して、当該業務プログラムの動作のために必要な前記前提プログラムの情報を取得し、

前記業務システム構成情報において、前記業務プログラムと前記前提プログラムについての、前記業務システムの構成の変更前と変更後に関する差分をとり、その差分の変更の実行に必要な前記スクリプト部品を複数取得し、複数の前記スクリプトテンプレートのうちの 1 つ以上を用いて、当該取得した複数のスクリプト部品を組み合わせることで前記スクリプトを生成し、

当該スクリプトを用いて前記業務プログラムを配備する

ことを特徴とする管理コンピュータ。

【請求項 13】

50

前記処理部は、  
 すでに配備されている前記業務プログラムのいずれかを削除する場合、  
 その業務プログラムの削除に必要なスクリプト部品を取得することで削除用のスクリプトを生成し、  
 当該削除用のスクリプトを用いて前記業務プログラムを削除することを特徴とする請求項 1 2 に記載の管理コンピュータ。

## 【請求項 1 4】

前記処理部は、  
 前記管理コンピュータに接続された監視コンピュータの入力画面からオペレータによって入力された業務プログラムの配備指示に基づいて、前記業務プログラムを配備することを特徴とする請求項 1 2 に記載の管理コンピュータ。

10

## 【請求項 1 5】

前記管理コンピュータは、  
 前記管理コンピュータに接続された監視コンピュータの入力画面からオペレータによって入力された業務プログラムの移動指示に基づき、  
 請求項 1 2 に記載の管理コンピュータによる業務プログラムの配備と、請求項 1 3 に記載の管理コンピュータによる業務プログラムの削除を組み合わせることで、前記業務プログラムを移動することを特徴とする管理コンピュータ。

## 【請求項 1 6】

前記処理部は、  
 前記業務システム構成情報を参照し、前記スクリプトテンプレートを構成する前記スクリプト部品のうち不要分を削除し、  
 その不要なスクリプト部品を削除したスクリプトテンプレートを複数集めることで前記スクリプトを生成することを特徴とする請求項 1 2 に記載の管理コンピュータ。

20

## 【請求項 1 7】

前記処理部は、  
 前記記憶部に記憶された変更前および変更後の前記業務システム構成情報を参照して、その取得したスクリプト部品の各パラメータを設定することを特徴とする請求項 1 2 に記載の管理コンピュータ。

## 【請求項 1 8】

前記処理部は、  
 前記第 1 の情報を参照して、当該業務プログラムが動作するための OS を配備可能な前記業務コンピュータを選択する場合、  
 該当する業務コンピュータが複数あるときは、  
 当該業務プログラムが動作するために必要な OS 修正用のパッチファイルを最も多く有する前記業務コンピュータを選択することを特徴とする請求項 1 2 に記載の管理コンピュータ。

30

## 【請求項 1 9】

前記記憶部は、  
 前記スクリプト部品ごとに、前記業務コンピュータのリポートが必要な場合はリポートフラグを記憶し、  
 前記処理部は、  
 前記スクリプトを用いて前記業務プログラムや前記前提プログラムを配備するときに、前記スクリプト部品を実行した場合、そのスクリプト部品が前記リポートフラグをともなっていたとき、あるいは、前記リポートフラグをともなっていないが前記業務コンピュータにリポート要求が出ていたときに、その業務コンピュータをリポートすることを特徴とする請求項 1 2 に記載の管理コンピュータ。

40

## 【請求項 2 0】

請求項 1 ないし請求項 1 1 のいずれか 1 項に記載の業務システム構成変更方法を前記管理コンピュータに実行させるためのプログラム。

50

## 【発明の詳細な説明】

## 【技術分野】

## 【0001】

本発明は、1台以上のコンピュータ装置に配備された複数の業務プログラムから構成される業務システムにおいて、業務システムへ影響を与えることなく、または与える影響を少なくして、任意の業務プログラムの配備先の変更などを行う技術に関する。

## 【背景技術】

## 【0002】

近年、企業などでは、銀行のATM(Automatic Teller Machine)システムや小売業のPOS(Point Of Sales)システムなどの業務システムが、多く使用されている。そして、そういった業務システムでは、規模が大きい場合、そのための業務プログラムやデータなど(以下、単に「業務プログラム」ともいう。)を複数のコンピュータ装置に配備することになる。

10

## 【0003】

また、業務システムにおいては、業務プログラムの変更(追加、削除などを含む)など、構成を変更する必要が生じることもある。その場合、業務システムの構成の変更やそれらにともなう環境設定などをすべて手動で行うと、少なからず手間がかかってしまうので、それらなるべく多くを自動化して、容易に構成の変更を行うことが望まれている。

## 【0004】

例えば、複数のコンピュータ装置を使用する分散システムにおけるイベント処理において、アクションの連鎖によって動的にデータフローを決定することで環境設定の一部を自動化するシステムが知られている(例えば特許文献1)。

20

【特許文献1】特開2001-282556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、イベントオブジェクト受信時に実行されるアクションを、イベントオブジェクトのタイプにより実行すべきアクションの中から選択してアクションの連鎖を実現するには、事前にイベントオブジェクトのタイプを事前に把握しなくてはならない。

## 【0006】

ところが、現実には、イベントが製品ごとに異なっており、イベントオブジェクトの形式を統一することは不可能に近い。また、イベントオブジェクトの形式を統一できた場合でも、イベントを発行するプログラムは無数に存在すると共に今後も増加していく。さらに、イベントを発行するプログラムの開発者は世界中に存在しており、そのプログラムのタイプは次々に増え続けていく。このタイプを事前に全て把握し、アクション・属性記憶部に記憶される入力パターンの定義に入力することは、極めて困難である。

30

## 【0007】

そこで、本発明は、前記問題点に鑑みてなされてものであり、業務システムの構成が大規模や複雑な場合でも、その構成を容易に変更することを目的とする。

## 【課題を解決するための手段】

## 【0008】

前記課題を解決するために、本発明は、複数の業務プログラムを1台以上の業務コンピュータが使用することによって実現される業務システムにおける業務プログラムの構成を、業務コンピュータと通信可能な管理コンピュータがスクリプトを用いて変更する業務システム構成変更方法である。

40

管理コンピュータは、処理部と記憶部を備え、記憶部は、業務システムにおける業務プログラムの構成を示す業務システム構成情報と、それぞれの業務プログラムの稼働条件と、その稼働条件に対応した業務コンピュータを関連付けた第1の情報と、それぞれの業務プログラム、および、それらに対応した前提プログラム情報を、プログラムレベルとして関連付けた第2の情報と、スクリプトを構成し、少なくとも、実行する対象の業務コンピ

50

ユーザ名をパラメタとして入力可能な複数のスクリプト部品と、を記憶する。

【0009】

そして、処理部は、業務プログラムを配備する場合、第1の情報を参照して、当該業務プログラムの稼動条件に対応した業務コンピュータを選択し、第2の情報を参照して、当該業務プログラムに必要な前提プログラムを取得し、業務システム構成情報において、業務プログラムと前提プログラムについての、業務システムの構成の変更前と変更後に関する差分をとり、その差分の変更の実行に必要なスクリプト部品を複数取得し、前記前提プログラム情報に基づいて前記スクリプト部品の順序を決定することでスクリプトを生成し、当該スクリプトを用いて業務プログラムを配備する。

その他の手段については後記する。

10

【発明の効果】

【0010】

本発明によれば、業務システムの構成を容易に変更することができる。

【発明を実施するための最良の形態】

【0011】

以下、図面を参照しながら、スクリプトを用いて業務システム構成変更方法を実現するプログラム配備システムの一実施形態について説明する(適宜、参照図以外の図も参照)。

なお、業務システムとは、ネットワークにより接続された複数のコンピュータ間で業務を実行するプログラム(以下、業務プログラムという。)が連携しながら実行することで稼動するシステムである。

20

【0012】

また、スクリプトとは、対象となるプログラムのインストールや設定などの手順を、コンピュータで実行可能な形式で記述した簡易プログラムである。さらに、スクリプトにパラメタ(業務プログラム名、対象コンピュータ名など)を設定する前の状態のものを、スクリプトシナリオという。

【0013】

スクリプトは、複数のスクリプトテンプレート(スクリプトを構成するパーツのひな形。図22~図26で詳述)から構成され、各スクリプトテンプレートは、複数のスクリプト部品(スクリプトテンプレートを構成する部品の単位。図21で詳述)から構成される。

30

【0014】

図1は、本実施形態のプログラム配備システムの全体構成を示す図である。図1に示すように、本実施形態のプログラム配備システムSは、監視コンピュータ1、管理コンピュータ11、および、複数の業務コンピュータ4から構成されている。そして、管理コンピュータ11と複数の業務コンピュータ4は、ネットワーク3により接続されている。

監視コンピュータ1は、業務システムの構成や業務プログラムを、ネットワーク3を介して、複写、移動、削除などするための操作画面を表示するG U I (Graphical User Interface) 10を備えている。

40

【0015】

管理コンピュータ11は、表示部12と、業務システム管理部13と、データ処理部14と、イベント処理部15と、エージェント通信部16とを有している。

表示部12は、監視コンピュータ1のG U I 10(入力画面)に業務システムの構成を表示させたり、G U I 10からの要求を受け付けたりする処理部である。業務システム管理部13は、業務コンピュータ4で稼動する業務プログラム31の構成を管理し、業務構成を変更した場合にシステム構成情報22(業務システム構成情報)の更新を行う処理部である。

【0016】

データ処理部14は、プログラムライブラリ管理テーブル21~スクリプト25に対して、各種のデータの読み込みや書き込みを行う処理部である。イベント処理部15は、G

50

UI 10からの要求や、イベントによる要求を実現するために、スクリプト25の生成を行う処理部である。エージェント通信部16は、業務コンピュータ4のエージェント処理部30と通信し、要求やデータの送受信を行う処理部である。

#### 【0017】

管理コンピュータ11が取り扱うデータとして、業務コンピュータ4にリモートインストールやディスクコピーによって、業務プログラムを配備するためのインストールイメージ(プログラムをインストール形式でアーカイブされたもの)やディスクイメージ(ハードディスクの内容をそのまま複製(コピー)することでインストールできるもの)を保持するプログラムライブラリ管理テーブル21と、様々な業務システムの構成情報を格納するシステム構成情報22と、業務プログラムごとに配備や削除、環境設定などの制御を行うスクリプト部品を保持するスクリプトライブラリ管理テーブル23と、用途ごとにスクリプト部品の組み合わせを定義したスクリプトテンプレート24と、テンプレートとスクリプト部品を組み合わせで生成したスクリプト25がある。

10

なお、本実施形態でいう各テンプレートとは、それぞれにおける代表的な例(ひな形)、という意味である。

#### 【0018】

業務コンピュータ4は、業務システムのための実行プログラムである業務プログラム31と、管理コンピュータ11のエージェント通信部16と通信するためのエージェント処理部30が格納されている。なお、コンピュータによっては、業務プログラム31が存在しないものもあり、複数存在するものもある。

20

#### 【0019】

図2は、本実施形態の監視コンピュータと管理コンピュータのハードウェア構成を示した図である。管理コンピュータ11は、CPU(Central Processing Unit)2721(処理部)、メモリ2722(記憶部)、記憶装置2723(記憶部)、表示インターフェイス2724、ディスクインターフェイス2726、通信インターフェイス2727を備えた管理コンピュータ11上に構築することができる。また、プログラム配備システムSのGUI10は、CPU2731、メモリ2732、記憶装置2733、表示インターフェイス2734を備える監視コンピュータ1上に構築することができる。

#### 【0020】

この場合、プログラム配備用プログラム6は、CPU2721が記憶装置2723に格納されていた各プログラムをメモリ2722上にロードして実行することで実現できる。また、外部のデータベース8(記憶部)とはディスクインターフェイス2726を介して接続され、業務コンピュータとは通信インターフェイス2727を介して接続される。一方、監視コンピュータ1において、GUI10は、CPU2731が記憶装置2733に格納されていたプログラムをメモリ2732上にロードして実行する。

30

また、図1のプログラムライブラリ管理テーブル21~スクリプト25は、たとえば、データベース8に格納される。

#### 【0021】

図3は、本実施形態の業務コンピュータのハードウェア構成を示した図である。業務コンピュータ4において、エージェント処理部30は、CPU2741、メモリ2742、記憶装置2743、通信インターフェイス2747によって実現することができる。つまり、エージェント処理部30は、CPU2741が記憶装置2743に格納されていたエージェントプログラムをメモリ2742上にロードして実行することで、実現される。また、業務コンピュータ4は、管理コンピュータ11と、通信インターフェイス2747およびネットワーク3を介して接続される。

40

#### 【0022】

図4は、システム構成情報22が保持するテーブルを示す図である。

システム構成情報22において、プログラム構成管理テーブル101(第2の情報)は、業務プログラムが前提とするOS(Operating System)パッチの情報や、前提とするプログラムの情報、および業務プログラムが連携するプログラムの情報をユーザが定義(予

50

め入力)したものである。

【0023】

なお、パッチとは、プログラムの修正を行なうために、修正による変更点(差分情報)のみを抜き出して列挙したファイル(パッチファイル)のことである。

また、プログラムAがプログラムBを前提とする、とは、プログラムAが動作するにはプログラムBが必要である、という意味である。

【0024】

業務構成情報管理テーブル102は、各業務を構成する業務プログラム名と、業務プログラムが稼動するコンピュータの対応づけをユーザが定義したものである。

OS情報管理テーブル103(第1の情報)は、業務コンピュータ4に配備可能なOS情報をユーザが定義したものである。

物理構成情報管理テーブル104(第2の情報)は、プログラム配備システムSが管理する業務コンピュータ4のOS情報や、OSパッチ情報、配備済みのプログラム情報などの物理情報を収集したものである。

【0025】

プログラム構成差分情報管理テーブル105は、プログラム配備先コンピュータのプログラム構成の変更前と変更後の差分情報を保持するものである。

配備対象プログラム管理テーブル106は、業務プログラムを複写、移動する場合に配備先コンピュータに必要なプログラムの情報を保持するものである。

【0026】

データモデルテンプレート107は、業務システムに関連するデータの構造を定義したものである。

データモデル108は、業務システムのデータをデータモデルテンプレート107に取り込んで生成したものである。換言すれば、データモデルテンプレート107において、必要に応じ、各プログラムなどを追加、削除したり、データを穴埋めしたりして作成したものが、データモデル108となる。

