

(19)日本国特許庁(JP)

(12)特許公報(B2)

(11)特許番号
特許第7018356号
(P7018356)

(45)発行日 令和4年2月10日(2022.2.10)

(24)登録日 令和4年2月2日(2022.2.2)

(51)国際特許分類 F I
G 0 6 F 8/34 (2018.01) G 0 6 F 8/34

請求項の数 13 (全38頁)

(21)出願番号	特願2018-100054(P2018-100054)	(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
(22)出願日	平成30年5月24日(2018.5.24)	(74)代理人	110002365 特許業務法人サンネクスト国際特許事務所
(65)公開番号	特開2019-204373(P2019-204373 A)	(72)発明者	嶋田 敬太 東京都千代田区丸の内一丁目6番6号 株式会社日立製作所内
(43)公開日	令和1年11月28日(2019.11.28)	(72)発明者	中村 秀樹 東京都千代田区丸の内一丁目6番6号 株式会社日立製作所内
審査請求日	令和2年11月18日(2020.11.18)	(72)発明者	鈴木 克典 東京都千代田区丸の内一丁目6番6号 株式会社日立製作所内

最終頁に続く

(54)【発明の名称】 ビジュアルプログラミングツールを用いてプログラムを作成することを支援する装置および方法

(57)【特許請求の範囲】

【請求項1】

(A) ビジュアルプログラミングツールを用いて作成されるプログラムを構成するN個のモジュール(Nは2以上の整数)のうち選択されたM個のモジュール(Mは、2以上且つN以下の整数)であるサブフローのうち、P個のモジュール(Pは、2以上且つM以下)を決定し、

各モジュールは、一個のプロパティを有し、

各プロパティは、一個以上のプロパティ要素を含み、

各プロパティ要素は、プロパティ項目とプロパティ値との組を含み、

(B) 一個のプロパティ項目として共通化可能であり当該一個のプロパティ項目の更新が反映されるQ個のプロパティ項目(Qは2以上且つP以下)であるプロパティ項目群がP個のプロパティに一個以上あれば、当該一個以上のプロパティ項目群の各々について、当該プロパティ項目群であり互いに異なるプロパティに属するQ個のプロパティ項目を一個のプロパティ項目として共通化可能であることを提示する、

ことを計算機に実行させるコンピュータプログラム。

【請求項2】

前記P個のモジュールの各々は、それぞれが複数個のモジュールを含む複数個のモジュールグループのうちモジュール関連ポイントが相対的に又は一定値以上に高いモジュールグループに属するモジュールであり、

各モジュールグループについて、モジュール関連ポイントは、それぞれがモジュール間の

関係性の評価のために用意された複数個のポリシーである複数個のモジュール比較ポリシーのうち当該モジュールグループが満たす一個以上のモジュール比較ポリシーに基づき算出された評価値である、

請求項 1 に記載のコンピュータプログラム。

【請求項 3】

前記複数個のモジュール比較ポリシーは、モジュール間のホップ数に関するポリシーである接続関係ポリシーと、前記ビジュアルプログラミングツールが提供するワークスペースにおけるモジュール位置に関するポリシーである位置ポリシーとのうちの少なくとも一個を含み、

前記各モジュールグループについて、前記モジュール関連ポイントの算出には、前記接続関係ポリシー及び前記位置ポリシーのうちの少なくとも一個が、前記接続関係ポリシー及び前記位置ポリシー以外のモジュール比較ポリシーに比べて優先的に使用される、

請求項 2 に記載のコンピュータプログラム。

【請求項 4】

前記複数個のモジュール比較ポリシーは、モジュールが配備される物理的又は仮想的な計算機リソースであるITインフラに関するポリシーであるITインフラポリシーを含む、

請求項 2 に記載のコンピュータプログラム。

【請求項 5】

前記一個以上のプロパティ項目群の各々は、それぞれがプロパティ要素間の関係性の評価のために用意された 1 又は複数個のポリシーである 1 又は複数個のプロパティ比較ポリシーのうち当該プロパティ項目群に適用されたプロパティ比較ポリシーを満たしたプロパティ項目群である、

請求項 1 に記載のコンピュータプログラム。

【請求項 6】

前記一個以上のプロパティ項目群の各々について、前記適用されたプロパティ比較ポリシーは、下記 (x) が下記 (y) に適合したプロパティ比較ポリシーである、

(x) それぞれがプロパティ要素に関する評価のために用意された 1 又は複数個のポリシーである 1 又は複数個のプロパティポリシーのうち、当該プロパティ項目群に属するプロパティ項目が満たすプロパティポリシー、

(y) 前記 1 又は複数個のプロパティポリシーのうち、当該プロパティ比較ポリシーに予め関連付けられていたプロパティポリシー、

請求項 5 に記載のコンピュータプログラム。

【請求項 7】

前記 P 個のモジュールの各々は、それぞれが複数個のモジュールを含む複数個のモジュールグループのうちモジュール関連ポイントが相対的に又は一定値以上に高いモジュールグループのうち、プロパティ関連ポイントが相対的に又は一定値以上に高いモジュールグループに属するモジュールであり、

各モジュールグループについて、

モジュール関連ポイントは、それぞれがモジュール間の関係性の評価のために用意された一又は複数個のポリシーである一又は複数個のモジュール比較ポリシーのうち当該モジュールグループが満たす一個以上のモジュール比較ポリシーに基づき算出された評価値であり、

プロパティ関連ポイントは、前記一又は複数個のプロパティ比較ポリシーのうち当該モジュールグループが満たす一個以上のプロパティ比較ポリシーに基づき算出された評価値である、

請求項 5 に記載のコンピュータプログラム。

【請求項 8】

前記サブフローのうちのいずれかのモジュールが変更された場合に、当該変更後のモジュールを含む前記 P 個のモジュールについて (B) を実行する、

請求項 1 に記載のコンピュータプログラム。

10

20

30

40

50

【請求項 9】

前記複数個のモジュール比較ポリシーは、
 ソースコードデータを管理するソースコード管理システムに対してモジュールグループに
 属する全モジュールのソースコードが同時期に新規に作成されたことを定義したポリシー
 と、
 前記ソースコード管理システムに対してモジュールグループに属する全モジュールのソー
 スコードが同時期に更新されたことを定義したポリシーと
 の少なくとも 1 個のポリシーを含む、
 請求項 2 に記載のコンピュータプログラム。

【請求項 10】

前記複数個のモジュール比較ポリシーは、モジュールグループに属する全モジュールが同
 一の開発者によって開発されたことを定義するポリシーを含む、
 請求項 2 に記載のコンピュータプログラム。

【請求項 11】

前記複数個のモジュール比較ポリシーは、モジュールグループに属する全モジュールにつ
 いて、基となるモジュール又はモジュール雛形のカテゴリが同一であることを定義するポ
 リシーを含む、
 請求項 2 に記載のコンピュータプログラム。

【請求項 12】

集約候補生成部と、
集約実行部と
を備え、
 前記集約候補生成部は、ビジュアルプログラミングツールを用いて作成されるプログラム
 を構成する N 個のモジュール (N は 2 以上の整数) のうちの選択された M 個のモジュール
 (M は、2 以上且つ N 以下の整数) であるサブフローのうち、P 個のモジュール (P は、
 2 以上且つ M 以下) を決定し、
各モジュールは、一個のプロパティを有し、
 各プロパティは、一個以上のプロパティ要素を含み、
 各プロパティ要素は、プロパティ項目とプロパティ値との組を含み、
 前記集約実行部は、一個のプロパティ項目として共通化可能であり当該一個のプロパティ
項目の更新が反映される Q 個のプロパティ項目 (Q は 2 以上且つ P 以下) であるプロパテ
ィ項目群が P 個のプロパティに一個以上あれば、当該一個以上のプロパティ項目群の各々
について、当該プロパティ項目群であり互いに異なるプロパティに属する Q 個のプロパテ
ィ項目を一個のプロパティ項目として共通化可能であることを提示する、
 プログラム作成支援装置。