プログラム稼動最大構成テンプレート109は、業務プログラム31が稼動に必要なプログラム構成を定義したものである。

【0027】

図5は、プログラム稼動最大構成テンプレート109を示す図である。業務プログラム202が稼動する場合に必要なプログラムの構成が最大となるのは、OS206と、業務プログラム202が前提とするOSパッチ205と、業務プログラム202が前提とする業務プログラムである前提プログラム204と、前提プログラム204のパッチである前提プログラムパッチ203と、業務プログラム202と、業務プログラム202のパッチである業務プログラムパッチ201とを組み合わせたときである。

【0028】

なお、OSパッチ205や前提プログラム204、前提プログラムパッチ203、業務プログラムパッチ201は存在しない場合もあり、複数存在する場合もある。本テンプレートを用いて、ある業務プログラムを配備する場合に、業務コンピュータ4に配備する必要があるプログラム情報として、配備対象プログラム管理テーブル106を生成する。

【0029】

図6は、プログラムライブラリ管理テーブル21を示す図である。プログラムライブラリ(不図示)は、OSやOSパッチ、業務プログラム、業務プログラムのパッチ等のディスクイメージやインストールイメージを格納し、管理している。そして、プログラムライブラリ管理テーブル21は、ディスクイメージやインストールイメージを識別するプログラムライブラリID301を保持している。

なお、OSのディスクイメージは、エージェント処理部30をインストールした状態のディスクイメージとし、OSの配備時にはエージェント処理部30も同時に配備されるものとする。

【0030】

10

20

30

40

50

プログラムライブラリ管理テーブル 2 1 は、ユーザの定義により、プログラム名 3 0 3 とそのディスクイメージを配備可能な OS 名を示す対象 OS 3 0 4 と、配備した場合に業務コンピュータ 4 のリポート（再起動）が必要なのか否かを示すリポートフラグ 3 0 5 の情報を保持する。なお、複数の OS に対応したプログラムでは、それぞれにプログラムライブラリ ID 3 0 1 が割り当てられる。

【 0 0 3 1 】

また、プログラムライブラリ管理テーブル 2 1 では、OS、OS パッチ、業務プログラムといったプログラムレベル 3 0 2 でレベル分けを行っており、プログラムレベル 1 は OS、プログラムレベル 2 は OS パッチ、プログラムレベル 3 は業務プログラムおよび業務プログラムのパッチとする。プログラムレベルは、当該プログラムライブラリ管理テーブル 2 1 のプログラム名 3 0 3 がプログラム構成管理テーブル 1 0 1（図 1 1 参照）に保持される OS 名 8 0 2 と一致する場合にはプログラムレベル 1 を、前提 OS パッチ名 8 0 3 と一致する場合にはプログラムレベル 2 を、それらと一致しない場合にはプログラムレベル 3 が割り当てられる。

10

リポートフラグ 3 0 5 は、ユーザが定義しない場合に初期値として OFF（リポートしない）をセットするが、プログラムを配備した際にプログラムがリポート要求を返す場合はフラグが OFF の場合でもリポートを行うものとする。

【 0 0 3 2 】

図 7 は、業務プログラムごとに各種スクリプト部品を格納したスクリプトライブラリ管理テーブル 2 3 を示す図である。スクリプトライブラリ管理テーブル 2 3 に格納するスクリプト部品は、図 2 1 の例で示すような、プログラムを制御するスクリプトであり、ユーザによって定義されるものである。

20

【 0 0 3 3 】

スクリプト部品は、業務プログラム名 4 0 1 に対応するプログラムライブラリ ID 4 0 2 ごとに、対象プログラム（対象となる業務プログラム、OS、OS パッチ）のディスクイメージあるいはインストールイメージを配備する配備用スクリプト部品 4 0 3 と、対象プログラムを削除する削除用スクリプト部品 4 0 4 と、対象プログラムの環境設定を行う環境設定用スクリプト部品 4 0 5 と、対象プログラムの連携先の情報を設定する連携プログラム設定用スクリプト部品 4 0 6 と、対象プログラムを連携先とするプログラムの連携先を変更する連携元プログラム設定変更用スクリプト部品 4 0 7 と、対象プログラムを開始するプログラム開始用スクリプト部品 4 0 8 と、対象プログラムを停止するプログラム停止用スクリプト部品 4 0 9 が存在し、プログラムライブラリ ID 4 0 2 ごとに格納される。

30

【 0 0 3 4 】

なお、それぞれのスクリプト部品は、異なる OS ごとにそれぞれ用意しておいても良い。

また、業務プログラムの配備に必要なないスクリプト部品は存在させなくても良い。

さらに、配備用スクリプト部品 4 0 3 は、ユーザ定義の無い場合は、プログラムライブラリ ID 4 0 2 のディスクイメージやインストールイメージをリモートインストール（ネットワーク 3 経由でのインストール）するものである。

40

【 0 0 3 5 】

図 8 は、業務コンピュータ 4 に配備可能な OS の情報を保持した OS 情報管理テーブル 1 0 3 を示す図である。本テーブルでは、管理コンピュータ 1 1 で管理する業務コンピュータ 4 ごとに、ユーザが定義した入れ替え可能（配備可能）な OS（稼動条件）の情報を保持している。

【 0 0 3 6 】

図 9 は、業務システムを構成する業務プログラムの情報と、業務プログラムが稼動するコンピュータ名（コンピュータの名称）を管理するための業務構成情報管理テーブル 1 0 2 を示す図である。本テーブルでは、業務システム名 6 0 1 と、複数の業務プログラムをくくるグループ名 6 0 2 と、業務を構成する業務プログラム名 6 0 3 と、業務プログラム

50

が稼動するコンピュータ名604を格納している。なお、グループ名602は無くても良い。本テーブルは、まず、ユーザ定義により作成されるが、業務プログラムの複写や移動、削除によって構成が変更された場合は、プログラム配備システムSによりその変更が反映される。

#### 【0037】

図10は、本実施形態のプログラム配備システムが管理する業務コンピュータ4の物理構成を保持する物理構成情報管理テーブル104を示す図である。本テーブルでは、コンピュータごと(コンピュータ1~4)に、プログラムレベル1(プログラムレベル701が1。以下同様)としてコンピュータに配備されているOS名703(プログラム種別702がOSである行の右側に記載した「OS1」など。以下同様)と、プログラムレベル2としてOSパッチ名704と、プログラムレベル3として業務プログラム名705を保持する。

10

本テーブルは、プログラムレベルごとに各コンピュータのエージェント処理部30から、コンピュータ内に配備されているOS名、OSパッチ名、プログラム名を収集し、各プログラムレベルに応じて格納し、作成する。

#### 【0038】

図11は、各業務プログラムの構成情報を管理するプログラム構成管理テーブル101を示す図である。本テーブルでは、業務プログラム名801の各業務プログラムに関して、業務プログラムの前提となるOS名802と、前提OSパッチ名803と、前提となる業務プログラムである前提プログラム名804と、連携先の業務プログラムである連携先プログラム名805を保持している。なお、前提とするOS名やOSパッチ、前提プログラム、連携先のプログラムがない場合は、それらが無くても良い。本テーブルはユーザ定義により作成される。

20

#### 【0039】

図12は、プログラム配備システムのメイン処理の流れを示した図である。

まず、その概要を説明すると、本実施形態のプログラム配備システムSは、オペレータによるGUI10からの業務プログラムの移動、複写、削除など(以下、「変更」ともいう。)の要求、もしくは、各要求イベントの発生があるまで、待機を続ける。そして、要求やイベントを受け取ると、それぞれの要求を実現するスクリプトの生成と実行を行い、要求によって業務システムの構成に変更が生じた場合は、業務システムの構成情報を業務コンピュータ4のエージェント処理部30から収集し、業務構成情報管理テーブル102および、物理構成情報管理テーブル104に反映する。なお、プログラム配備システムSは、終了要求もしくはイベントが発生すると、処理を終了する。

30

#### 【0040】

具体的には、管理コンピュータ11において、CPU2721は、記憶装置2723に格納されていたプログラム配備用プログラム6をメモリ2722上にロードすることで、以下の各処理を実行する(ステップ906~912の他図のフローチャートの処理も同様)。

まず、業務プログラムの変更の要求あるいは要求イベントの発生があると、ステップ900でフラグに1をセットする。次に、ステップ901では、フラグが1の場合(Yes)はステップ902へ進み、フラグが1以外の場合(No)は処理を終了する。

40

#### 【0041】

ステップ902では、GUI10から、業務プログラムの複写、移動や削除など、業務システムの構成の変更要求があった場合(Yes)はステップ903へ進み、そうでない場合(No)はステップ904へ進む。

#### 【0042】

ステップ903では、構成を変更した内容を業務構成情報管理テーブル102に反映し、また、プログラムの移動、複写、削除を行った業務コンピュータ4のエージェント処理部30から、コンピュータ内に配備されているプログラム情報を取得し、物理構成情報管理テーブル104を更新する。その後、ステップ901へ戻る。

50

## 【 0 0 4 3 】

ステップ 9 0 4、9 0 5では、それぞれ、業務プログラムの移動要求、複写要求があった場合 ( Y e s ) にステップ 9 0 6へ進み、そうでない場合 ( N o ) はステップ 9 1 0へ進む。

ステップ 9 0 6では、対象の業務プログラムの各種情報から、業務プログラムを配備可能なコンピュータを選択し ( 詳細は後記する図 1 7 の説明参照 )、ステップ 9 0 7へ進む。

## 【 0 0 4 4 】

ステップ 9 0 7では、配備先のコンピュータに対してプログラム構成差分情報管理テーブル 1 0 5を生成し ( 詳細は図 2 0 )、ステップ 9 0 8へ進む。

ステップ 9 0 8では、生成したプログラム構成差分情報管理テーブル 1 0 5に基づいて、要求を実現するスクリプトを生成し ( 詳細は図 3 0、図 3 2 )、ステップ 9 0 9へ進む。

ステップ 9 0 9では、生成したスクリプトを実行して ( 詳細は図 3 6 )、要求の実現を行い、ステップ 9 0 1へ戻る。

## 【 0 0 4 5 】

また、ステップ 9 1 0では、プログラムの削除要求があった場合 ( Y e s ) にステップ 9 1 1へ進み、そうでない場合 ( N o ) はステップ 9 1 3へ進む。

ステップ 9 1 1では、プログラムを削除するスクリプトを生成し ( 詳細は図 3 2 )、ステップ 9 1 2へ進む。

ステップ 9 1 2では、生成したスクリプトを実行して ( 詳細は図 3 6 )、要求の実現を行い、ステップ 9 0 1へ戻る。

ステップ 9 1 3では、プログラム配備システム S の終了要求があった場合 ( Y e s ) にステップ 9 1 4へ進み、そうでない場合 ( N o ) はステップ 9 0 1へ戻る。

ステップ 9 1 4では、フラグを 0 にしてステップ 9 0 1へ戻る。

## 【 0 0 4 6 】

次に、図 1 3 ~ 図 1 6を参照しながら、業務システムの構成を変更する場合における、G U I 1 0の表示の変化の例について説明する。なお、その場合のプログラム配備システム S での具体的な処理については、図 1 7以降で詳述する。

## 【 0 0 4 7 】

図 1 3は、本実施形態の監視コンピュータ 1 の G U I 1 0において表示される業務システムの構成を表示する画面の例を示す図である。

G U I 1 0では、業務システムの構成を階層的に表示した業務構成フィールド 1 1 0 0と、業務プログラムの連携を示す連携構成フィールド 1 1 0 1と、業務プログラムが稼動するコンピュータを示す物理構成フィールド 1 1 0 2がある。

## 【 0 0 4 8 】

業務構成フィールド 1 1 0 0は、ユーザが定義した業務構成情報管理テーブル 1 0 2 ( 図 9 参照 ) のデータを基に、業務システム名 6 0 1、グループ名 6 0 2、業務プログラム名 6 0 3を階層的に表示する。

連携構成フィールド 1 1 0 1は、ユーザが定義したプログラム構成管理テーブル 1 0 1 ( 図 1 1 参照 ) の連携先プログラム名 8 0 5のデータを基に、業務プログラムとその連携先のプログラムを関連付けて表示する。

物理構成フィールド 1 1 0 2は、ユーザが定義した業務構成情報管理テーブル 1 0 2 ( 図 9 参照 ) のコンピュータ名 6 0 4のデータを基に、業務プログラムと、稼動するコンピュータ名を関連付けて表示する。

## 【 0 0 4 9 】

図 1 3で示す例では、業務 4 ( 符号 1 0 0 1、以下「符号」を省略 ) が、Webシステム 1 0 1 0、APシステム 1 0 2 0および A P 1 0 ( 1 0 3 0 ) から構成されている。また、Webシステム 1 0 1 0は、業務プログラムである A P 8 ( 1 0 1 1 ) と A P 8 の前提プログラムである A P 7 ( 1 0 1 2 ) で構成され、コンピュータ 7 ( 1 0 4 1 ) 上で稼

10

20

30

40

50

動し、APシステム1020と連携している。APシステム1020は、業務プログラムAP9(1021)とAP9の前提プログラムであるAP7(1022)で構成され、コンピュータ8(1042)上で稼動し、AP10(1030)と連携している。AP10(1030)はコンピュータ9(1043)上で稼動している。

なお、GUI10から業務プログラムの複写、移動、削除の要求を出す場合は、オペレータが業務構成フィールド1100もしくは、連携構成フィールド1101で表示されている業務プログラムを指定し、複写、移動、削除を選択することができる。

#### 【0050】

図14は、業務プログラムの移動を実施した後の画面を示す図である。

業務プログラムの移動では、ユーザもしくはイベントにより、業務プログラムの移動要求が発生した場合に、本実施形態のプログラム配備システムSにおいて、管理コンピュータ11は、指定された業務プログラムを移動可能なコンピュータを選択し、移動用のスクリプトの生成と実行を行った後、業務システムの構成情報を更新し、GUI10は各フィールドの再表示を行う。

10

図14で示す例は、図13で示した業務4(1001)のAP9(1021)の移動を実施した後の画面である。AP9(1021)は、前提プログラムであるAP7(1022)と共にコンピュータ8(1042)からコンピュータ4(1044)へ移動したものとする。なお、移動前のコンピュータ8(1042)は、AP9(1021)とAP7(1022)が削除され、解放されている。

#### 【0051】

20

図15は、業務プログラムの複写を実施した後の画面を示す図である。

業務プログラムの複写では、ユーザもしくはイベントにより、業務プログラムの複写要求が発生した場合に、本実施形態のプログラム配備システムSにおいて、管理コンピュータ11は、指定された業務プログラムを複写可能なコンピュータを選択し、複写用のスクリプトの生成と実行を行った後、業務システムの構成情報を更新し、GUI10は各フィールドの再表示を行う。

#### 【0052】

図15で示す例は、図13で示した業務4(1001)のAP8(1011)の複写を実施した後の画面を示す図である。AP8(1011)は、前提プログラムであるAP7(1012)と共にコンピュータ4(1044)へ複写したものとする。

30

#### 【0053】

図16は、業務プログラムの削除を実施した後の画面を示す図である。

業務プログラムの削除では、ユーザもしくはイベントにより、業務プログラムの削除要求が発生した場合に、本実施形態のプログラム配備システムSにおいて、管理コンピュータ11は、指定された業務プログラムの削除用スクリプトの生成と実行を行った後、業務システムの構成情報を更新し、GUI10は各フィールドの再表示を行う。

図16で示す例は、図13で示した業務4(1001)のAP10(1030)の削除を実施した後の画面を示す図である。AP10(1030)は、コンピュータ9(1043)上から削除され、コンピュータ9(1043)は解放される。

#### 【0054】

40

図17は、業務プログラムの移動要求や複写要求が発生した場合に、管理コンピュータ11がプログラムの配備先のコンピュータを選択する処理内容を示した図である。

具体的には、管理コンピュータ11において、CPU2721は、記憶装置2723に格納されていたプログラム配備用プログラム6をメモリ2722上にロードすることで、以下の各処理を実行する(以下、他図における管理コンピュータ11の処理についても同様)。

なお、ここでは、業務プログラムは、複数のOSに対応したそれぞれの業務プログラムが、用意されているものとする。

#### 【0055】

まず、ステップ1401では、プログラムライブラリ管理テーブル21内で、複写、移

50

動要求で指定された業務プログラムのインストールイメージが存在するライブラリの対象 OS 304 を取得する。

ステップ 1402 では、業務構成情報管理テーブル 102 で業務プログラムが使用するコンピュータ名と物理構成情報管理テーブル 104 で保持するコンピュータ名を比較し、業務システムで使用していないコンピュータ（以下、空きコンピュータという。）名を取得し、ステップ 1401 で取得した指定業務プログラムの配備可能な OS（業務プログラムが複写、移動前に稼動していた OS や、業務プログラムのインストールイメージが対応する OS（プログラムライブラリ管理テーブル 21 に存在）など）と同じ OS を配備している空きコンピュータが存在する場合（Yes）はステップ 1403 へ進む。そうでない場合（No）には、ステップ 1406 へ進む。

10

【0056】

ステップ 1403 では、指定業務プログラムが元々稼動しているコンピュータと同じ OS の空きコンピュータが存在するかを調べ、存在する場合（Yes）はステップ 1405 へ進み、同じ OS の空きコンピュータの中から最も前提 OS パッチが多く配備されているコンピュータを選択する。

【0057】

存在しない場合（ステップ 1403 で No）は、ステップ 1404 へ進み、元々稼動しているコンピュータの OS とは異なるが、指定業務プログラムを配備可能な空きコンピュータの中で最も前提 OS パッチが多く配備されているコンピュータを選択する。

なお、前提 OS パッチが最も多く配備されているコンピュータを選択するのは、前提 OS パッチの配備数が多いほど前提 OS パッチをインストールする手間が省略できるからである。

20

【0058】

ステップ 1406 では、空きコンピュータの内、OS をステップ 1401 で取得した配備可能 OS に入れ替え可能かを OS 情報管理テーブル 103 で調べ、入れ替え可能なコンピュータが存在する場合（Yes）はステップ 1407 へ進む。そうでない場合（No）には、ステップ 1409 へ進む。

ステップ 1407 では、OS を入れ替えるため、入れ替え可能な空きコンピュータのうち任意のコンピュータを選択し、ステップ 1408 で、入れ替える対象の OS 名（OS 情報）を、記憶装置 2723（図 2 参照）に保存する。

30

【0059】

ステップ 1409 では、業務構成情報管理テーブル 102 で業務プログラムが使用するコンピュータ名と物理構成情報管理テーブル 104 で保持するコンピュータ名を比較し、業務プログラムで使用しているコンピュータ（以下使用中コンピュータ）を取得し、ステップ 1401 で取得した配備可能 OS と一致する使用中コンピュータが存在する場合（Yes）はステップ 1410 へ進む。そうでない場合（No）にはステップ 1413 へ進む。

【0060】

ステップ 1410 では、指定業務プログラムが稼動しているコンピュータと同じ OS の使用中コンピュータが存在するかを調べ、存在する場合（Yes）はステップ 1412 へ進み、同じ OS の使用中コンピュータの中から最も前提 OS パッチが多く配備されているコンピュータを選択する。存在しない場合（No）は、ステップ 1411 へ進み、稼動コンピュータの OS とは異なるが、指定業務プログラムを配備可能な使用中コンピュータの中で最も前提 OS パッチが多く配備されているコンピュータを選択する。

40

【0061】

ステップ 1413 では、指定業務プログラムを配備するコンピュータが存在しないことを、GUI 10 に表示する

以上の処理によって、指定業務プログラムの情報を基に、指定業務プログラムを配備するコンピュータを適切に選択することができる。

【0062】

50

図18は、配備対象プログラム管理テーブル106を示した図である。本テーブルは、複写や移動要求で指定された業務プログラム（ここでは、業務プログラム1515の「AP8」）を配備する場合に、必要なプログラム情報を保持するテーブルであり、管理コンピュータ11が、プログラム稼動最大構成テンプレート109（図5参照）を用いて生成する。本テーブルは、業務プログラムを配備する場合に必要な前提プログラムの情報をプログラム稼動最大構成テンプレート109にコピーすることによって、生成する。なお、プログラムの情報には、プログラムレベルも設定する。

**【0063】**

プログラムレベル1としてOS1511には、配備先コンピュータに配備するOS名を入力（管理コンピュータ11による自動入力。詳細は図20で後記。以下同様）し、プログラムレベル2としてOSパッチ1512に指定業務プログラムやその前提プログラムの前提OSパッチ名を入力する。また、プログラムレベル3には、業務プログラム1515に指定業務プログラム名を入力し、業務プログラムパッチ1516には指定業務プログラムのパッチ名を入力し、前提プログラム1513には、指定業務プログラムの前提プログラム名や前提プログラムの前提プログラム名を入力し、前提プログラムパッチ1514には、前提プログラム1513に入力されたプログラムのパッチ名を入力する。

10

**【0064】**

これらを換言すると、ここでは、「AP8」という業務プログラムを配備するためには、「AP8」のほかに、「OS1」、「Patch11」、「Patch12」、「AP7」、「AP7 Patch1」というプログラムが必要となり、それらを一覧表にしたものが配備対象プログラム管理テーブル106である。

20

なお、指定した業務プログラムによっては、テンプレートに対応するプログラム情報が存在しない場合がある。

**【0065】**

図19は、配備先コンピュータの物理構成情報（物理構成情報管理テーブル104から取得）と、配備対象プログラム管理テーブル106の差分情報をプログラムレベル1601ごとに保持するプログラム構成差分情報管理テーブル105を示した図である。なお、本テーブルの差分種別1602では、配備先コンピュータにのみ存在する情報を-、配備対象プログラム管理テーブル106にのみ存在するものを+として表現する。なお、差分がない場合は、差分種別1602に何も入力しない。また、プログラム名1603に対象のプログラム名を格納する。本テーブルは、管理コンピュータ11によって作成される。なお、本テーブルは、監視コンピュータ1のGUI10などに表示するようにしても良い。

30

**【0066】**

図20は、配備対象プログラム管理テーブル106と、プログラム構成差分情報管理テーブル105を作成する処理内容を示した図である。管理コンピュータ11は、まず、配備対象プログラム管理テーブル106を作成し、次に作成した配備対象プログラム管理テーブル106と物理構成情報管理テーブル104の情報からプログラム構成差分情報管理テーブル105を作成する。

**【0067】**

管理コンピュータ11は、ステップ1701において、配備先のコンピュータのOSを入れ替える必要があるか（図17の処理を行うときにステップ1408を通ったか）を調べ、OSを入れ替える必要がある場合（Yes）は、ステップ1703へ進み、配備対象プログラム管理テーブル106（図18参照）のOS1511にステップ1408で求めたOS名を入力（自動入力。以下同様）する。OSを入れ替える必要がない場合（ステップ1701でNo）は、ステップ1702へ進み、配備先コンピュータに配備されているOS名をOS1511に入力する。

40

**【0068】**

ステップ1704では、プログラム構成管理テーブル101から、入力したOS1511に対応した指定業務プログラムの情報を取得し、指定業務プログラムの前提OSパッチ

50

名 803 や前提プログラム名 804 を、配備対象プログラム管理テーブル 106 のそれぞれ対応する項目 (OS パッチ 1512、前提プログラム 1513) に入力する。

【0069】

ステップ 1705 では、プログラムライブラリ管理テーブル 21 (図 6 参照) を参照して、OS 1511 に対応した指定業務プログラムのパッチや、前提プログラム 1513 のパッチが存在するかを調べ、存在する場合 (Yes) はステップ 1706 に進み、該当するパッチ名を配備対象プログラム管理テーブル 106 の対応する項目 (前提プログラムパッチ 1514、業務プログラムパッチ 1516) に入力し、配備対象プログラム管理テーブル 106 を完成させる。また、存在しない場合 (No) はステップ 1707 へ進む。

【0070】

次に、ステップ 1707 では、プログラムレベルごとに配備対象プログラム管理テーブル 106 と物理構成情報管理テーブル 104 の配備先コンピュータの情報を比較し、プログラム構成差分情報管理テーブル 105 (図 19 参照) に未設定のプログラムレベルがある場合 (Yes) はステップ 1708 へ進み、全て設定した場合 (No) は処理を終了する。

ステップ 1708 では、未設定のプログラムレベルを選択し、ステップ 1709 へ進む。

【0071】

ステップ 1709 では、対象プログラムレベルごとに、配備先コンピュータ (物理構成情報管理テーブル 104 から取得した情報) にのみ存在するプログラム名を差分種別 - とし、そのプログラム名と差分種別をプログラム構成差分情報管理テーブル 105 に入力し、配備対象プログラム管理テーブル 106 にのみ存在するプログラムを差分種別 + とし、そのプログラム名と差分種別をプログラム構成差分情報管理テーブル 105 に入力する。双方に同じプログラムが配備されている場合は、差分種別 1602 には何も入力せず、プログラム名 1603 のみを入力する。その後、ステップ 1707 に戻る。

【0072】

図 21 は、業務プログラムの配備に使用するスクリプト部品の例を示す図である。スクリプト部品 1800 には、実行コンピュータ名 1801 の項目と、実行スクリプト 1802 の項目があり、実行スクリプト 1802 は実行コンピュータ名 1801 の業務コンピュータ 4 のエージェント処理部 30 へ転送され、実行するスクリプトである。

【0073】

また、これらの項目は、プログラム配備システム S 内で提供される変数を使用することができる。ここで、変数とは、後記するデータモデル 108 で定義されるデータモデルの各要素を示すアドレスのことである。この変数は、\$ と \$ の文字の間に、指定したデータモデルの要素名を先頭の業務名 2401 (図 34 で後記) から要素名ごとに「.」(ピリオド) を挟むことで表現され、要素と連結された要素名を指定するものである。

【0074】

変数によって取得できる値は、データモデルの要素を変数で順にたどった最後の要素に格納されている値である (実行コンピュータ名 1801 の例では「稼動コンピュータ名」)。変数を使用することにより、業務システムの構成が変更された場合も、スクリプト部品の定義を変えずに、変更後の値を格納したデータモデルから必要な値を取得し、実行することが可能である。

【0075】

なお、実行コンピュータ名 1801 には、変数を使用する場合に、コンピュータ名を指定する変数を使用する必要がある。

また、配備用スクリプト部品 403 (図 7 参照) には、配備対象のディスクイメージのプログラムライブラリ ID 301 (図 6 参照) を指定する必要がある。

このようにして、必要な変数を入力 (格納) することで、スクリプト部品が実際に使用可能な状態となる。

【0076】

10

20

30

40

50

図 2 2 は、プログラムライブラリ管理テーブル 2 1 ( 図 6 参照 ) で保持している OS のディスクイメージを配備するスクリプトテンプレートを示した図である。本テンプレートは、プログラムレベル 1 のテンプレートである。

本テンプレートは、スクリプトテンプレート 2 4 ( 図 1 参照 ) に格納されており、OS のディスクイメージを配備する OS 配備スクリプト部品 1 9 0 1 と、配備した OS の環境設定を行う環境設定スクリプト部品 1 9 0 3 と、エージェント処理部 3 0 の設定と起動を行うエージェント処理部設定スクリプト部品 1 9 0 5 と、スクリプト部品同士を連結する連結スクリプト部品 1 9 0 2 から構成されている。

【 0 0 7 7 】

図 2 2 中の矢印はスクリプト部品の実行順序を示しており、符号 1 9 0 1、1 9 0 2、1 9 0 3、1 9 0 2、1 9 0 5、1 9 0 2 の順で実行される。なお、OS 配備スクリプト部品 1 9 0 1 には、スクリプトライブラリ管理テーブル 2 3 ( 図 7 参照 ) の配備用スクリプト部品 4 0 3 が格納され、環境設定スクリプト部品 1 9 0 3 は、同様に環境設定用スクリプト部品 4 0 5 が格納される。

【 0 0 7 8 】

なお、連結スクリプト部品 1 9 0 2 は、コンピュータをリブートするか否かを定義するリブートフラグを持つ、本実施形態のプログラム配備システム S で制御するスクリプト部品であって、ユーザの定義は必要ない。

また、エージェント処理部設定スクリプト部品 1 9 0 5 は、OS のディスクコピーと共に配備されたエージェント処理部の設定と起動を行うものであって、ユーザの定義は必要ない。

【 0 0 7 9 】

なお、図 2 2 のスクリプトテンプレートは、各スクリプト部品の組み込みを予定しているだけで、実際にはまだコピーされていないものである。この点、図 2 3 ~ 図 2 6 のスクリプトテンプレートについても同様である。

【 0 0 8 0 】

図 2 3 は、プログラムライブラリ管理テーブル 2 1 で保持している OS パッチのインストールイメージを配備するスクリプトテンプレートを示した図である。本テンプレートは、プログラムレベル 2 のテンプレートである。