【請求項 13】

(A) コンピュータが、ビジュアルプログラミングツールを用いて作成されるプログラム
 を構成する N 個のモジュール (N は 2 以上の整数) のうちの選択された M 個のモジュール
 (M は、2 以上且つ N 以下の整数) であるサブフローのうち、P 個のモジュール (P は、
2 以上且つ M 以下) を決定し、
各モジュールは、一個のプロパティを有し、
 各プロパティは、一個以上のプロパティ要素を含み、
 各プロパティ要素は、プロパティ項目とプロパティ値との組を含み、
 (B) コンピュータが、一個のプロパティ項目として共通化可能であり当該一個のプロパ
 ティ項目の更新が反映される Q 個のプロパティ項目 (Q は 2 以上且つ P 以下) であるプロ
 パティ項目群が P 個のプロパティに一個以上あれば、当該一個以上のプロパティ項目群の
 各々について、当該プロパティ項目群であり互いに異なるプロパティに属する Q 個のプロ
パティ項目を一個のプロパティ項目として共通化可能であることを提示する、
 プログラム作成支援方法。

【発明の詳細な説明】

10

20

30

40

50

【技術分野】

【0001】

本発明は、概して、ビジュアルプログラミングツールを用いたプログラム開発の支援に関する。

【背景技術】

【0002】

近年、プログラミングに対する深い知識がないユーザでも素早くアプリケーションプログラム（開発対象のプログラムの一例）を作成できるビジュアルプログラミングツールに注目が集まっている。

【0003】

ビジュアルプログラミングツールによれば、専用のワークスペース上に、アイコンで表現された特定の処理を有するモジュールが配置され、それらをワイヤといったモジュール間の接続関係を表す線で繋げられることにより、アプリケーションプログラムが作成される。

【0004】

ビジュアルプログラミングの長所の一個としては、テキストエディタなどでソースコードを直接編集するプログラミングよりもプログラム全体の処理を理解しやすく、モジュールの形式で処理をまとめておけば、それを再利用してプログラミングすることで作業量を減らすことがある。

【0005】

一部のビジュアルプログラミングツール（例えばNode-REDなど）では、アプリケーションプログラムを構成する複数個のモジュールとそれらを接続するワイヤとの集合であるサブフローを一個のモジュールとする機能（サブフロー機能）が用意されている。サブフローを構成する複数個のモジュールは、例えば、相対的に関連性の高い複数個のモジュールである。特許文献1は、モジュール間の関連性を評価する技術を開示する。具体的には、例えば、特許文献1に開示の計測装置は、プログラムをモジュール単位に分割し、モジュールに属するデータとモジュールの呼び出し操作を計測することでモジュール間の依存度を定量化する。

【先行技術文献】

【特許文献】

【0006】

【文献】特開2000-215045号公報

【発明の概要】

【発明が解決しようとする課題】

【0007】

ビジュアルプログラミングツールでは、モジュール毎にプロパティが存在する。プロパティは、1以上（典型的には複数）のプロパティ要素を含む。各プロパティ要素は、プロパティ項目（例えば「IPアドレス」）とプロパティ値（例えば、“1.1.1.1”）との組を含む。

【0008】

サブフロー機能によって複数個のモジュールが一個のサブフローとされても、当該サブフローを再利用（例えば、他のアプリケーションプログラムへ再利用）するためには、当該サブフロー内部の構成を表す階層に移動して個々のモジュールに必要なプロパティを見直し必要に応じて修正をしなければならない。結果として、ユーザの設定工数が増えてしまう。

【課題を解決するための手段】

【0009】

プログラム作成支援装置が、ビジュアルプログラミングツールを用いて作成されるプログラムを構成するN個のモジュール（Nは2以上の整数）のうちの選択されたM個のモジュール（Mは、2以上且つN以下の整数）であるサブフローのうち、P個のプロパティをそれぞれ有するP個のモジュール（Pは、2以上且つM以下）を決定する。当該装置は、一個のプロパティ項目として共通化可能であり当該一個のプロパティ項目の更新が反映されるQ個

10

20

30

40

50

のプロパティ項目（Qは2以上且つP以下）であるプロパティ項目群がP個のプロパティに一個以上あれば、当該一個以上のプロパティ項目群の各々について、共通化可能であることを提示する。

【発明の効果】

【0010】

サブフローのプロパティについてユーザの設定工数が減少する。

【図面の簡単な説明】

【0011】

【図1】本発明の実施例1における計算機システムの一例を示すブロック図である。

【図2】本発明の実施例1における集約モジュール生成の一例の概要を示す図である。

10

【図3】本発明の実施例1の一比較例の概要を示す図である。

【図4】本発明の実施例1でのプロパティ設定の概要の一例を示す図である。

【図5】本発明の実施例1におけるモジュール管理テーブルの一例を示す図である。

【図6】本発明の実施例1におけるポリシー管理テーブルの一例を示す図である。

【図7】本発明の実施例1におけるインフラ管理テーブルの一例を示す図である。

【図8】本発明の実施例1における選択範囲管理テーブルの一例を示す図である。

【図9】本発明の実施例1におけるモジュール比較テーブルの一例を示す図である。

【図10】本発明の実施例1におけるプロパティテーブルの一例を示す図である。

【図11】本発明の実施例1における表示条件テーブルの一例を示す図である。

【図12】本発明の実施例1における表示グループテーブルの一例を示す図である。

20

【図13】本発明の実施例1におけるポリシー登録部により行われる処理の一例を示す。

【図14】本発明の実施例1におけるインフラ関連生成部により行われる処理の一例を示す。

【図15A】本発明の実施例1における集約候補部により行われる処理の一例の前部分を示す。

【図15B】本発明の実施例1における集約候補部により行われる処理の一例の中部分を示す。

【図15C】本発明の実施例1における集約候補部により行われる処理の一例の後部分を示す。

【図16】本発明の実施例1における表示グループ生成部により行われる処理の一例を示す。

30

【図17】本発明の実施例1における集約実行部により行われる処理の一例を示す。

【図18】本発明の実施例1におけるプロパティ共通化画面の一例を説明する図である。

【図19】本発明の実施例1における確認画面の一例を説明する図である。

【図20】本発明の実施例2における変更実行部により行われる処理の一例を示す。

【発明を実施するための形態】

【0012】

以下の説明では、「インタフェース部」は、一個以上のインタフェースで良い。当該一個以上のインタフェースは、ユーザインタフェース部と、通信インタフェース部とのうちの少なくとも通信インタフェース部を含んで良い。ユーザインタフェース部は、一個以上のI/Oデバイス（例えば入力デバイス（例えばキーボードおよびポインティングデバイス）と出力デバイス（例えば表示デバイス））と表示用計算機とのうちの少なくとも一個のI/Oデバイスでも良いし、それに代えてまたは加えて、当該少なくとも一個のI/Oデバイスに対するインタフェースデバイス（I/Oインタフェース）でも良い。通信インタフェース部は、一個以上の通信インタフェースデバイスで良い。一個以上の通信インタフェースデバイスは、一個以上の同種の通信インタフェースデバイス（例えば一個以上のNIC（Network Interface Card））であっても良いし二個以上の異種の通信インタフェースデバイス（例えばNICとHBA（Host Bus Adapter））であっても良い。

40

【0013】

また、以下の説明では、「メモリ部」は、一個以上のメモリであり、典型的には主記憶デ

50

デバイスで良い。メモリ部における少なくとも一個のメモリは、揮発性メモリであっても良いし不揮発性メモリであっても良い。

【0014】

また、以下の説明では、「PDEV部」は、一個以上のPDEVであり、典型的には補助記憶デバイスで良い。「PDEV」は、物理的な記憶デバイス（Physical storage DEvice）を意味し、典型的には、不揮発性の記憶デバイス、例えばHDD（Hard Disk Drive）またはSSD（Solid State Drive）である。

【0015】

また、以下の説明では、「記憶部」は、メモリ部およびPDEV部のうちの少なくとも一個（典型的には少なくともメモリ部）である。

10

【0016】

また、以下の説明では、「プロセッサ部」は、一個以上のプロセッサである。少なくとも一個のプロセッサは、典型的には、CPU（Central Processing Unit）のようなマイクロプロセッサであるが、GPU（Graphics Processing Unit）のような他種のプロセッサでも良い。少なくとも一個のプロセッサは、シングルコアでも良いしマルチコアでも良い。少なくとも一個のプロセッサは、処理の一部または全部を行うハードウェア回路（例えばFPGA（Field-Programmable Gate Array）またはASIC（Application Specific Integrated Circuit））といった広義のプロセッサでも良い。