【 0 0 8 1 】

本テンプレートは、スクリプトテンプレート 2 4 に格納されており、OS パッチのディスクイメージを配備する OS パッチ配備スクリプト部品 1 9 1 1 と、スクリプト部品同士を連結する連結スクリプト部品 1 9 0 2 から構成されている。また、スクリプト部品の実行順序は、符号 1 9 1 1、1 9 0 2 の順である。なお、OS パッチ配備スクリプト部品 1 9 1 1 には、スクリプトライブラリ管理テーブル 2 3 ( 図 7 参照 ) の配備用スクリプト部品 4 0 3 が格納される。

【 0 0 8 2 】

図 2 4 は、プログラムライブラリ管理テーブル 2 1 ( 図 6 参照 ) で保持している業務プログラムのインストールイメージを配備するスクリプトテンプレートを示した図である。本テンプレートは、プログラムレベル 3 のテンプレートである。

【 0 0 8 3 】

本テンプレートは、スクリプトテンプレート 2 4 に格納されており、業務プログラムのディスクイメージを配備するプログラム配備スクリプト部品 1 9 2 1 と、配備したプログラムの環境設定を行う環境設定スクリプト部品 1 9 2 3 と、配備したプログラムの連携プログラムの設定を行う連携プログラム設定スクリプト部品 1 9 2 5 と、配備したプログラムを開始するプログラム開始スクリプト部品 1 9 2 7 と、スクリプト部品同士を連結する連結スクリプト部品 1 9 0 2 から構成されている。また、スクリプト部品の実行順序は、符号 1 9 2 1、1 9 0 2、1 9 2 3、1 9 0 2、1 9 2 5、1 9 0 2、1 9 2 7、1 9 0 2 の順である。

【 0 0 8 4 】

10

20

30

40

50

なお、プログラム配備スクリプト部品 1 9 2 1 にはスクリプトライブラリ管理テーブル 2 3 ( 図 7 参照 ) の配備用スクリプト部品 4 0 3 が格納され、環境設定スクリプト部品 1 9 2 3 には同様に環境設定用スクリプト部品 4 0 5 が格納され、連携プログラム設定スクリプト部品 1 9 2 5 には連携プログラム設定用スクリプト部品 4 0 6 が設定され、プログラム開始スクリプト部品 1 9 2 7 には、プログラム開始用スクリプト部品 4 0 8 が格納される。

【 0 0 8 5 】

図 2 5 は、プログラムライブラリ管理テーブル 2 1 で保持している業務プログラムのプログラムパッチを配備するスクリプトテンプレートを示した図である。本テンプレートは、プログラムレベル 3 のテンプレートである。

10

【 0 0 8 6 】

本テンプレートは、スクリプトテンプレート 2 4 ( 図 1 参照 ) に格納されており、パッチを当てるプログラムを停止させるプログラム停止スクリプト部品 1 9 3 1 と、プログラムパッチを配備するプログラムパッチ配備スクリプト部品 1 9 3 3 と、配備したプログラムを開始するプログラム開始スクリプト部品 1 9 2 7 と、スクリプト部品同士を連結する連結スクリプト部品 1 9 0 2 から構成されている。また、スクリプト部品の実行順序は、符号 1 9 3 1、1 9 0 2、1 9 3 3、1 9 0 2、1 9 2 7、1 9 0 2 の順である。

【 0 0 8 7 】

なお、プログラム停止スクリプト部品 1 9 3 1 にはスクリプトライブラリ管理テーブル 2 3 ( 図 7 参照 ) のプログラム停止用スクリプト部品 4 0 9 が格納され、プログラムパッチ配備スクリプト部品 1 9 3 3 には、配備用スクリプト部品 4 0 3 が格納され、プログラム開始スクリプト部品 1 9 2 7 にはプログラム開始用スクリプト部品 4 0 8 が格納される。

20

【 0 0 8 8 】

図 2 6 は、業務プログラムを削除するスクリプトテンプレートを示した図である。本テンプレートは、プログラムレベル 3 のテンプレートである。

本テンプレートは、スクリプトテンプレート 2 4 ( 図 1 参照 ) に格納されており、削除対象のプログラムを連携先に設定しているプログラムの連携先を変更 ( 解除 ) する連携元変更スクリプト部品 1 9 4 1 と、削除するプログラムを停止させるプログラム停止スクリプト部品 1 9 3 1 と、そのプログラムを削除するプログラム削除スクリプト部品 1 9 4 5 と、スクリプト部品同士を連結する連結スクリプト部品 1 9 0 2 から構成されている。

30

【 0 0 8 9 】

また、スクリプト部品の実行順序は、符号 1 9 4 1、1 9 0 2、1 9 3 1、1 9 0 2、1 9 4 5、1 9 0 2 の順である。なお、プログラム停止スクリプト部品 1 9 3 1 にはスクリプトライブラリ管理テーブル 2 3 ( 図 7 参照 ) のプログラム停止用スクリプト部品 4 0 9 が格納され、連携元変更スクリプト部品 1 9 4 1 には連携元プログラム設定変更用スクリプト部品 4 0 7 が格納され、プログラム削除スクリプト部品 1 9 4 5 には削除用スクリプト部品 4 0 4 が格納される。

【 0 0 9 0 】

図 2 7 は、業務プログラムを複写するために必要なスクリプトテンプレートを組み合わせたプログラム複写スクリプトシナリオを示した図である。

40

本シナリオは、スクリプトテンプレート 2 4 ( 図 1 参照 ) に格納されており、業務プログラムを複写するために、OS 配備スクリプトテンプレート 1 9 0 0、OS パッチ配備スクリプトテンプレート 1 9 1 0、プログラムを配備するプログラム配備スクリプトテンプレート 1 9 2 0、プログラムのパッチを配備するプログラムパッチ配備スクリプトテンプレート 1 9 3 0 を、この順で実行するものである。

【 0 0 9 1 】

これらを換言すると、本来複写したい業務プログラムを複写するためのテンプレートはプログラム配備スクリプトテンプレート 1 9 2 0 であり、それ以外の 3 つのスクリプトテンプレート ( 符号 1 9 0 0、1 9 1 0 および 1 9 3 0 ) はその補助的な役割を果たすもの

50

である。

なお、前提プログラムが1つ以上存在する場合は、プログラムパッチ配備スクリプトテンプレート1930を実行後、プログラム配備スクリプトテンプレート1920とプログラムパッチ配備スクリプトテンプレート1930を必要回数分繰り返し、必要プログラムを配備する。

【0092】

図28は、業務プログラムを削除するために必要なスクリプトテンプレートを組み合わせたプログラム削除スクリプトシナリオ1960を示した図である。

本シナリオは、スクリプトテンプレート24に格納されており、業務プログラムを削除するために、指定プログラムを削除するプログラム削除スクリプトテンプレート1940

10

【0093】

なお、指定業務プログラムの前提プログラムが1つ以上あり、その前提プログラムが他の業務プログラムなどで使われていない場合は、この業務プログラムの削除によって不要になるので、プログラム削除スクリプトテンプレート1940を必要回数分繰り返すことで、その前提プログラムも削除する。

【0094】

図29は、業務プログラムを移動するために必要なスクリプトシナリオを組み合わせたプログラム移動スクリプトシナリオ1970を示した図である。

本シナリオは、スクリプトテンプレート24に格納されており、業務プログラムを移動

20

【0095】

図30は、プログラム複写要求が発生した場合にステップ908(図12参照)で実施する複写スクリプトの生成処理の内容を示した図である。この場合、管理コンピュータ11が以下の処理を行う。

まず、ステップ2001では、複写スクリプトのひな形となるプログラム複写スクリプトシナリオ1950(図27参照)をスクリプトテンプレート24から取得し、これ以降、スクリプトシナリオの必要な項目に必要なスクリプト部品をセットしていく。

【0096】

30

ステップ2002では、すでに生成したプログラム構成差分情報管理テーブル105(図19参照)のプログラムレベル1の項目に、OSの追加を表す差分種別+があるか調べる。+の差分がある場合(Yes)はステップ2003へ進み、それ以外の場合(No)はステップ2010へ進む。

【0097】

ステップ2003では、プログラム構成差分情報管理テーブル105のプログラムレベル1の項目において、差分種別が+であるOS名を基に、プログラムライブラリ管理テーブル21(図6参照)から、該当するOSのプログラムライブラリID(符号301)とリポートフラグ(符号305)を取得する。

【0098】

40

また、取得したプログラムライブラリID(符号301)に該当するスクリプト部品を、スクリプトライブラリ管理テーブル23から取得する。取得したスクリプト部品は、プログラム複写スクリプトシナリオ1950(図27参照)のOS配備スクリプトテンプレート1900(図22参照)内で定義されるスクリプト部品と一致するスクリプト部品(OS配備スクリプト部品1901、環境設定スクリプト部品1903)をそれぞれのスクリプト部品としてコピーする。

また、OS配備スクリプト部品1901には、取得したプログラムライブラリIDをセットする。

【0099】

次に、ステップ2004では、ステップ2003で取得したリポートフラグをOS配備

50

スクリプト部品 1901 (図 22 参照) の直後の連結スクリプト部品 1902 へセットし、ステップ 2013 へ進む。

ステップ 2013 では、複写元のコンピュータのエージェント処理部 30 から設定情報を収集し、エージェント処理部設定スクリプト部品 1905 にコピーする。その後、ステップ 2005 へ進む。

#### 【0100】

ステップ 2010 では、OS の配備を行う必要がないため、プログラム複写スクリプトシナリオ 1950 (図 27 参照) から、OS 配備スクリプトテンプレート 1900 を削除し、ステップ 2005 へ進む。

このようにして、必要に応じて、自動的に、プログラム複写スクリプトシナリオ 1950 において、構成するスクリプトテンプレートの一部を削除することができる。

#### 【0101】

ステップ 2005 では、プログラム構成差分情報管理テーブル 105 のプログラムレベル 2 の項目に差分種別 + があるか調べる。+ の差分がある場合 (Yes) はステップ 2006 へ進み、それ以外の場合 (No) はステップ 2011 へ進む。

ステップ 2006 では、プログラム構成差分情報管理テーブル 105 (図 19 参照) のプログラムレベル 2 の項目において、差分種別が + である OS パッチ名を基に、プログラムライブラリ管理テーブル 21 (図 6 参照) から、該当する OS パッチのプログラムライブラリ ID とリポートフラグを取得する。

#### 【0102】

また、取得したプログラムライブラリ ID に該当するスクリプト部品を、スクリプトライブラリ管理テーブル 23 (図 7 参照) から取得する。取得したスクリプト部品は、プログラム複写スクリプトシナリオ 1950 (図 27 参照) の OS パッチ配備スクリプトテンプレート 1900 (図 22 参照) 内で定義されるスクリプト部品と一致するスクリプト部品 (OS パッチ配備スクリプト部品 1911) を、それぞれのスクリプト部品としてコピーする。また、配備スクリプト部品には取得したプログラムライブラリ ID をセットする。

#### 【0103】

さらに、ステップ 2007 では、ステップ 2006 で取得したリポートフラグを OS パッチ配備スクリプト部品 1911 (図 23 参照) の直後の連結スクリプト部品 1902 へセットする。

ステップ 2011 では、OS パッチの配備を行う必要がないため、プログラム複写スクリプトシナリオ 1950 (図 27 参照) から、OS パッチ配備スクリプトテンプレート 1910 を削除し、ステップ 2008 へ進む。

#### 【0104】

次に、ステップ 2008 では、プログラム構成差分情報管理テーブル 105 (図 19 参照) のプログラムレベル 3 の項目に差分種別 + があるかを調べる。+ の差分がある場合 (Yes) はステップ 2009 へ進み、それ以外の場合 (No) は 2012 へ進む。

#### 【0105】

ステップ 2009 では、図 31 で示すプログラム配備テンプレート設定処理を実施する。

ステップ 2012 では、プログラムの配備を行う必要がないため、プログラム複写スクリプトシナリオ 1950 (図 27 参照) から、プログラム配備スクリプトテンプレート 1920 とプログラムパッチ配備スクリプトテンプレート 1930 を削除する。

以上のように、必要に応じてスクリプトテンプレートを追加、削除し、スクリプト部品をコピーして生成した複写スクリプトを、スクリプト 25 に格納する。

#### 【0106】

図 31 は、図 30 のステップ 2009 で実施するプログラム配備スクリプトテンプレート設定処理の内容を示す図である。ステップ 2101 では、プログラム構成差分情報管理テーブル 105 (図 19 参照) のプログラムレベル 3 の項目において、未設定の差分種別

10

20

30

40

50

+のプログラムが存在する場合 (Yes) はステップ 2 1 0 2 へ進み、それ以外の場合 (No) は処理を終了する。

【 0 1 0 7 】

ステップ 2 1 0 2 では、プログラム構成差分情報管理テーブル 1 0 5 の未設定の差分種別 + のプログラム名を取得する。

次にステップ 2 1 0 3 では、ステップ 2 1 0 2 で取得したプログラム名の前提プログラムが未設定かを調べる。前提プログラムが未設定の場合 (Yes) は、ステップ 2 1 0 2 へ進み、設定済みの場合 (No) はステップ 2 1 0 4 へ進む。

【 0 1 0 8 】

ステップ 2 1 0 4 では、プログラム名がパッチ名かを調べる。パッチ名の場合 (Yes) はステップ 2 1 0 5 へ進み、そうでない場合 (No) はステップ 2 1 0 7 へ進む。

ステップ 2 1 0 5 では、パッチの対象となるプログラムが未設定かを調べる。未設定の場合 (Yes) はステップ 2 1 0 1 へ進み、設定済みの場合 (No) はステップ 2 1 0 6 へ進む。

【 0 1 0 9 】

ステップ 2 1 0 6 では、プログラムのパッチを配備するプログラムパッチ配備スクリプトテンプレート 1 9 3 0 (図 2 5 参照) に、対象プログラムパッチのスクリプト部品をコピーする。なお、スクリプトテンプレートに対象プログラムのスクリプト部品をコピーするこのステップ 2 1 0 6 の処理は、図 3 3 で詳しく説明する。その後、ステップ 2 1 0 1 へ戻る。

【 0 1 1 0 】

ステップ 2 1 0 7 では、プログラムの配備を行うプログラム配備スクリプトテンプレート 1 9 2 0 (図 2 4 参照) に対象プログラムのスクリプト部品をコピーする。なお、プログラム配備スクリプトテンプレート 1 9 2 0 が設定済みの場合は、最後尾のテンプレートに新たにプログラム配備スクリプトテンプレート 1 9 2 0 を連結してスクリプト部品のコピーを行う。

その後、ステップ 2 1 0 1 へ戻る。

【 0 1 1 1 】

図 3 2 は、プログラム削除要求が発生した場合にステップ 9 1 1 (図 1 2 参照) で実施する削除用スクリプトの生成処理の内容を示した図である。

まず、ステップ 2 2 0 1 では、削除用スクリプトのひな形となるプログラム削除スクリプトシナリオ 1 9 6 0 (図 2 8 参照) を取得し、これ以降、このスクリプトシナリオの必要な項目に必要なスクリプト部品をセットしていく。

【 0 1 1 2 】

まず、ステップ 2 2 0 2 では、指定業務プログラムを削除するプログラム削除スクリプトテンプレート 1 9 4 0 (図 2 6 参照) に削除対象プログラムのスクリプト部品をコピーし、ステップ 2 2 0 3 へ進む。なお、スクリプトテンプレートに対象プログラムのスクリプト部品をコピーするこのステップ 2 2 0 2 の処理は、図 3 3 で詳しく説明する。

【 0 1 1 3 】

ステップ 2 2 0 3 では、テンプレートにスクリプト部品をコピーしたプログラム名を、設定済みプログラム名とする。その後、ステップ 2 2 0 4 へ進む。

ステップ 2 2 0 4 では、設定済みプログラムに前提プログラムがあるか否かをプログラム構成管理テーブル 1 0 1 から取得 (判断) し、前提プログラムが存在する場合 (Yes) はステップ 2 2 0 5 へ進み、そうでない場合 (No) は処理を終了する。

【 0 1 1 4 】

ステップ 2 2 0 5 では、最後尾のテンプレートに新たにプログラム削除スクリプトテンプレート 1 9 4 0 (図 2 6 参照) を追加し、前提プログラム名のスクリプト部品をコピーする (図 3 3 で説明)。その後ステップ 2 2 0 3 へ戻る。

以上のように削除スクリプトシナリオのスクリプトテンプレートを追加し、スクリプト部品をコピーして生成した削除スクリプトを、スクリプト 2 5 に格納する。

10

20

30

40

50

## 【 0 1 1 5 】

図 3 3 は、スクリプトテンプレートに、該当するプログラムのスクリプト部品をコピーする処理内容を示した図である。

ステップ 2 3 0 1 では、該当するプログラム名と該当プログラムが配備や削除を行う対象の OS 名を基に、プログラムライブラリ管理テーブル 2 1 ( 図 6 参照 ) から該当するプログラムのプログラムライブラリ ID とリポートフラグを取得する。また、取得したプログラムライブラリ ID を基に、スクリプトライブラリ管理テーブル 2 3 ( 図 7 参照 ) から該当するプログラムで使用するスクリプト部品を取得する。

ステップ 2 3 0 2 では、スクリプトテンプレートに未設定のスクリプト部品がある場合 ( Yes ) はステップ 2 3 0 3 へ進み、そうでない場合 ( No ) は処理を終了する。

10

## 【 0 1 1 6 】

ステップ 2 3 0 3 では、スクリプトテンプレートで次の未設定スクリプト部品を取得する。

次に、ステップ 2 3 0 4 では、未設定スクリプト部品が配備用スクリプト部品かを調べ、配備用スクリプト部品の場合 ( Yes ) はステップ 2 3 0 5 へ進み、そうでない場合 ( No ) はステップ 2 3 0 8 へ進む。

ステップ 2 3 0 5 では、プログラムのスクリプト部品から配備用スクリプト部品を取得し、テンプレートの未設定スクリプト部品へコピーし、ステップ 2 3 0 6 へ進む。また、配備スクリプト部品には、取得したプログラムライブラリ ID をセットする。

## 【 0 1 1 7 】

20

ステップ 2 3 0 6 では、テンプレートから次の未設定スクリプトである連結スクリプトを取得し、ステップ 2 3 0 7 では、プログラムのリポートフラグをプログラムライブラリ管理テーブル 2 1 から取得し、連結スクリプトにセットする。その後、ステップ 2 3 0 2 へ戻る。

ステップ 2 3 0 8 では、未設定スクリプト部品と一致するプログラムのスクリプト部品が存在するかを調べ、一致するスクリプト部品が存在する場合 ( Yes ) はステップ 2 3 0 9 へ進み、そうでない場合 ( No ) はステップ 2 3 1 2 へ進む。

## 【 0 1 1 8 】

次にステップ 2 3 0 9 では、プログラムのスクリプト部品から一致するスクリプト部品を取得し、テンプレートの未設定スクリプト部品へコピーし、ステップ 2 3 1 0 へ進む。

30

ステップ 2 3 1 0 では、テンプレートから次の未設定スクリプトである連結スクリプトを取得し、ステップ 2 3 1 1 では、何もセットしない。もしくは、未設定スクリプトである連結スクリプトを削除しても良い。その後、ステップ 2 3 0 2 へ戻る。

## 【 0 1 1 9 】

ステップ 2 3 1 2 では、未設定のスクリプト部品と一致するスクリプト部品が存在しないため、何もセットしない。もしくは、未設定のスクリプト部品を削除しても良い。

同様に、ステップ 2 3 1 3 で、テンプレートから次の未設定スクリプトである連結スクリプトを取得し、何もセットしない。もしくは、未設定スクリプトである連結スクリプトを削除しても良い。その後、ステップ 2 3 0 2 へ戻る。

## 【 0 1 2 0 】

40

図 3 4 は、スクリプト部品に使用する変数のデータ構造を表した、パラメタ用のデータモデルテンプレート 1 0 7 を示す図である。本テンプレートを使用して、業務プログラムごとにデータモデルを生成する。

データモデルテンプレート 1 0 7 では、業務プログラム名 2 4 0 2 が、所属する業務名 2 4 0 1 と、グループ名 2 4 1 4 に連結される。ただし、グループ名 2 4 1 4 は無くても良い。

## 【 0 1 2 1 】

また、業務プログラム名 2 4 0 2 は、稼動コンピュータ名 2 4 0 3 と、前提 OS パッチ名 2 4 0 4 と、前提プログラム名 2 4 0 5 と、関連プログラム名 2 4 0 6 と、配備先コンピュータ名 2 4 0 7 と連結する。ただし、前提 OS パッチ名 2 4 0 4 と、前提プログラム

50

名 2 4 0 5 と、関連プログラム名 2 4 0 6 は、無くても良い。また、削除対象の業務プログラムのデータモデルは、配備先コンピュータ名 2 4 0 7 が存在しない。

【 0 1 2 2 】

稼動コンピュータ名 2 4 0 3 には、OS 名 2 4 0 8 と、OS パッチ名 2 4 0 9 と、業務プログラム名 2 4 1 0 が連結される。ただし、OS パッチ名 2 4 0 9 と、業務プログラム名 2 4 1 0 は無くても良い。

なお、生成したデータモデルは、業務名 2 4 0 1 と、グループ名 2 4 1 4 と、業務プログラム名 2 4 0 2 により、プログラム配備システム S 内で一意に定まるものである。

【 0 1 2 3 】

また、生成したデータモデルにおいて、前提プログラム名 2 4 0 5 と、関連プログラム名 2 4 0 6 は、対象のプログラムのデータモデルと連結することが可能である。業務プログラムに前提プログラムが存在する場合は、データモデルの前提プログラム名 2 4 0 5 は、前提プログラムのデータモデルの業務プログラム名 2 4 0 2 と連結する。同様に業務プログラムに関連プログラムが存在する場合は、関連プログラム名 2 4 0 6 は、関連プログラムのデータモデルの業務プログラム名 2 4 0 2 と連結する。

10

【 0 1 2 4 】

なお、1つの業務でも同じ業務プログラムが異なるコンピュータで複数稼動する場合があるため、前提プログラムのデータモデルは、同じ業務名 2 4 0 1 の同じグループ名 2 4 1 4 を持ち、前提プログラム名と一致する業務プログラム名 2 4 0 2 のデータモデルとする。また、関連プログラムのデータモデルは、同じ業務名 2 4 0 1 を持ち、関連プログラム名と一致する業務プログラムのデータモデルとする。

20

【 0 1 2 5 】

以上のように、業務プログラムごとに生成されるデータモデルを、そのデータモデルの前提プログラム、関連プログラムに対応したデータモデルと連結することで、スクリプト部品の変数に割り当てるパラメタとして業務プログラムに関係するプログラムのデータの取得が可能となる。例えば、業務プログラムの関連プログラムが稼動するコンピュータ上で、実行スクリプト 1 8 0 2 ( 図 2 1 参照 ) を実行したい場合は、実行コンピュータ名 1 8 0 1 の変数として、「\$ 業務名 . グループ名 . プログラム名 . 関連プログラム名 . 稼動コンピュータ名 \$ 」と定義することで実現できる。

【 0 1 2 6 】

図 3 5 は、データモデルテンプレート 1 0 7 に業務プログラムのデータをコピーし、データモデル 1 0 8 を生成する処理内容を示した図である。データモデルは、管理コンピュータ 1 1 が、業務プログラムごとに、複写、移動、削除要求で指定された業務プログラムが存在する業務内の全ての業務プログラムで生成する。

30

【 0 1 2 7 】

まず、ステップ 2 5 0 1 では、対象業務内にデータモデルを生成していない業務プログラムが存在するかを調べ、存在する場合 ( Y e s ) はステップ 2 5 0 2 へ進み、そうでない場合 ( N o ) はステップ 2 5 0 7 へ進む。

ステップ 2 5 0 2 で、データモデルを生成していない業務プログラム名を取得し、ステップ 2 5 0 3 では、対象業務の取得した業務プログラム名に該当するデータを業務構成情報管理テーブル 1 0 2 ( 図 9 参照 ) から取得する。

40

また、ステップ 2 5 0 4 では、プログラム構成管理テーブル 1 0 1 ( 図 1 1 参照 ) から該当する業務プログラムのデータを取得する。

【 0 1 2 8 】

ステップ 2 5 0 5 では、ステップ 2 5 0 3 で取得した業務構成データから、稼動コンピュータ名を取得し、物理構成情報管理テーブル 1 0 4 ( 図 1 0 参照 ) から、該当するコンピュータのデータを取得する。

その後、ステップ 2 5 0 6 でデータモデルのテンプレートに、該当するデータをコピーする。まず、業務構成情報管理テーブル 1 0 2 ( 図 9 参照 ) から取得した、業務システム名 6 0 1 を業務名 2 4 0 1 へ、グループ名 6 0 2 をグループ名 2 4 1 4 へ、業務プログラ

50

ム名 6 0 3 を業務プログラム名 2 4 0 2 へ、コンピュータ名 6 0 4 を稼動コンピュータ名 2 4 0 3 へコピーする。

【 0 1 2 9 】

さらに、プログラム構成管理テーブル 1 0 1 から取得した前提 OS パッチ名 8 0 3 を前提 OS パッチ名 2 4 0 4 へ、前提プログラム名 8 0 4 を前提プログラム名 2 4 0 5 へ、連携先プログラム名 8 0 5 を関連プログラム名 2 4 0 6 へコピーする。次に、物理構成情報管理テーブル 1 0 4 から取得したコンピュータの OS 7 0 3 を OS 名 2 4 0 8、2 4 1 1 へ、OS パッチ名 7 0 4 を OS パッチ名 2 4 0 9、2 4 1 2 へ、業務プログラム 7 0 5 を業務プログラム名 2 4 1 0、2 4 1 3 へコピーする。

以上のようにデータをテンプレートにコピーすることで、業務プログラムのデータモデルを完成させる。なお、テンプレートに対応するデータがない場合は、何もコピーしないか、あるいは、削除しても良い。

10

【 0 1 3 0 】

次に、生成した業務プログラムのデータモデルにおいて、前提プログラム名や連携プログラム名に対応したデータモデルを連結する。まず、ステップ 2 5 0 7 で生成した全てのデータモデルで連結作業を行った場合 ( Y e s ) はステップ 2 5 1 3 へ進み、そうでない場合 ( N o ) はステップ 2 5 0 8 へ進む。

ステップ 2 5 0 8 では、連結作業を行っていないデータモデルを取得する。

ステップ 2 5 0 9 では、データモデルに前提プログラム名がセットされているかを調べる。セットされている場合 ( Y e s ) はステップ 2 5 1 0 へ進み、そうでない場合 ( N o ) はステップ 2 5 1 1 へ進む。

20

【 0 1 3 1 】

ステップ 2 5 1 0 では、同一業務名の同一グループ名のデータモデルで、前提プログラム名に該当するデータモデルを取得し、前提プログラム名 2 4 0 5 に該当データモデルの業務プログラム名 2 4 0 2 を連結し、ステップ 2 5 1 1 へ進む。

ステップ 2 5 1 1 では、データモデルに関連プログラム名がセットされているかを調べる。セットされている場合 ( Y e s ) はステップ 2 5 1 2 へ進み、そうでない場合 ( N o ) はステップ 2 5 0 7 へ戻る。

【 0 1 3 2 】

ステップ 2 5 1 2 では、同一業務名のデータモデルで、前提プログラム名に該当するデータモデルを取得し、関連プログラム名 2 4 0 6 に該当データモデルの業務プログラム名 2 4 0 2 を連結し、ステップ 2 5 0 7 へ進む。

30

また、ステップ 2 5 1 3 では、複写、移動要求で指定された業務プログラムのデータモデルにおいては、配備先コンピュータが選択されているため、物理構成情報管理テーブル 1 0 4 から該当する配備先コンピュータのデータを取得し、指定業務プログラムのデータモデルに配備先コンピュータ名 2 4 0 7、OS 名 2 4 1 1、OS パッチ名 2 4 1 2、業務プログラム名 2 4 1 3 をコピーする。

【 0 1 3 3 】

図 3 6 は、生成した複写、移動、削除のスクリプト 2 5 を実行する処理内容を示す図である。スクリプト 2 5 は、要求に応じたスクリプトシナリオから、スクリプトテンプレートの追加、削除を行い、スクリプトテンプレートにそれぞれ対応するスクリプト部品をコピーして完成されたものである。なお、スクリプト 2 5 の実行は、スクリプトの先頭から、コピーされたスクリプト部品を 1 つずつ取り出し、行われる。

40

【 0 1 3 4 】

ステップ 2 6 0 1 では、スクリプトに次のスクリプト部品が存在するか調べる。存在する場合 ( Y e s ) はステップ 2 6 0 2 へ進み、そうでない場合 ( N o ) はスクリプトの実行を終了する。