【0017】

また、以下の説明では、「kkk部」（インタフェース部、記憶部及びプロセッサ部を除く）の表現にて機能を説明することがあるが、機能は、一個以上のコンピュータプログラムがプロセッサ部によって実行されることで実現されても良いし、一個以上のハードウェア回路（例えばFPGA又はASIC）によって実現されても良い。プログラムがプロセッサ部によって実行されることで機能が実現される場合、定められた処理が、適宜に記憶部及び/又はインタフェース部等を用いながら行われるため、機能はプロセッサ部の少なくとも一部とされても良い。機能を主語として説明された処理は、プロセッサ部あるいはそのプロセッサ部を有する装置が行う処理としても良い。プログラムは、プログラムソースからインストールされても良い。プログラムソースは、例えば、プログラム配布計算機又は計算機が読み取り可能な記録媒体（例えば非一時的な記録媒体）であっても良い。各機能の説明は一例であり、複数個の機能が一個の機能にまとめられたり、一個の機能が複数個の機能に分割されたりしても良い。

20

30

【0018】

また、以下の説明では、「xxxテーブル」といった表現にて、入力に対して出力が得られる情報を説明することがあるが、当該情報は、どのような構造のデータでも良いし、入力に対する出力を発生するニューラルネットワークのような学習モデルでも良い。従って、「xxxテーブル」を「xxx情報」と言うことができる。また、以下の説明において、各テーブルの構成は一例であり、一個のテーブルは、二個以上のテーブルに分割されても良いし、二個以上のテーブルの全部又は一部が一個のテーブルであっても良い。

【0019】

以下、本発明の幾つか実施例について添付図面を用いて説明する。実施例で用いるテーブルに使用する識別情報は、当該テーブルに一個以上のカラムを用いてテーブルのエントリを特定できる場合、識別情報のカラムを省略することもできる。また、識別情報は昇順となる識別番号を自動的に割り当てられても良い。各図において共通の構成要素については同一の参照符号が付されている。同種の構成要素を区別しないで説明する場合には、参照符号のうちの共通符号を使用し、同種の構成要素を区別する場合は、参照符号を使用することがある。例えば、モジュールを区別しない場合には、「モジュール205」と言い、モジュールを区別する場合には、「モジュール205A」、「モジュール205B」のように言う。

40

【実施例1】

【0020】

50

実施例1のプログラム作成支援装置の一例である管理サーバは、ユーザが複数個のモジュールを指定して、それらを一個のモジュールに集約する（集約モジュール（サブフロー）を作成する）際に、集約モジュールを構成する複数個のモジュールにのうち同様の内容と思われるプロパティ要素を検出し、それらを共通化するか否かをユーザに提示する。集約モジュールの共通化されたプロパティ要素の更新が、当該集約モジュールに属する二個以上のモジュールの各々のプロパティ要素の更新となる。これにより、ユーザは集約モジュールのプロパティの数を削減することができ、結果として、ユーザの集約モジュールに対する設定工数が減少する。

【0021】

具体的には、実施例1では、管理サーバは、アプリケーションプログラム（プログラムの一例）を構成する複数個のモジュールを集約してモジュール（サブフロー）を作成する際に、モジュール間の接続関係、ワークスペース（ビジュアルプログラミングツールが提供する表示スペース）内の位置情報、及び、モジュールのITインフラ情報を含む複数個のポリシーのうちいずれのポリシーにモジュール間が合致するかにより、モジュール間の関連性を評価する。管理サーバは、関連性が高い（例えば、相対的に関連性が高い、又は、関連性が所定の閾値以上である）モジュールの組を抽出する。管理サーバは、抽出された組毎に、当該組を構成する二個のモジュールがそれぞれ有する二個のプロパティのうち関連性が高い（例えば、相対的に関連性が高い、又は、関連性が所定の閾値以上である）プロパティ要素の組を抽出する。そして、管理サーバは、抽出したプロパティ要素の組を、一個のプロパティ要素に共通化できる可能性が高い組としてユーザに提示する。そして、管理サーバは、ユーザに提示したプロパティ要素の組のうちユーザが共通化する組の選択を受け付ける。管理サーバは、ユーザから選択されたプロパティ要素の組の各々について、当該組に属する2つのプロパティ要素を、共通化する一個のプロパティ要素として決定し、且つ、当該プロパティ要素を、集約モジュールのプロパティにおける一個のプロパティ要素とする。

【0022】

以下、実施例1を詳細に説明する。なお、管理サーバが「情報を表示する」ことは、管理サーバが有する表示装置に表示用の情報を表示することであってもよいし、管理サーバが表示用の計算機に表示用の情報を送信することであってもよい（後者の場合は表示用の計算機によって表示用の情報が表示される）。

【0023】

図1は、実施例1の計算機システムの一例を示すブロック図である。

【0024】

実施例1の計算機システムは、管理サーバ101、アプリケーションプログラムをデプロイして実行するための実行環境102、ビジュアルプログラミングツール106から構成され、それらは管理ネットワーク（例えばIPネットワーク）103を介して互いに接続される。

【0025】

実行環境102は、複数の計算機、複数のストレージ装置、及び複数のネットワーク装置等を含んで良い。実行環境102は、物理的な装置そのものをリソースとして用いる環境でもよいし、仮想的な装置をリソースとして用いる環境でもよい。

【0026】

ビジュアルプログラミングツール106は、アプリケーションプログラムの処理フローを表示する。ビジュアルプログラミングツール106を用いたプログラム作成が管理サーバ101により支援される。ビジュアルプログラミングツール106は、入出力コンソールとしてのクライアント端末で実行されてもよいし、入出力コンソール（本実施例では入力装置104及び表示装置105）と通信可能な管理サーバ101で実行されてもよい。ビジュアルプログラミングツール106が提供する画面は、入出力コンソールに表示される。本実施例では、ビジュアルプログラミングツール106を用いたプログラミングが、管理サーバ101において制御部120がテーブル群130を基に行う処理により支援される。

【0027】

10

20

30

40

50

管理サーバ101は、ビジュアルプログラミングにおいて、複数個のモジュールを集約する際に利用する集約対象のモジュール間の関連性やプロパティ間の関連性を評価して、同様のプロパティを共通化する処理等を実施する。ここで、管理サーバ101のハードウェア構成及びソフトウェア構成について説明する。

【0028】

管理サーバ101は、CPU (Central Processing Unit) 111、メモリ112、ネットワークインタフェース113、及びI/Oインタフェース114を有する。これらのハードウェアは、内部バス(図示しない)を介して互いに接続される。CPU 111は、プロセッサ部の一例であり、メモリ112は、記憶部の一例であり、ネットワークインタフェース113及びI/Oインタフェース114は、インタフェース部の一例である。

10

【0029】

CPU 111は、メモリ112に格納される一個以上のプログラムを実行する。CPU 111は、メモリ112に格納されるプログラムにしたがって動作することによって、所定の機能を実現する制御部120として動作する。

【0030】

メモリ112は、CPU 111が実行する一個以上のプログラム、及び、当該一個以上のプログラムの実行において参照又は更新される複数個の管理テーブルである管理テーブル群130を格納する。また、メモリ112は、プログラムが一時的に使用するワークエリア(図示せず)を含む。

【0031】

ネットワークインタフェース113は、IPネットワーク等を介して外部の装置と通信するためのインタフェースである。I/Oインタフェース114は、入力装置104及び表示装置105と接続するインタフェースである。なお、I/Oインタフェース114は、USBメモリ及び外部ストレージ装置等、外部の記憶媒体に接続してもよい。

20

【0032】

入力装置104は、管理サーバ101にデータを入力するための装置(例えば、キーボード、マウス、及びタッチパネル等)である。表示装置105は、管理サーバ101の処理結果等を表示するための装置(例えば、ディスプレイ及びタッチパネル等)である。

【0033】

図1では、ハードウェアとしての各構成要素の数はそれぞれ一個であるが二個以上であってもよい。例えば、管理サーバ101が複数のネットワークと接続する場合、管理サーバ101やビジュアルプログラミングツール106は、各ネットワークと接続する複数個のネットワークインタフェースを有してもよい。

30

【0034】

ここで、管理サーバ101のメモリ112に格納されるプログラム及び情報について説明する。メモリ112は、制御部120を実現するプログラム、及び、管理テーブル群130を格納する。

【0035】

制御部120は、ポリシー登録部121(図13参照)、集約候補生成部122(図15A~図15C参照)、表示グループ生成部123(図16参照)、集約実行部124(図17参照)、インフラ情報取得部125及びインフラ関連生成部126(図14参照)を有する。

40

【0036】

管理テーブル群130は、モジュール管理テーブル131(図5参照)、ポリシー管理テーブル132(図6参照)、選択範囲管理テーブル133(図8参照)、モジュール比較テーブル134(図9参照)、プロパティテーブル135(図10参照)、表示条件テーブル136(図11参照)、表示グループテーブル137(図12参照)、及び、インフラ管理テーブル138(図7参照)を有する。