ステップ 2 6 0 2 では、スクリプトから次のスクリプト部品を取得する。

次に、ステップ 2 6 0 3 では、そのスクリプト部品が連結スクリプト部品の場合 ( Y e s ) はステップ 2 6 0 4 へ進み、そうでない場合 ( N o ) はステップ 2 6 0 8 へ進む。

50

ステップ2604では、連結スクリプト部品にセットされたリポートフラグを調べ、リポートフラグがONの場合(Yes)はステップ2605へ進み、OFFもしくは何もセットされていない場合(No)はステップ2612へ進む。89

【0135】

ステップ2605では、前にスクリプト部品を実行したコンピュータのリポートを行う。該当コンピュータのエージェント処理部30へリポート要求を送信し、ステップ2606へ進む。

ステップ2606では、前にスクリプト部品を実行したコンピュータのエージェント処理部30から起動完了通知を受信するまで処理を待ち、受信後ステップ2601へ戻る。

【0136】

ステップ2612では、前にスクリプト部品を実行したコンピュータのエージェント処理部30にリポート要求が出ていないか確認する。リポート要求が出ている場合(Yes)はステップ2605へ進み、出ていない場合(No)はステップ2607へ進む。

ステップ2607では、前にスクリプト部品を実行したコンピュータがリポートしているかを確認するため、該当コンピュータのエージェント処理部30へ起動しているかを確認する。自動的にリポートされたが、まだコンピュータが起動を完了していない場合(Yes)はステップ2606へ進み、リポートが完了するのを待ち、起動を完了していた場合(No)はステップ2601へ進む。

【0137】

次のスクリプト部品が連結スクリプト部品ではない場合(ステップ2603でNo)、ステップ2608で、スクリプト部品が未設定かを調べる。設定済みの場合(No)はステップ2609へ進み、未設定の場合(Yes)はステップ2601へ戻る。

ステップ2609で、スクリプト部品を実行するコンピュータである実行コンピュータ名1801(図21参照)と、実行スクリプト1802に使用されている変数で指定されたデータをデータモデル108(図4、図34参照)から取得し、パラメタとして変数に代入する。

【0138】

次に、ステップ2610では、実行コンピュータ名1801のエージェント処理部30に実行スクリプト1802を転送する。

その後、ステップ2611では、エージェント処理部30から実行スクリプト1802を実行した結果を受信するまで待ち、受信後、ステップ2601へ戻る。

【0139】

このように、本実施形態のプログラム配備システムSによれば、業務プログラムの複写、移動、削除要求に対し、指定された業務プログラムと業務コンピュータの状況に合わせて、スクリプトシナリオからスクリプトテンプレートの追加や削除を行う。また、そのスクリプトシナリオにおいて、必要なスクリプト部品の選択および設定と、パラメタの値の割付けを行うことで、スクリプトを生成する。そして、そのスクリプトを実行することで、業務プログラムの配備、削除を、業務コンピュータを意識することなく、容易に行うことができる。

【0140】

以下、本実施形態のプログラム配備システムSの応用例について、図37～図41を参照しながら説明する。

図37は、業務の複製や、移動、削除のイベントに対して対象業務の構成差分情報を保持する業務構成差分情報管理テーブル3000を示す図である。この業務構成差分情報管理テーブル3000は、プログラム構成差分情報管理テーブル105(図19参照)を拡張したものであり、たとえば、プログラム構成差分情報管理テーブル105とともにシステム構成情報22(図4参照)に記憶するようにしておけば良い。

【0141】

本テーブルには、業務レベル3001と、差分種別3002と、業務名3003と、業務プログラムまたはグループ名3004を保持する。差分種別3002は、業務を複製す

10

20

30

40

50

る場合には+、削除する場合には-として表現する。

【0142】

なお、業務の移動は、対象業務の複写を実施した後に業務の削除を実施することで実現する。また、業務レベル3001は、複製、削除を行う業務のプログラムにおいて、連携先プログラムによる依存度を表しており、例えば、連携先プログラムが無い場合はレベル1、ある場合はレベル2、連携先プログラムにも連携先プログラムがある場合にはレベル3とする。なお、連携先プログラムにさらに連携先プログラムがあれば、レベル4, 5・・・とすれば良い。このレベルに基づき、プログラムまたはグループを複写する場合は昇順に、削除の場合は降順に実施する。つまり、プログラム同士の前提状況や関連状況から、複写時は優先順位の高いものから複写し、削除時は優先順位の低いものから削除することで、処理にエラーが発生しないようにするものである。

10

【0143】

図38は、図37の業務プログラムまたはグループ名3004がグループ名の場合に、グループ構成差分情報管理テーブル3100を示す図である。このグループ構成差分情報管理テーブル3100は、業務構成差分情報管理テーブル3000と同様、プログラム構成差分情報管理テーブル105(図19参照)を拡張したものであり、たとえば、プログラム構成差分情報管理テーブル105とともにシステム構成情報22(図4参照)に記憶するようにしておけば良い。

【0144】

本テーブルには、グループレベル3101と、差分種別3102と、グループ名3103と、業務プログラム名3104を保持する。差分種別3102は、業務を複製する場合には+、削除する場合には-として表現する。

20

【0145】

また、グループレベル3101は、複製、削除を行うグループのプログラムにおいて、プログラムレベル3の前提プログラムの依存度を表しており、例えば、前提プログラムが無い場合はレベル1、ある場合はレベル2、前提プログラムにも前提プログラムがある場合にはレベル3とする。このレベルに基づき、プログラムを複写する場合は昇順に、削除の場合は降順に実施する。これも、図37の場合と同様、プログラム同士の前提状況や関連状況から、複写時は優先順位の高いものから複写し、削除時は優先順位の低いものから削除することで、処理にエラーが発生しないようにするものである。

30

【0146】

図39は、管理コンピュータ11が業務構成差分情報管理テーブル3000を作成する処理内容を示した図である。まず、ステップ3201では、業務を削除する場合(Yes)はステップ3203へ移動し、複写する場合(No)はステップ3202へ移動する。

ステップ3202では、業務構成情報管理テーブル102(図9参照)で複写対象の業務構成情報をコピーし、別名称をつける。例えば、業務1をコピーした場合に業務1-1とする。このとき、業務1-1は未生成のため、コンピュータ名604を空欄にし、複写後に反映されるものとする。

【0147】

次にステップ3203では、業務構成情報管理テーブル102から対象業務のグループ名(グループ名が無い場合はプログラム名)を業務構成差分情報管理テーブル3000のレベル1へコピーする。このとき、業務を削除する場合は、差分種別3002を-を、複写の場合は+をセットする。次に、ステップ3204では、対象レベルに1をセットする。

40

【0148】

ステップ3205では、業務構成差分情報管理テーブル3000の全てのレベルで設定が終わった場合(Yes)は処理を終了し、それ以外の場合(No)はステップ3206へ進む。

ステップ3206では、業務構成差分情報管理テーブル3000において、対象レベルの全てのグループ(もしくはプログラム)で設定が完了した場合(Yes)はステップ3

50

210へ進み、それ以外の場合（No）はステップ3207へ進む。

ステップ3207では、対象レベルにおいて未設定のグループ（もしくはプログラム）を取得し、ステップ3208へ進む。

【0149】

ステップ3208では、対象グループ内のプログラム（グループではない場合はそのプログラム）において連携先プログラム名をプログラム構成管理テーブル101の805から取得し、連携先プログラム名が対象レベル内に存在する場合（Yes）はステップ3209へ進み、それ以外の場合（No）はステップ3206へ進む。

ステップ3209では、グループ（もしくはプログラム）を対象レベル+1のレベルに設定し、ステップ3206へ進む。

10

ステップ3210では、対象レベルに1を足し、ステップ3205へ進む。

【0150】

図40は、管理コンピュータ11がグループ構成差分情報管理テーブル3100を作成する処理内容を示した図である。まず、ステップ3301では、業務構成情報管理テーブル102から対象グループのプログラム名をグループ構成差分情報管理テーブル3100のレベル1へコピーする。このとき、業務を削除する場合は、差分種別3302を-を、複写の場合は+をセットする。

【0151】

次にステップ3302では、対象レベルに1をセットする。

ステップ3303では、業務構成差分情報管理テーブル3000の全てのレベルで設定が終わった場合（Yes）は処理を終了し、それ以外の場合（No）はステップ3304へ進む。

20

ステップ3304では、グループ構成差分情報管理テーブル3100において、対象レベルの全てのプログラムで設定が完了した場合（Yes）はステップ3308へ進み、それ以外の場合（No）はステップ3305へ進む。

ステップ3305では、対象レベルにおいて未設定のプログラムを取得し、ステップ3306へ進む。

【0152】

ステップ3306では、対象プログラムにおいて前提プログラム名をプログラム構成管理テーブル101（図11参照）の前提プログラム名804から取得し、前提プログラム名が対象レベル内に存在する場合（Yes）はステップ3307へ進み、それ以外の場合（No）はステップ3304へ進む。

30

ステップ3307では、プログラムを対象レベル+1のレベルに設定し、ステップ3304へ進む。

ステップ3308では、対象レベルに1を足し、ステップ3303へ進む。

【0153】

図41は、管理コンピュータ11が業務の複写、移動、削除を実施する処理内容を示した図である。

なお、この処理は、業務プログラムごとではなく、業務ごとに複写などの変更を実施するものである。この場合、管理コンピュータ11は、まず、図12の処理を開始し、図12におけるステップ902のNoの直後に、業務ごとの変更があるか否かを判断し、業務ごとの変更があると判断した場合に、図41の処理を行うことになる。そして、図41のステップ3408において図12のステップ906～909を実行し、図41のステップ3413において図12のステップ911と912を実行することになる。

40

また、業務ごとの変更があるか否かの判断は、図12において、ステップ902のNoの直後でなくても、そのほかに、ステップ910のNoの直後、あるいは、ステップ913のNoの直後などであってもよい。

【0154】

図41について説明すると、まず、ステップ3401では、複写、移動、削除に応じた業務構成差分情報管理テーブル3000を生成する（図39参照）。

50

次にステップ3402では、生成した業務構成差分情報管理テーブル3000の業務プログラム名またはグループ名3004にグループ名が設定されている場合(Yes)、グループ数分のグループ構成差分情報管理テーブル3100をステップ3403で生成する(図40参照)。

【0155】

次に、ステップ3404では、対象レベルを1とし、ステップ3405へ進む。

ステップ3405では、生成した業務構成差分情報管理テーブル3000の全てのレベルで差分種別3002が+の要素(グループまたはプログラム)の複写を実施した場合(Yes)はステップ3410へ進み、それ以外の場合(No)はステップ3406へ進む。

10

【0156】

ステップ3406では、対象レベルの全ての要素について複写を実施した場合(Yes)はステップ3409へ進み、それ以外の場合(No)はステップ3407へ進む。

ステップ3407では、対象レベルの次の要素を取得し、ステップ3408へ進む。

【0157】

ステップ3408では、要素の複写を実施する。このとき、要素がプログラムの場合はプログラムの複写イベントを出し、要素がグループの場合は、該当するグループ構成差分情報管理テーブル3100において、グループレベル3101の昇順に業務プログラム名3104の複写イベントを出す。

【0158】

ステップ3409では、対象レベルに1を足し、ステップ3405へ進む。

次にステップ3410では、生成した業務構成差分情報管理テーブル3000の全てのレベルで差分種別3002が-の要素(グループまたはプログラム)の削除を実施した場合(Yes)は処理を終了し、それ以外の場合(No)はステップ3411へ進む。

ステップ3411では、対象レベルの全ての要素について削除を実施した場合(Yes)はステップ3414へ進み、それ以外の場合(No)は3412へ進む。

20

【0159】

ステップ3412では、対象レベルの次の要素を取得し、ステップ3413へ進む。

ステップ3413では、要素の削除を実施する。このとき、要素がプログラムの場合はプログラムの削除イベントを出し、要素がグループの場合は、該当するグループ構成差分情報管理テーブル3100において、グループレベル3101の降順に業務プログラム名3104の削除イベントを出す。

30

ステップ3414では、対象レベルから1を引き、ステップ3410へ進む。

【0160】

このようにして、業務ごとに複写、移動、削除などの変更を行う場合でも、業務プログラムの変更の順序決定や各種設定などを自動で行うことができ、業務システムの構成を容易に変更することができる。

【0161】

また、本実施形態のプログラム配備システムSによる業務システム構成変更方法は、コンピュータ及びプログラムによって実現することができ、そのプログラムをコンピュータによる読み取り可能な記録媒体に記録することで、その記録媒体によって提供することが可能である。また、そのプログラムを、ネットワークを通して提供することも可能である。

40

【0162】

以上で実施形態の説明を終えるが、本発明の態様はこれらに限定されるものではない。

たとえば、本実施形態では、監視コンピュータに表示画面(GUI)を設けるようにしたが、監視コンピュータを用いず、管理コンピュータにGUIなどの表示画面を設けるようにしても良い。