【0037】

ビジュアルプログラミングツール106は、ユーザに対してモジュール、ワイヤ、それらを配置するための専用のワークスペース等をGUI(Graphical User Interface)として表示

50

して、ユーザが視覚的にプログラミングを行えるようにする。ビジュアルプログラミングツール106は、画面表示部141及びプログラム管理部142を有する。

【0038】

画面表示部141は、アプリケーションプログラムを視覚的に表示し、ユーザがそれを編集できるようにするためのGUIを表示する。プログラム管理部142は、画面表示部141を通じて取得したアプリケーションプログラムに対するユーザの操作を管理サーバ101や実行環境102上に存在するプログラム実行部151に送信したり、管理サーバ101で生成した集約モジュールを作成する際にユーザに視覚化する必要のある情報（共通化するプロパティか否かを判断させるための情報等）を受信して画面表示部141に引き渡したりする。

【0039】

なお、本実施例においては、画面表示部141及びプログラム管理部142を有するビジュアルプログラミングツール106は、管理サーバ101とは別のサーバで実行されるが、これは一例である。例えば、ビジュアルプログラミングツール106は、管理サーバ101のCPU 11により実行されても良いし、実行環境102に存在する物理サーバや仮想サーバなどの計算機リソースにより実行されても良い。

【0040】

プログラム実行部151及びソースコードデータ152は、実行環境102に存在する物理サーバや仮想サーバなどの計算機リソースに展開されて処理が行われる。

【0041】

プログラム実行部151は、ビジュアルプログラミングツール106のプログラム管理部142からアプリケーションプログラムのソースコードデータ152（アプリケーションプログラムを構成するモジュールの情報やモジュールを接続するワイヤ情報等）を受信し、その受信したソースコードデータ152に記述された処理を実行する。実行方法は、プログラム実行部151がソースコードデータ152を読み込み、更にソースコードデータ152内の各モジュールで定義された処理をワイヤ等の接続関係を基に順序付けて実行する。なお、モジュール間でデータの受け渡しなどが発生する場合には、そのデータの中継なども行われる。また、モジュールで定義された処理によっては、プログラム実行部151を実行する計算機以外の実行環境102上の計算機や、管理ネットワーク103等のネットワークを経由して、実行環境102の外部にあるSaaS（Software as a Service）などのITサービスや計算機リソースが呼び出されても良い。もしくは、プログラム実行部151の代わりに、計算機上で実行可能なバイナリが生成されてもよい。

【0042】

図2は、本実施例における集約モジュール生成の一例の概要を示している。

【0043】

ビジュアルプログラミングツール106が提供するワークスペース（GUI）上に、アプリケーションプログラムの少なくとも一部を表現した処理フロー204が表示装置105に表示される。処理フロー204は、モジュール205A～205Fと複数のワイヤ206とを含む。ビジュアルプログラミングツール106は、例えば、処理フロー204の一部であるモジュール205B～205Eを集約範囲207とした指定をユーザ（開発者）202から入力装置104が受け付けると、図示の通り、それらを集約モジュール208とする。

【0044】

図3は、本実施例の一比較例の概要を示している。

【0045】

モジュール205には多くの場合そのモジュール205で処理を実行するため、モジュール205ごとに用意されたプロパティ211におけるプロパティ項目毎にプロパティ値を設定する必要がある。破線矢印312B～312Eは、モジュール205B～205Eとプロパティ311B～311Eとの関連を表したものである。

【0046】

集約モジュール208に関してプロパティ値を変更しようとした場合、集約モジュール208は一個のモジュールであるにも関わらず、集約モジュール内部314に示したように、ユー

10

20

30

40

50

ザ202は、集約モジュール208を構成する複数個のモジュール205B～205Eが分かるように展開して、更にプロパティの変更が必要なモジュール205を探し、そのモジュール205のプロパティ311の設定画面（例えばGUI）経由でプロパティ値の変更などを行う必要がある。つまり、集約モジュールの再展開、修正対象のモジュール205とプロパティ311の探索といった複雑な作業をユーザ202は行う必要がある。また、複数個のモジュール205をまとめて利用していると、同じプロパティ項目についての設定を複数個のモジュール205に跨って利用しているケースもあり、その場合は、それら複数個のモジュール205の各々についてプロパティ設定画面を個別に開いて同じ値を入力する必要がある。

【0047】

図4は、本実施例でのプロパティ設定の概要の一例を示している。

10

【0048】

本実施例では、ユーザ202が集約モジュール208を生成する際に、制御部120が、プロパティ共通化画面（例えばGUI）1800を表示して、その中で共通化レコメンデーション1804（例えば図18参照）を行う。共通化レコメンデーション1804とは、集約するモジュール205群のうち、共通化できる可能性があるプロパティ要素421C及び421Dの組み合わせを可視化したものである。ユーザ202は、共通化レコメンデーション1804の対象であるプロパティ要素421C及び421Dの組み合わせについて、共通化するか否を判断する。共通化することの指定がプロパティ共通化画面1800経由でされた組み合わせ（プロパティ要素421の組み合わせ）が、管理サーバ101に記憶され、制御部120が、それら共通化されたプロパティ要素の集合を、集約モジュール208のプロパティ423として作成する。制御部120が、集約モジュール208のプロパティ423を表示しても良い。

20

【0049】

集約モジュールのプロパティ423を作成しておくことで、ユーザ202は、集約モジュール208についてのプロパティの設定変更に関して、修正するプロパティ要素を探すために毎回集約モジュール内部314の編集画面に移行する作業や、集約モジュール内部314を表す処理フローから修正対象のプロパティ要素を有するモジュール205を探索する作業や、モジュール205毎にプロパティ設定画面を開いて設定を行う作業が必要なくなる。言い換えれば、集約モジュール208のプロパティ423から設定変更対象のプロパティ要素を見つけて変更すれば、集約モジュール208に属する二個以上のモジュール205の各々の該当のプロパティ要素が変更される。

30

【0050】

図5は、実施例1のモジュール管理テーブル131の一例を示す図である。

【0051】

モジュール管理テーブル131は、モジュール205を管理するための情報である。モジュール管理テーブル131は、モジュール205毎にエントリを有する。各エントリは、モジュールID 501、モジュール名502、X座標503、Y座標504、接続先ID 505、プロパティ506、作成コミット507、更新コミット508、カテゴリ509及び表示順510といった情報を格納する。以下、一個のモジュール205を例に取る（図5の説明において「対象モジュール205」）。なお、エントリには、図示の情報以外の情報が格納されてもよい（この点は、他のテーブル132～138についても同様である）。

40

【0052】

モジュールID 501は、対象モジュール205を識別するための識別子を示す。

【0053】

モジュール名502は、対象モジュール205を表示する際の名称を示す。

【0054】

X座標503及びY座標504は、対象モジュール205のワークスペース上での位置を示す位置情報に相当する。X座標503は、ワークスペース上でのX座標を示し、Y座標504は、ワークスペース上でのY座標を示す。本実施例では、一例として、X座標及びY座標は、ワークスペースの左上隅を原点とし、そこから対象モジュール205の中心部までの距離をピクセルという単位で示す。

50

【 0 0 5 5 】

接続先ID 505は、対象モジュールの一端（出力側）に一個のワイヤ経由で（他のモジュール205の介在無しに）接続されているモジュール205である接続先モジュール205のモジュールIDを示す。

【 0 0 5 6 】

プロパティ506は、対象モジュール205で処理を行うために必要な設定値を含むプロパティを示している。例えば、対象モジュール205がメール送信を行うモジュールの場合は、プロパティにおけるプロパティ項目は、メールサービスに接続するためのIPアドレス、ポート番号、ログインユーザ名、ログインパスワードなどがある。

【 0 0 5 7 】

作成コミット507は、対象モジュール205を含んだソースコード（ビジュアルプログラミングにおいては、プロパティを含んだモジュール205やモジュール205を接続するワイヤ206などの情報）をユーザが最初にgitなどのソースコード管理システム（図示せず）に登録した際にソースコード管理システムにより発行されたコミットIDを示す。コミットIDが同一である二個以上のモジュール205が、同時期にソースコード管理システムに登録されたモジュールであると判断することが可能となる。コミットIDは、同時期にソースコードの保存が行われていることが確認できれば、対象モジュール205の作成日時などが利用されても良い。また、「ソースコード管理システム」は、ソースコードデータを管理し当該ソースコードデータが登録されるシステムである。

【 0 0 5 8 】

更新コミット508は、ユーザがソースコードを編集して更新した際のコミットID（ソースコード管理システムに更新後のソースコードが登録された際に発行されるコミットID）を示す。コミットIDは、同時期にソースコードの保存（更新）が行われていることが確認できれば、更新日時などが利用されても良い。

【 0 0 5 9 】

カテゴリ509は、対象モジュール205が過去に作成されたモジュールやモジュールの雛形を基に作成されたモジュールである場合に、その過去に作成されたモジュールやモジュールの雛形が属していたカテゴリを示す。

【 0 0 6 0 】

表示順510は、モジュールの集約処理を進めていく上で表示するモジュール205の順番を示す。本実施例では、数値が小さいほど先に表示される。この順番は、それぞれのモジュールのX座標503及びY座標504の情報を基に計算しても良い。実施例1の例では、各モジュールが座標（0,0）からどの程度離れているかをX座標503及びY座標504から算出して、座標（0,0）に近いモジュールから昇順で順番が付与される。

【 0 0 6 1 】

図6は、実施例1のポリシー管理テーブル132の一例を示す図である。

【 0 0 6 2 】

ポリシー管理テーブル132は、モジュール205間の関係性の度合いや、プロパティ間の関係性の度合いや、プロパティの特徴などを評価するために利用する各種ポリシーの情報を格納する。ポリシー管理テーブル132は、ポリシー毎にエントリを有する。各エントリは、ポリシーID 601、種別名602、ポリシー内容603、ポイント604及び事前ポリシー605といった情報を格納する。以下、一個のポリシーを例に取る（図6の説明において「対象ポリシー」）。

【 0 0 6 3 】

ポリシーID 601は、対象ポリシーを識別するための識別子を示す。

【 0 0 6 4 】

種別602は、対象ポリシーの種別を示す。本実施例では、ポリシーの種別として、“モジュール比較”、“プロパティ比較”及び“プロパティ”の三つがある。“モジュール比較”のポリシーは、モジュール同士の関連性を評価するために利用されるポリシーである。“プロパティ比較”のポリシーは、プロパティ同士の関連性を評価するために利用されるポリシーである

10

20

30

40

50

。“プロパティ”のポリシーは、プロパティが特定の条件を満たしているかを評価するために利用されるポリシーである。

【0065】

ポリシー内容603は、対象ポリシーの内容（具体的には、モジュール205やプロパティを評価するためのパターンなど）を示す。

【0066】

ポイント604は、対象ポリシーを満たした場合に加算されるポイントを示す。対象ポリシーが、“モジュール比較”のポリシーの場合、ポイントは、後述のモジュール関連ポイントに加算される。対象ポリシーが、“プロパティ比較”又は“プロパティ”のポリシーの場合、ポイントは、後述のプロパティ関連ポイントに加算される。なお、本実施例では、モジュール関連ポイントやプロパティ関連ポイントといった評価値は、満たされたポリシーに対応したポイントの累計であるが、累計以外であっても良い（例えば、低いポイント（例えば、相対的に低い又は閾値未満のポイント）は、ノイズになる可能性があるため加算対象から外されてもよい）。また、例えば、ポリシーには追加の情報として、特定のモジュールやプロパティに対してはポリシーを適用しないなどして、過度なポリシーによる比較を避けるようにしても良い。また、ポイント604は、ユーザが他のポリシーとの兼ね合いをみて変更しても良い。

【0067】

事前ポリシー605は、別のポリシーを適用して評価を行う前に事前に評価しておくべきポリシーの識別情報である。

【0068】

なお、同一のプロパティ項目であるがプロパティ名が違う場合には、辞書などで同一のプロパティ項目であることが特定されても良い。

【0069】

図7は、実施例1のインフラ管理テーブル138の一例を示す図である。

【0070】

インフラ管理テーブル138は、モジュールが実際にデプロイされる物理サーバ、仮想サーバ、コンテナ実行基盤等のITインフラに関する情報を格納し、またITインフラ上で稼働するモジュールとの関連を表す情報を格納するテーブルである。また、モジュールがSaaS（Software as a Service）などのサービスに接続して処理を行うものであった場合は、SaaSに関する情報をインフラに関する情報として格納し、また、SaaSを利用するモジュールとの関連を表す情報も格納する。インフラ管理テーブル138は、ITインフラ毎にエントリを有する。各エントリが、インフラID 701、管理IPアドレス702、認証情報703、OS 704、構成705及び関連モジュール706といった情報を格納する。以下、一個のインフラを例に取る（図7の説明において「対象ITインフラ」）。

【0071】

インフラID701は、対象ITインフラを一意に識別するための識別子である。

【0072】

管理IPアドレス702は、対象ITインフラに割り当てられているIPアドレスを示す。認証情報703は、対象ITインフラに関する認証情報（例えばユーザID及びパスワード）である。このIPアドレスと認証情報とを基に対象ITインフラから対象ITインフラに関する情報を取得することができる。

【0073】

OS 704は、対象ITインフラのOSに関する情報である。構成705は、対象ITインフラのOSやコンテナ基盤となるミドルウェアなどの情報である。なお、構成705は、CPUやメモリなどのハードウェアの情報を含んだ情報でも良い。

【0074】

関連モジュール706は、対象ITインフラの上で稼働するモジュールの識別子を示す。結果として、対象ITインフラとモジュール205の結びつきが表現されている。なお、対象ITインフラとモジュールとの結びつきは、インフラ管理テーブル138のエントリに関連モジュ

10

20

30

40

50

ール706が格納されることに代えて又は加えて、モジュール管理テーブル131のエントリに、関連するITインフラの識別子を示す情報が格納されることで、表現されても良い。

【0075】

図8は、実施例1の選択範囲管理テーブル133の一例を示す図である。

【0076】

選択範囲管理テーブル133は、ユーザが特定のモジュール群を指定して集約処理を開始する際に集約する各モジュールの識別子を示す情報を格納する。また、登録日時を示す情報が格納されることで、集約処理の履歴としての情報が含まれても良い。選択範囲管理テーブル133は、選択範囲（集約モジュール）毎に、エントリを有する。各エントリは、選択範囲ID 801、モジュールID 802及び登録日時803といった情報を格納する。以下、一個の選択範囲を例に取る（図8の説明において「対象選択範囲」）。

10

【0077】

選択範囲ID801は、集約処理とそれを実施する際にユーザに選択されたモジュール群を一個に識別するための識別子（対象選択範囲の識別子）である。

【0078】

モジュールID802は、集約処理を実施する際にユーザに選択された（対象選択範囲に属する）各モジュールのモジュールIDを示す。

【0079】

登録日時803は、集約処理を開始した（対象選択範囲が指定された）日時を示す。なお、登録日時803が示す日時は、後からユーザが過去に実施された集約処理を確認するためのものであり、集約処理が実施されたことが判断できるのであれば開始日時でなくても良い。

20

【0080】

図9は、実施例1のモジュール比較テーブル134の一例を示す図である。

【0081】

モジュール比較テーブル134は、二つのモジュールを比較して関連性の度合いを記録しておくために利用するテーブルである。モジュール比較テーブル134は、比較グループ（二つのモジュール205の組）毎に、エントリを有する。各エントリは、比較グループID 901、比較グループ902、モジュール関連ポイント903、関連プロパティ904、プロパティ関連ポイント905、集約906、非集約907及び選択範囲ID 908といった情報を格納する。以下、一個の比較グループを例に取る（図9の説明において「対象比較グループ」）。

30

【0082】

比較グループID901は、対象比較グループを識別するための識別子を示す。

【0083】

比較グループ902は、対象比較グループに属する二つのモジュールの各々のモジュールIDを示す。

【0084】

モジュール関連ポイント903は、対象比較グループに属するモジュール同士の関連性の高さを示す数値であるモジュール関連ポイントを示す。本実施例では、上述したように、“モジュール比較”のポリシーを対象比較グループが満たした場合、そのポリシーに予め割り当てられているポイントが、対象比較グループのモジュール関連ポイントに加算される。例えば、モジュール関連ポイントの初期値は“0”であり、“モジュール比較”のポリシーが適合される都度に、そのポリシーに対応したポイントが加算されても良い。