【0163】

また、業務プログラムを配備可能なコンピュータが複数あった場合、セキュリティレベ

50

ルがより高いOSを備えたコンピュータに、その業務プログラムを配備するようにしても良い。

その他、具体的な構成について、本発明の趣旨を逸脱しない範囲で適宜変更が可能である。

【図面の簡単な説明】

【0164】

【図1】本実施形態のプログラム配備システムの概要を示した図である。

【図2】本実施形態のプログラム配備システムのハードウェア構成を示した図である。

【図3】本実施形態のエージェントのハードウェア構成を示した図である。

【図4】本実施形態のシステム構成情報を示した図である。

10

【図5】本実施形態のプログラム稼動最大構成テンプレートを示した図である。

【図6】本実施形態のプログラムライブラリ管理テーブルを示した図である。

【図7】本実施形態のスク립トライブラリ管理テーブルを示した図である。

【図8】本実施形態のOS情報管理テーブルを示した図である。

【図9】本実施形態の業務構成管理テーブルを示した図である。

【図10】本実施形態の物理構成情報管理テーブルを示した図である。

【図11】本実施形態のプログラム構成管理テーブルを示した図である。

【図12】本実施形態のメイン処理フローを示した図である。

【図13】本実施形態の初期画面を示した図である。

【図14】本実施形態の移動画面を示した図である。

20

【図15】本実施形態の複写画面を示した図である。

【図16】本実施形態の削除後の画面を示した図である。

【図17】本実施形態のプログラムの移動、複写先のコンピュータを選択するフローを示した図である。

【図18】本実施形態の配備対象プログラム管理テーブルを示した図である。

【図19】本実施形態のプログラム構成差分情報管理テーブルを示した図である。

【図20】本実施形態のプログラム構成差分生成フローを示した図である。

【図21】本実施形態のスク립ト部品を示した図である。

【図22】本実施形態のOS配備スク립トテンプレートを示した図である。

【図23】本実施形態のOSパッチ配備スク립トテンプレートを示した図である。

30

【図24】本実施形態のプログラム配備スク립トテンプレートを示した図である。

【図25】本実施形態のプログラムパッチ配備スク립トテンプレートを示した図である。

【図26】本実施形態のプログラム削除スク립トテンプレートを示した図である。

【図27】本実施形態のプログラム複写スク립トシナリオを示した図である。

【図28】本実施形態のプログラム削除スク립トシナリオを示した図である。

【図29】本実施形態のプログラム移動スク립トシナリオを示した図である。

【図30】本実施形態の複写スク립ト生成フローを示した図である。

【図31】本実施形態のプログラム配備テンプレート設定処理を示した図である。

【図32】本実施形態の削除スク립ト生成フローを示した図である。

40

【図33】本実施形態のスク립トテンプレートにスク립ト部品設定するフローを示した図である。

【図34】本実施形態のパラメタ用データモデルのテンプレートを示した図である。

【図35】本実施形態のデータモデル生成フローを示した図である。

【図36】本実施形態のスク립ト実行フローを示した図である。

【図37】本実施形態の業務構成差分情報管理テーブルを示した図である。

【図38】本実施形態のグループ構成差分情報管理テーブルを示した図である。

【図39】本実施形態の業務構成差分情報管理テーブル生成フローを示した図である。

【図40】本実施形態のグループ構成差分情報管理テーブル生成フローを示した図である。

50

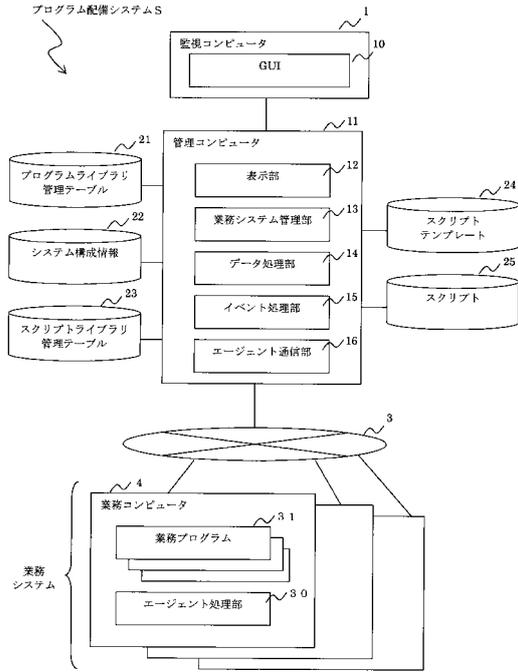
【図 4 1】本実施形態の業務の複写、移動、削除を実施する処理フローを示した図である。

【符号の説明】

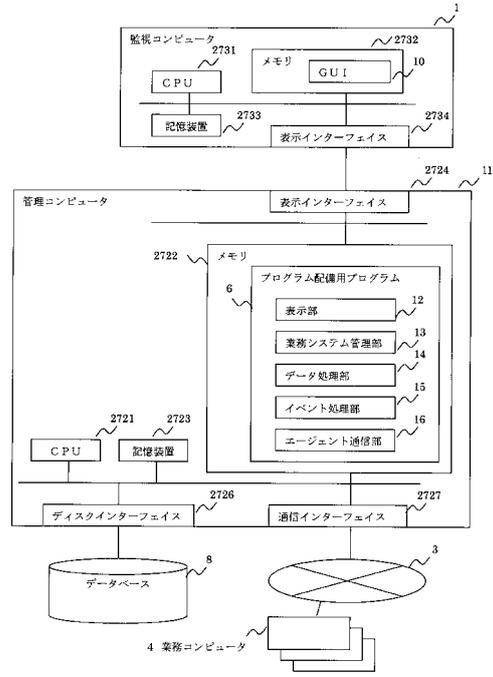
【 0 1 6 5 】

1	監視コンピュータ	
4	業務コンピュータ	
8	データベース	
1 0	G U I	
1 1	管理コンピュータ	
2 1	プログラムライブラリ管理テーブル	10
2 2	システム構成情報	
2 3	スクリプトライブラリ管理テーブル	
2 4	スクリプトテンプレート	
2 5	スクリプト	
1 0 1	プログラム構成管理テーブル	
1 0 2	業務構成情報管理テーブル	
1 0 3	OS 情報管理テーブル	
1 0 4	物理構成情報管理テーブル	
1 0 5	プログラム構成差分情報管理テーブル	
1 0 6	配備対象プログラム管理テーブル	20
1 0 7	データモデルテンプレート	
1 0 8	データモデル	
1 0 9	プログラム稼動最大構成テンプレート	
2 7 2 1、2 7 3 1	C P U	
2 7 2 3、2 7 3 3	記憶装置	
2 7 2 2、2 7 3 2	メモリ	

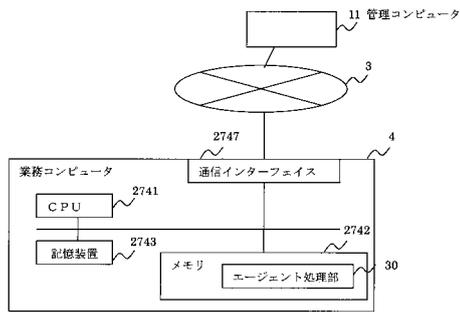
【図1】



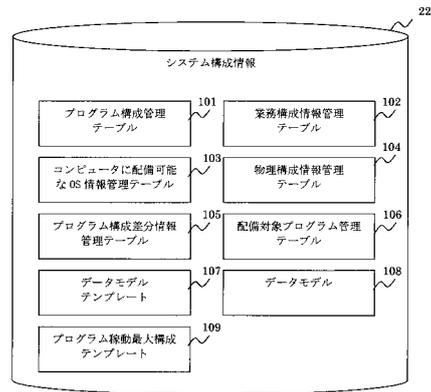
【図2】



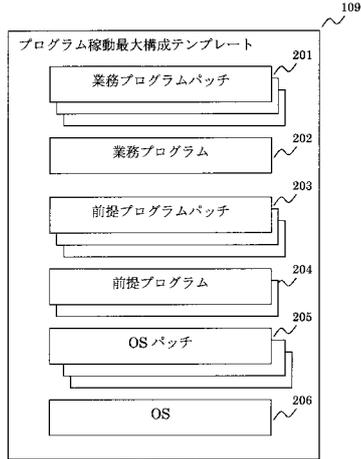
【図3】



【図4】



【図5】



【図6】

プログラムライブラリ ID	プログラムレベル	プログラム名	対象 OS	レポートフラグ
0001	1	OS1	—	ON
0002	1	OS2	—	ON
0003	2	Patch11	OS1	ON
0004	2	Patch12	OS1	ON
0005	3	AP1	OS1	OFF
0006	3	AP1	OS2	OFF
0007	3	AP2	OS1	ON
0008	3	AP2	OS2	ON
0009	3	AP2 Patch1	OS1	OFF
0010	3	AP2 Patch1	OS2	OFF

【図7】

項目	値
401 業務プログラム名	AP 1
402 プログラムライブラリ ID	0005
403 配備用スクリプト部品	スクリプト 1
404 削除用スクリプト部品	スクリプト 2
405 環境設定用スクリプト部品	スクリプト 3
406 連携プログラム設定用スクリプト部品	スクリプト 4
407 連携元プログラム設定変更用スクリプト部品	スクリプト 5
408 プログラム開始用スクリプト部品	スクリプト 6
409 プログラム停止用スクリプト部品	スクリプト 7

【図8】

コンピュータ名	配備可能 OS 名
コンピュータ 1	OS1, OS2
コンピュータ 2	OS1, OS2
コンピュータ 3	OS1, OS2
コンピュータ 4	OS1, OS2
コンピュータ 5	OS1, OS2

【図 9】

業務システム名	グループ名	業務プログラム名	コンピュータ名
業務 1		AP1	コンピュータ 1
業務 2		AP2	コンピュータ 2
		AP3	コンピュータ 2
業務 3		AP4	コンピュータ 3
	グループ 1	AP5	コンピュータ 3
	グループ 1	AP6	コンピュータ 3
業務 4	グループ 2	AP7	コンピュータ 7
	グループ 2	AP8	コンピュータ 7
	グループ 3	AP7	コンピュータ 8
	グループ 3	AP9	コンピュータ 8
		AP10	コンピュータ 9

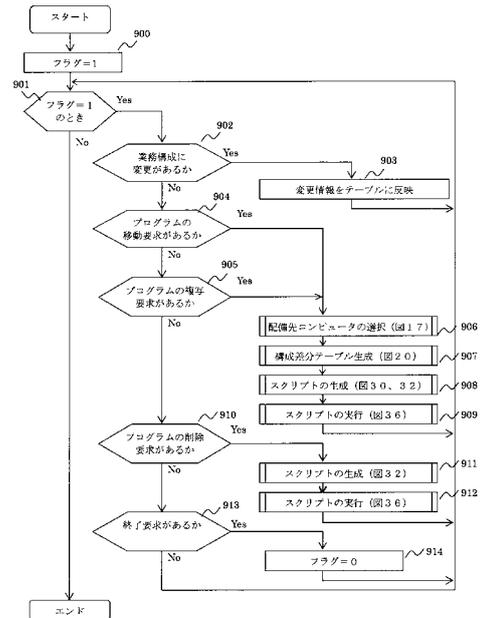
【図 10】

プログラムレベル	プログラム種別	コンピュータ 1	コンピュータ 2	コンピュータ 3	コンピュータ 4
1	OS	OS1	OS2	OS1	OS1
2	OS パッチ	Patch11	Patch21	Patch11	
		Patch12	Patch22		
3	業務プログラム	AP1	AP2	AP4	
			AP3	AP5	
				AP6	

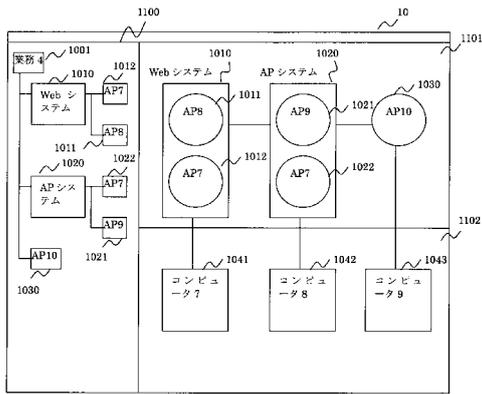
【図 11】

業務プログラム名	OS 名	OS パッチ名	前提プログラム名	連携先プログラム名
AP1				
AP2	OS2	Patch21		
		Patch22		
AP3			AP2	AP4
AP4	OS1	Patch11		AP6
AP4	OS2	Patch21		AP6

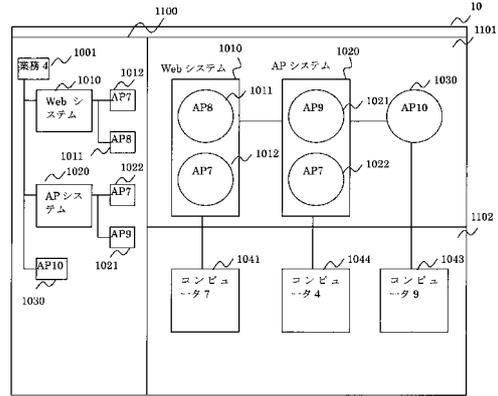
【図 12】



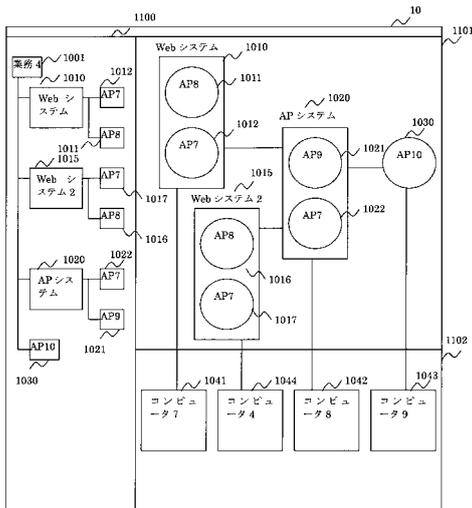
【図13】



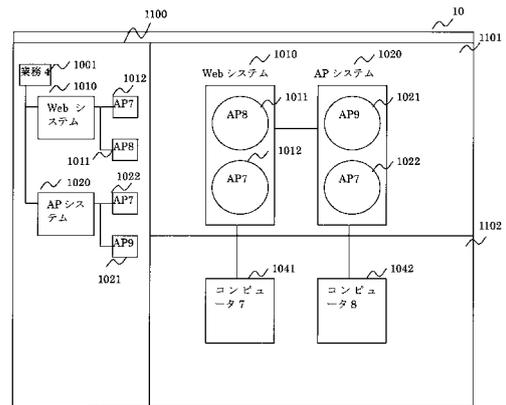
【図14】



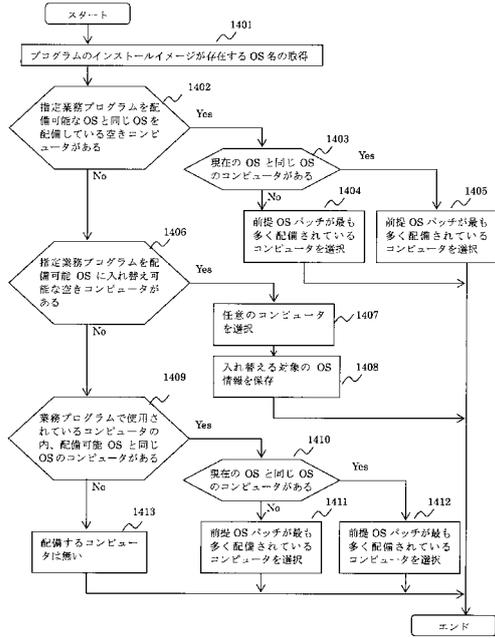
【図15】



【図16】



【図 17】



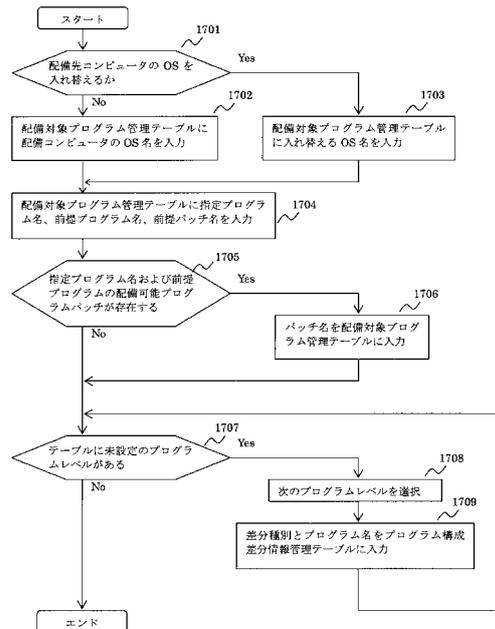
【図 18】

プログラムレベル	最大構成テンプレート	プログラム名
1511	OS	OS1
1512	OS パッチ	Patch11
		Patch12
1513	前提プログラム	AP7
		前提プログラム パッチ
1514	業務プログラム	AP8
1515	業務プログラム パッチ	
1516		

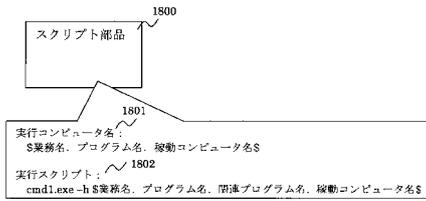
【図 19】

プログラムレベル	差分種別	プログラム名
1		OS1
2		Patch11
	+	Patch12
3	-	AP1
	-	AP4
	-	AP4 Patch1
	+	AP7
	+	AP7 Patch1
	+	AP8

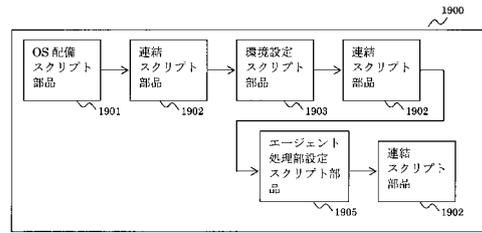
【図 20】



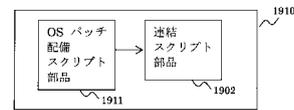
【図 2 1】



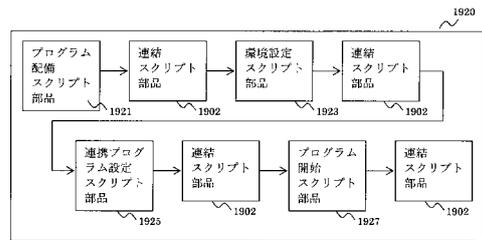
【図 2 2】



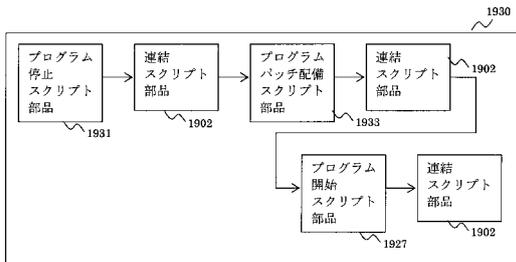
【図 2 3】



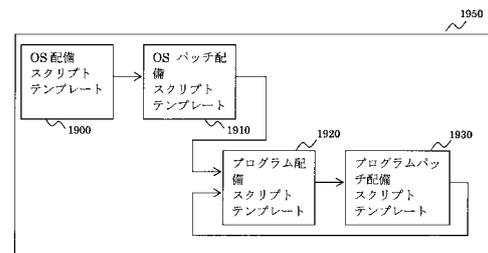
【図 2 4】



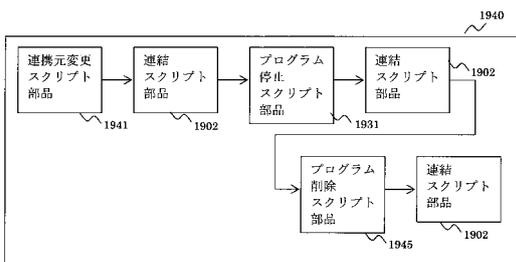
【図 2 5】



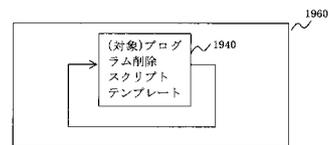
【図 2 7】



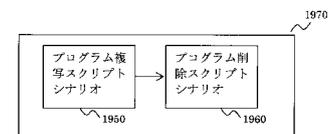
【図 2 6】



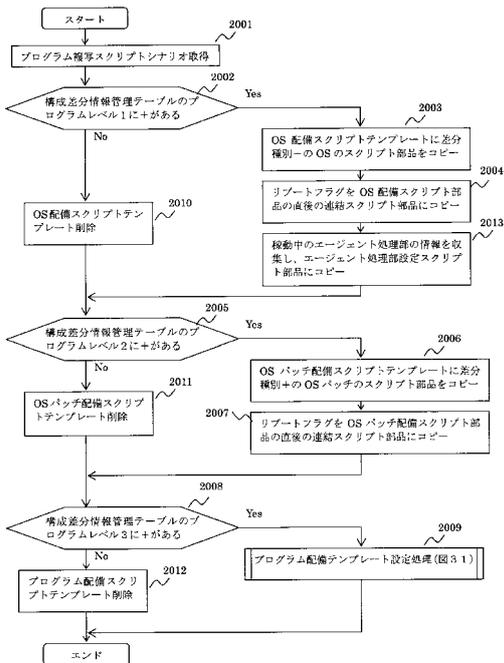
【図 2 8】



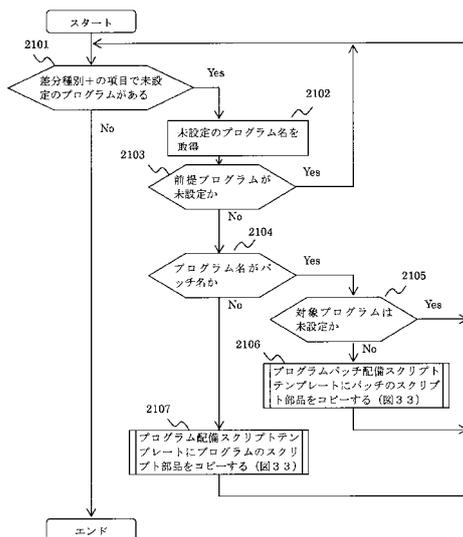
【図 2 9】



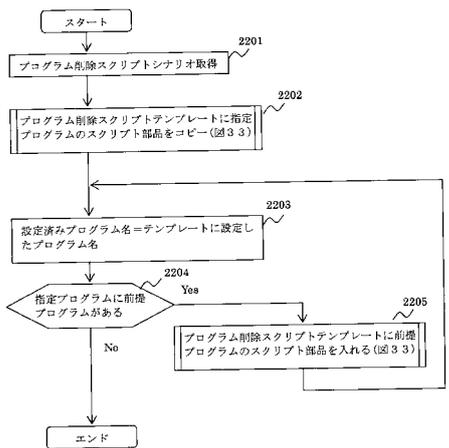
【図30】



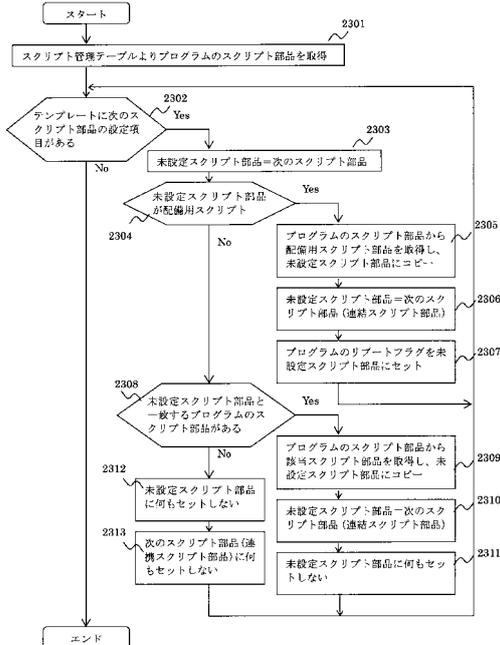
【図31】



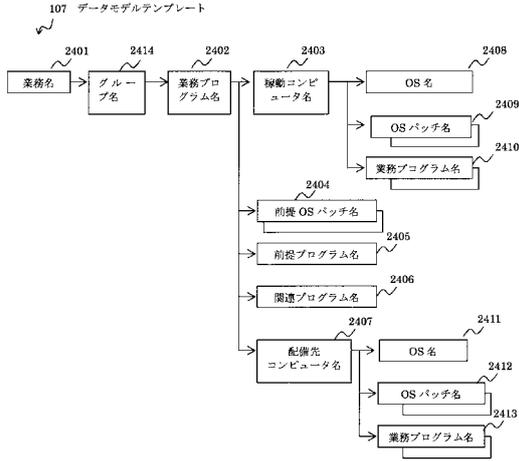
【図32】



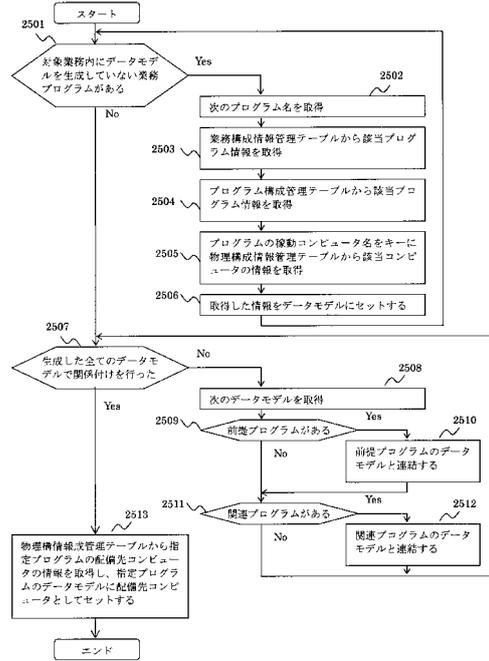
【図33】



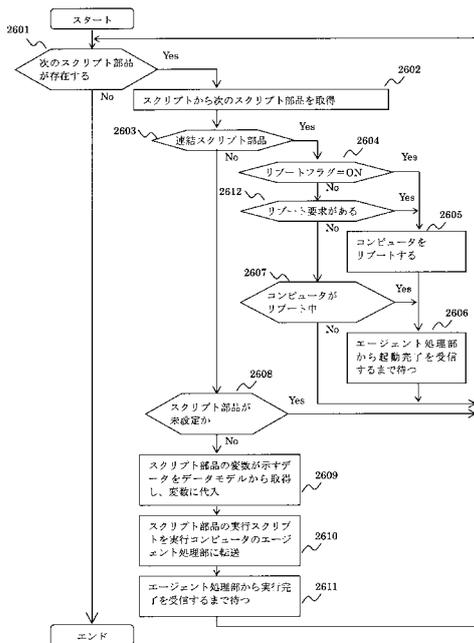
【図34】



【図35】



【図36】



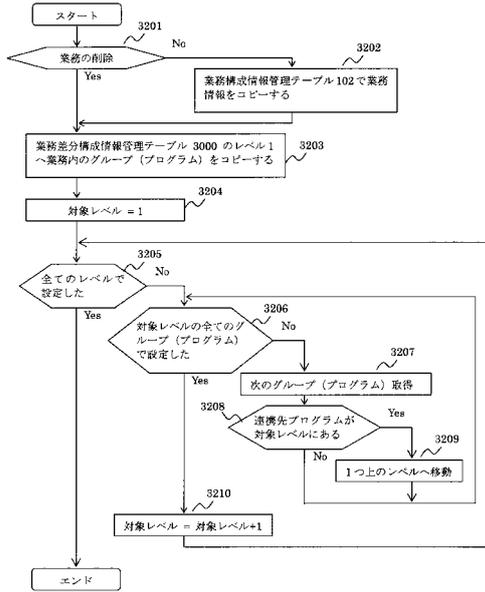
【図37】

業務レベル	差分種別	業務名	業務プログラム名又はグループ名
1	+	業務 4-1	AP10
	-	業務 4	AP10
2	+	業務 4-1	グループ 3
	-	業務 4	グループ 3
3	+	業務 4-1	グループ 2
	-	業務 4	グループ 2

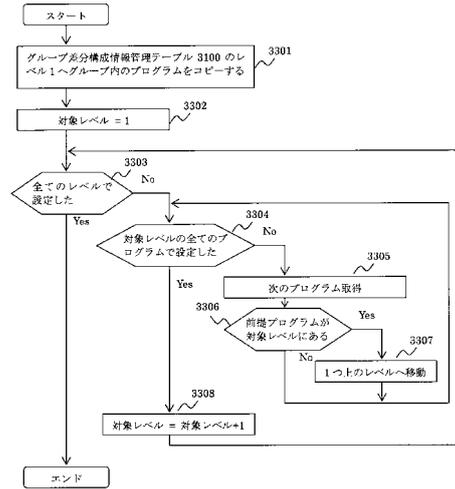
【図38】

グループレベル	差分種別	業務名	グループ名	業務プログラム名
1	+	業務 4-1	グループ 2	AP7
	+	業務 4-1	グループ 2	AP8

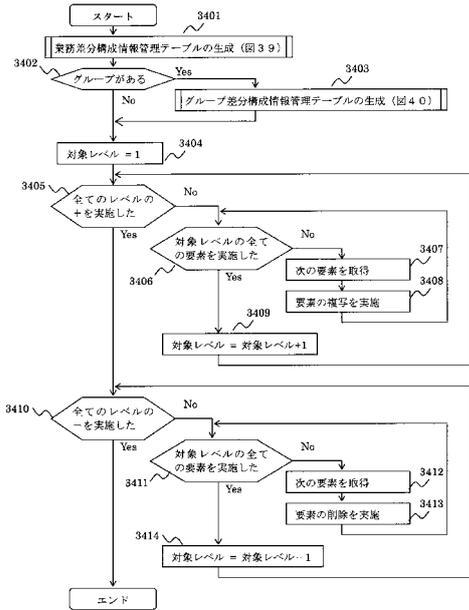
【図 39】



【図 40】



【図 41】



---

フロントページの続き

- (56)参考文献 特開2002-049493(JP,A)  
特開2005-234766(JP,A)  
特開2000-099348(JP,A)  
特開2000-353087(JP,A)

- (58)調査した分野(Int.Cl., DB名)  
G06F 9/445