40

【0085】

関連プロパティ904は、対象比較グループに属する二つのモジュールが有するそれぞれのプロパティのうち、特定のポリシーで評価した場合に関連性が高いプロパティ項目の組み合わせであって、後の処理でユーザにプロパティ共通化を提案するプロパティ項目の組み合わせを示す。

【0086】

プロパティ関連ポイント905は、対象比較グループに属するモジュールのプロパティ同士の関連性の高さを示す数値であるプロパティ関連ポイントを示す。本実施例では、上述し

50

たように、“プロパティ比較”又は“プロパティ”のポリシーを対象比較グループが満たした場合、そのポリシーに予め割り当てられているポイントが、対象比較グループのプロパティ関連ポイントに加算される。例えば、プロパティ関連ポイントの初期値は“0”であり、“プロパティ比較”又は“プロパティ”のポリシーが適合される都度に、そのポリシーに対応したポイントが加算されても良い。

【0087】

集約906は、対象比較グループに属する二つのモジュールのプロパティのうち、共通化されたプロパティ項目を示す。一方、非集約907は、対象比較グループに属する二つのモジュールのプロパティのうち、共通化されなかったプロパティ項目を示す。

【0088】

選択範囲ID 908は、対象比較グループが属する選択範囲の識別子である。

【0089】

図10は、実施例1のプロパティテーブル135の一例を示す図である。

【0090】

プロパティテーブル135は、モジュールのプロパティに対してポリシー管理テーブル132の種別が“プロパティ”のポリシーを適用した際にポリシーを満たしたか否かを登録しておくためのテーブルである。なお、本実施例では、集約処理中にプロパティテーブル135の生成が行われるが、予めプロパティに対してポリシー種別が“プロパティ”のポリシーが事前ポリシーとして登録されていても良い。

【0091】

プロパティテーブル135は、プロパティ要素毎に、エントリを有する。各エントリは、プロパティID 1001及び事前ポリシー1002といった情報を格納する。以下、一個のプロパティ要素を例に取る（図10の説明において「対象プロパティ要素」）。

【0092】

プロパティID1001は、対象プロパティ要素を識別するための識別子である。なお、当該識別子は、例えば、対象プロパティ要素におけるプロパティ項目（例えばプロパティ項目名）と、当該プロパティ項目を持つプロパティに関連したモジュールのモジュールIDとを含む。

【0093】

事前ポリシー1002は、対象プロパティ要素が満たす“プロパティ”のポリシーのポリシーIDを示す。

【0094】

図11は、実施例1の表示条件テーブル136の一例を示す図である。

【0095】

表示条件テーブル136は、ユーザにモジュールの集約処理をGUI（ワークスペース）で行わせる際に表示するモジュール数などの条件を格納する。表示条件テーブル136は、表示条件ごとにエントリを有する。各エントリは、表示条件ID 1101及び最大モジュール表示数902といった情報を格納する。以下、一個の表示条件を例に取る（図11の説明において「対象表示条件」）。

【0096】

表示条件ID1101は、対象表示条件を一意に識別するための識別子を示す。

【0097】

最大モジュール表示数902は、ワークスペース上に表示されるモジュールの数の上限である最大モジュール表示数を示す。最大モジュール表示数は、ユーザが設定できるようにしていても良いし、ユーザが利用する入力装置の画面サイズのようなユーザの利用環境の情報を利用してシステム（例えば制御部120）が自動的に値を設定しても良い。最大モジュール表示数が画面表示の制約の一例であるが、最大モジュール表示数は、画面サイズではなく、システムやユーザが任意に決めた値であっても良いし、設定ウィザード画面のアクティブウィンドウの表示倍率が動的に変更された場合に（拡大又は縮小が動的に行われた場合に）表示倍率に応じて最大モジュール表示数が動的に変更されても良い。また、モジ

10

20

30

40

50

ルール関連ポイントが一定値以上である全ての比較グループが表示されるようにモジュールの表示サイズが自動調整されても良い。

【0098】

図12は、実施例1の表示グループテーブル137の一例を示す図である。

【0099】

表示グループテーブル137は、ユーザにモジュールの集約処理をGUIで行わせる際に同時に表示する単位である表示グループに関する情報を格納する。表示グループテーブル137は、表示グループ毎にエントリを有する。各エントリは、表示グループID 1201、比較グループID 1202、共通化対象1203、表示条件1204及び選択範囲ID 1205といった情報を格納する。以下、一個の表示グループを例に取る（図12の説明において「対象表示グループ」）。

10

【0100】

表示グループID1201は、対象表示グループを一意に識別するための識別子を示す。

【0101】

比較グループID1202は、対象表示グループを構成する一個以上の比較グループの各々の識別子を示す。

【0102】

共通化対象1203は、対象表示グループを構成する比較グループの中でユーザが共通化を行うと判断した比較グループの比較グループIDを示す。

【0103】

表示条件1204は、対象表示グループの表示を行う際に適用する表示条件の表示条件IDを示す。例えば、表示グループID：DG2によれば、二個の比較グループが一画面に表示され得る。

20

【0104】

選択範囲ID 1205は、対象表示グループが対応する1回の集約処理に関連した選択範囲の選択範囲IDを示す。

【0105】

図13は、ポリシー登録部121の一例を説明するフローチャートである。なお、図13～17及び図20の「S」は、「ステップ」の略である。

【0106】

ポリシー登録部121は、集約モジュールのモジュールまたはプロパティが共通化できるかをユーザが判断するための候補のグループを生成するために利用するポリシーを登録する。

30

【0107】

具体的には、ポリシー登録部121は、管理サーバ101の入力装置104等からユーザのポリシー登録の開始要求を受け付ける（ステップ1301）。その開始要求に応答して、ポリシー登録部121は、管理サーバ101の入力装置104や表示装置105等を通じて、ユーザに新たに登録したいポリシーを設定させる（ステップ1302）。なお、ポリシー登録画面（例えばGUI）は図示しないが、ポリシー登録画面を介して入力されたポリシーの情報が入力される。ポリシー登録部121は、ユーザが入力したポリシー情報を、ポリシー管理テーブル132に登録する（ステップ1303）。

40

【0108】

なお、ポリシーは、前述の通りユーザからの入力でも良いし、予めシステムに登録されていても良い（予めポリシー管理テーブル132が用意されていても良い）。また、ポリシー登録部121は、新しく登録するポリシーが既に登録されていないか否かを判断するようにしても良いし、ユーザが新しく登録するポリシーと類似するポリシーが存在していないか否かをポリシーの一覧を見せることによって確認させても良い。また、ユーザからのポリシー情報の入力は、管理サーバ101以外にも管理ネットワーク103等のネットワークで接続された、ビジュアルプログラミングツール106や他の計算機から行われても良い。

【0109】

50

図14は、インフラ関連生成部126の一例を説明するフローチャートである。

【0110】

インフラ関連生成部126は、モジュールとそのモジュールが実行されるITインフラの情報を結びつける処理を行う。

【0111】

具体的には、インフラ関連生成部126は、ユーザがビジュアルプログラミングツール106の画面表示部141が表示したワークスペースでモジュールを配置してそれらを接続するなどのプログラミングが終わった後に、プログラムの実行命令がプログラム管理部142を通してプログラム実行部151に対して送信されたことを検知して処理を開始する（ステップ1401）。

10

【0112】

インフラ関連生成部126は、デプロイされるモジュールとそれがデプロイされる先のITインフラを一意に特定できる情報（インフラIDなど）をプログラム実行部151より取得する（ステップ1402）。

【0113】

なお、プログラム実行部151がプログラムの実行命令を受信した際にインフラ関連生成部126にプログラムを実行する際にプログラム内部の処理に関連するITインフラ情報が通知されても良い。

【0114】

もしくは、ITインフラ情報を取得するインフラ情報取得部125（処理に関するフローチャート等は省略）が、インフラ管理テーブル138の管理IPアドレス702及び認証情報703を基にITインフラから情報を直接取得するようにしても良い。その場合、情報取得方法としては、情報取得用エージェントを経由して取得する方法や、SSH（Secure Shell）のようなOSにデフォルトで導入されている通信手段を利用して情報を取得する方法や、IPMI（Intelligent Platform Management Interface）などのハードウェアから情報取得する方法などが考えられるが特に方法は限定しない。もしくは、管理サーバ101がITインフラの運用管理ソフトウェアやサービスと連携して、連携先のソフトウェアやサービスから取得した情報をインフラ管理テーブル138に登録するようにしても良い。また、ステップ1401のようにデプロイ実行命令の検知を契機に処理を開始するのではなく、管理サーバ101がスケジュールに合わせて定期的にITインフラの情報取得を行っても良いし、ITインフラの構成変更を検知した場合にITインフラの情報取得を行うのでも良い。

20

30

【0115】

インフラ関連生成部126は、インフラ管理テーブル138の関連モジュール706として、デプロイされるモジュールのモジュールIDを登録する（ステップ1403）。

【0116】

インフラ関連生成部126は、処理完了を管理サーバ101に通知する（ステップ1404）。

【0117】

インフラ関連生成部126により管理サーバ101はモジュールとITインフラの情報を結びつけて情報を管理することが可能となり、ポリシー管理テーブル132のポリシーID：MP3のポリシーのようなITインフラに関係する情報を利用するポリシーでモジュール間の関連性を評価することが可能になる。

40

【0118】

図15A～図15Cは、集約候補生成部122の一例を説明するフローチャートである。

【0119】

集約候補生成部122は、ユーザから選択されたモジュール群のそれぞれの関連性の高さ、及び、そのモジュール群に属する複数個のモジュールがそれぞれ持つプロパティの関連性の高さを評価する。これにより、ユーザがモジュールを集約した際にプロパティの共通化できるか否かを判断するための指標ができる。

【0120】

集約候補生成部122は、ユーザからモジュールの集約開始と集約対象とするモジュール群

50

の指定を検知する（ステップ1501）。ユーザがモジュール群を指定する際は、ビジュアルプログラミングツール106のワークスペース上でマウスをドラックすることによって矩形のボックスを生成し、それで覆われたモジュール群を選択範囲のモジュール群といったように指定できるようにしても良い。

【0121】

次に、集約候補生成部122は、選択範囲（ユーザが選択した範囲）にいる全モジュール205のモジュールIDをモジュールID 602として選択範囲管理テーブル133に保存する（ステップ1502）。

【0122】

次に、集約候補生成部122は、選択範囲にいる各モジュール205について、当該モジュール205のモジュールIDをキーに、当該モジュール205の情報を、モジュール管理テーブル131から取得する（ステップ1503）。

【0123】

次に、集約候補生成部122は、ステップ1503で取得したモジュール情報と、ポリシー管理テーブル132から取得したポリシー情報を用いて集約候補生成処理を実施する。

【0124】

集約候補生成処理では、選択範囲から全ての二個のモジュールの組み合わせを生成する（ステップ1504）。図8のようにモジュールIDがそれぞれMod1、Mod2、Mod3、Mod4といった4つのモジュールが選択されている場合は、Mod1-Mod2、Mod1-Mod3、Mod1-Mod4、Mod2-Mod3、Mod2-Mod4、Mod3-Mod4といった六つのペア（六つの比較グループ）が生成される。ただし、選択範囲内に一定数を超えるモジュールが存在する場合は、モジュールの組み合わせが膨大になるため、以降のステップ1508で述べる二個のモジュール間の接続関係とモジュール間の距離を表すホップ数などで関連性が高いモジュールの組み合わせを算出することにより、全てのモジュールの組み合わせを生成しないようにしても良い。

【0125】

次に、集約候補生成部122は、ステップ1504で生成した全比較グループに関する情報をモジュール比較テーブル134に登録する（ステップ1505）。ここでは、比較グループID 901、比較グループ902及び選択範囲ID 908として有効な値が登録され、その他の情報903～907は初期値又は無効値である。

【0126】

次に、集約候補生成部122は、ポリシー管理テーブル132から、“モジュール比較”のポリシーのうち最もポイント604が高いポリシーを取得（選択）する。図6の例によれば、ポリシーID：MP1の“（モジュール間ホップ数）== 1？”が取得される（ステップ1506）。このポリシーは、モジュールがワイヤで接続状態にありかつ隣接している（間に他のモジュールが存在しない）モジュール同士が否かを評価するために利用される。モジュール間の「ホップ数」とは、モジュール間のワイヤの数である。すなわち、モジュール間のワイヤの数が1であれば（言い換えれば、モジュール間に介在するモジュールの数が0であれば）、ホップ数は1である。

【0127】

次に、集約候補生成部122は、モジュール比較テーブル134から特定される比較グループそれぞれに対してループ処理（ループ1）を実施する（ステップ1507）。なお、ここでは、モジュール比較テーブル134には、上記六つの比較グループ（Mod1-Mod2、Mod1-Mod3、Mod1-Mod4、Mod2-Mod3、Mod2-Mod4、Mod3-Mod4）に関する情報が登録されているとする。以下、一個の比較グループを例に取る（ループ1の説明において「対象比較グループ」）。

【0128】

集約候補生成部122は、ステップ1506で予め取得しておいたポリシーID：MP1のポリシー“（モジュール間ホップ数）== 1？”を対象比較グループが満たすか否かを判断する（ステップ1508）。具体的には、集約候補生成部122は、対象比較グループに属する各モジ

10

20

30

40

50

ジュールのモジュールIDに対応した接続先ID 505にどちらか一方のモジュールIDが含まれているか否かを判断する。接続先ID505にどちらか一方のモジュールIDが含まれていれば、ホップ数が1である。

【0129】

ステップ1508の判断結果が真の場合は、処理がステップ1509に移動する。そうでない場合は、処理がステップ1510に移動する。

【0130】

ステップ1509では、集約候補生成部122は、対象比較グループがポリシーID：MP1のポリシーを満たしたため、対象比較グループに対応したモジュール関連ポイント903に、ポリシーID：MP1に対応したポイント“5”を加算する。

10

【0131】

全ての比較グループに対して処理が完了した場合は、集約候補生成部122は、ステップ1507から開始したループ処理（ループ1）を終了する（ステップ1510）。

【0132】

次に、集約候補生成部122は、“モジュール比較”のポリシーのうち次にポイントが高いポリシーを取得する（ステップ1511）。図6の例によれば、ポリシーID：MP2の“（モジュール間距離） 200pixel？”が取得される。このポリシーは、モジュールがビジュアルプログラミングツールのワークスペース上で近くに配置されているか否かを評価するために利用する。これは、モジュールの位置が近くに配置されている場合、同時に利用される可能性が高いと仮定した時に設定するポリシーである。処理フローのトポロジ的には遠い（例えばホップ数が多い）モジュール同士であっても、それらをワークスペース上において近接して配置することがユーザにとって好ましいことがあり、また、そのようなモジュール同士はまとまりとして何らかの意味がある可能性が高い。本実施例では、ポリシーID：MP2のようなポリシーにより、ワークスペースにおいて近い距離にあるモジュール同士は、関連性の高いモジュール同士であると評価され得る。

20

【0133】

次に、集約候補生成部122は、モジュール関連ポイント903がポイント閾値“5”以上の比較グループに対してループ処理（ループ2）を開始する（ステップ1512）。以下、一個の比較グループを例に取る（ループ2の説明において「対象比較グループ」）。なお、ポイント閾値“5”とは、ステップ1509にてポリシーID：MP1での評価で付加されたポイントを指しており、“5”を閾値とするのはあくまでも一例である。また、評価する比較グループが増加するが、モジュール比較テーブル134の比較グループ全てに対してループ処理を行っても良い。

30

【0134】

集約候補生成部122は、ステップ1511で予め取得しておいたポリシーID：MP2のポリシー“（モジュール間距離） 200pixel？”を対象比較グループが満たすか否かを判断する（ステップ1513）。具体的には、集約候補生成部122は、対象比較グループの各モジュールのモジュールIDに対応したX座標及びY座標を取得する。対象比較グループがMod1-Mod2だとすると、x_Mod1（Mod1のx座標）、y_Mod1（Mod1のy座標）、x_Mod2（Mod2のx座標）、y_Mod2（Mod2のy座標）の値が取得できる。このため、モジュールMod1とモジュールMod2間の距離は、下記である。

40

（モジュールMod1とモジュールMod2間の距離）

$$= \sqrt{((x_Mod1(\text{Mod1のx座標}) - x_Mod2(\text{Mod2のx座標}))^2 + (y_Mod1(\text{Mod1のy座標}) - y_Mod2(\text{Mod2のy座標}))^2)}$$

【0135】

図5の例によれば、モジュールMod1とモジュールMod2間の距離は、182pixelであり、200pixel未満である。このため、対象比較グループが、ポリシーID：MP2のポリシーを満たすことになる。

【0136】

ステップ1513の判断結果が真の場合は、処理がステップ1514に移動する。そうでない場

50

合は、処理がステップ1515に移動する。

【0137】

ステップ1514では、集約候補生成部122は、対象比較グループがポリシーID：MP2のポリシーを満たしたため、対象比較グループに対応したモジュール関連ポイント903に、ポリシーID：MP2に対応したポイント“4”を加算する。

【0138】

ここで全ての比較グループに対して処理が完了した場合は、集約候補生成部122は、ステップ1512から開始したループ処理（ループ2）を終了する（ステップ1515）。

【0139】

次に、集約候補生成部122は、“モジュール比較”のポリシーのうち次にポイントが高いポリシーを取得する（ステップ1516）。図6の例によれば、ポリシーID：MP3の“同一ホスト上にデプロイされている？”を取得される（ステップ1516）。このポリシーは、モジュールが同一のホスト上で稼働するか否かを判断して、同一ホスト上に存在している場合は関連性が高いと判断するために利用される。

10

【0140】

次に、集約候補生成部122は、モジュール関連ポイント903がポイント閾値“5”以上の比較グループに対してループ処理（ループ3）を開始する（ステップ1517）。ポイント閾値の“5”はあくまでも一例である。例えば、ループ処理を行う対象の比較グループを計算量削減のために更に制限したい場合は、ステップ1514にてポリシーID：MP2での評価で付加されたポイント“4”を考慮して、モジュール関連ポイント903が“9”以上の比較グループに対してループ処理が行われても良い。また、評価する比較グループが増加するが、モジュール比較テーブル134の比較グループ全てに対してループ処理が行われても良い。以下、一つの比較グループを例に取る（ループ3の説明において「対象比較グループ」）。

20

【0141】

集約候補生成部122は、ステップ1516で予め取得しておいたポリシーID：MP3のポリシー“同一ホスト上にデプロイされている？”を対象比較グループが満たすか否かを判断する（ステップ1518）。具体的には、集約候補生成部122は、インフラ管理テーブル138を参照し、関連モジュール706に対象比較グループに属する二個のモジュールのモジュールIDが含まれているか否かを判断する。

【0142】

ステップ1518の判断結果が真の場合は、処理がステップ1519に移動する。そうでない場合は、処理がステップ1520に移動する。

30

【0143】

ステップ1519では、集約候補生成部122は、対象比較グループがポリシーID：MP3のポリシーを満たしたため、対象比較グループに対応したモジュール関連ポイント903に、ポリシーID：MP3に対応したポイント“3”を加算する。

【0144】

ここで全ての比較グループに対して処理が完了した場合は、集約候補生成部122は、ステップ1517から開始したループ処理（ループ3）を終了する（ステップ1520）。

【0145】

なお、図示は行わないが、モジュール間の関連性を前述のモジュールの接続関係、位置情報及びITインフラ情報からだけでなく、別の観点から評価する方法として、ステップ1506からステップ1520の間でITインフラに関係する情報を利用するポリシーを適用する処理が挿入されても良い。

40

【0146】

例えば、図6の例によれば、ポリシーID：MP4のポリシーは、モジュールを含んだソースコード（ビジュアルプログラミングにおいては、プロパティ情報を含んだモジュールやモジュールを接続するワイヤなどの情報）をユーザが最初にgitなどのソースコード管理システム（図示せず）に登録した際にソースコード管理システムにより発行されたコミットIDをモジュール間で比較し、コミットIDが一致した場合は、同時期に開発されたモジュール

50

であると判断可能であり、ポリシーID：MP4に対応したポイントがモジュール関連ポイント903に加算される。本実施例では、モジュール管理テーブル131に作成コミット507としてコミットIDが格納される。

【0147】

また、図6の例によれば、ポリシーID：MP5のポリシーは、モジュールを含んだソースコードを編集して更新した際に前述のポリシーID：MP4のポリシーの場合と同様にgitなどのソースコード管理システムから発行されたコミットIDをモジュール間で比較し、コミットIDが一致した場合は、同時期に変更が行われたモジュールであると判断可能であり、ポリシーID：MP5に対応したポイントがモジュール関連ポイント903に加算される。本実施例では、モジュール管理テーブル131の更新コミット508としてコミットIDが格納される。

10

【0148】

また、図6の例によれば、ポリシーID：MP6のポリシーは、モジュールを含んだソースコードをgitなどのソースコード管理システムに登録した際に、そのユーザの情報をモジュール管理テーブル131にAuthorとして登録しておく（本実施例のモジュール管理テーブル131では省略）。その情報を基にモジュール間で比較を行い、Authorが一致した場合は、同一のユーザ（モジュール開発者）が開発に携わっている可能性が高く、その設定プロパティも一致する可能性が高いということが想定される。そのような場合、ポリシーID：MP6に対応したポイントがモジュール関連ポイント903に加算される。

【0149】

また、図6の例によれば、ポリシーID：MP7のポリシーは、データ取得、データ解析、データ保存など再利用するために過去に作成したモジュールをカテゴリ分けしておき、それをgit等のソースコード管理システムなどに登録しておく。それを再利用して新たなアプリケーションプログラムを作る場合において、カテゴリを識別する情報をモジュール管理テーブル131にカテゴリ509として登録しておく。その情報を基にモジュール間で比較を行う。カテゴリが一致した場合は、ポリシーID：MP7に対応したポイントがモジュール関連ポイント903に加算される。

20

【0150】

ここからは、集約候補生成部122は、モジュール内のプロパティ同士の関連性を評価する。このプロパティ同士の関連性の高さについてポリシーを用いて評価しておくことで、のちにユーザに共通化できる可能性が高いプロパティ群として提示するために利用する。

30

【0151】

集約候補生成部122は、ポリシー管理テーブル132から、種別602==“プロパティ”及びポリシー種別==“プロパティ比較”のポリシーを取得する（ステップ1521）。

【0152】

集約候補生成部122は、モジュール比較テーブル134から、比較グループの情報を取得する（ステップ1522）。集約候補生成部122は、このステップで取得した各比較グループに属する二個モジュールがそれぞれ持つプロパティについて共通化できる可能性があるものを検出する。ちなみに、この比較グループの情報を取得する際、全ての比較グループに対して処理を行うと膨大な計算量になる可能性があるため、比較グループのうちモジュール関連ポイント903が低い（例えば、相対的に低い又は一定値未満である）ものについては、集約候補生成部122は、この後のプロパティの評価の対象外としても良い。

40

【0153】

次に、集約候補生成部122は、モジュール比較テーブル134の比較グループそれぞれに対してループ処理（ループ4）を実施する（ステップ1523）。以下、一個の比較グループを例に取る（ループ4の説明において「対象比較グループ」）。なお、このループ処理の時点でモジュール関連ポイント903が低い比較グループを除外することで計算量を減らしても良い。

【0154】

集約候補生成部122は、対象比較グループを構成する二個のモジュールの各プロパティ要

50

素を全てプロパティテーブル135に登録する（ステップ1524）。

【0155】

次に、集約候補生成部122は、ステップ1524でプロパティテーブル135に登録した全プロパティ要素に対して、ポリシー管理テーブル132のうちの“プロパティ”のポリシーを全て適用していくループ処理（ループ5）を開始する（ステップ1525）。以下、一個のプロパティ要素を例に取る（ループ5の説明において「対象プロパティ要素」）。なお、今回は各プロパティ要素に対してポリシーを適用していくが、集約候補生成部122は、特定のポリシーとプロパティ要素に対してはポリシー適用を避けたり、プロパティ要素が特定のポリシーを満たした場合は他のポリシーの適用を避けたりするといったような制限をつけておくことで全てのポリシーを各プロパティ要素に適用するのを避けても良い。また、集約処理のタイミングではなく定期的、もしくは新たなポリシー又はプロパティ要素が登録されたタイミングなどで、ポリシーがプロパティ要素に適用され、結果として、ループ処理（ループ5）が省略されても良い。

10

【0156】

集約候補生成部122は、対象プロパティ要素に対して種別602が“プロパティ”のポリシーを適用するループ処理を開始する（ステップ1526）。以下、“プロパティ”のポリシーのうちの一個のポリシーを例に取る（ループ6の説明において「対象ポリシー」）。

【0157】

集約候補生成部122は、対象プロパティ要素に対して対象ポリシーを適用する（ステップ1527）。集約候補生成部122は、対象プロパティ要素が対象ポリシーを満たすか否かを判断する（ステップ1528）。

20

【0158】

ステップ1528の判断結果が真の場合は、処理がステップ1529に移動し、そうでない場合は、処理がステップ1530に移動する。

【0159】

例えば、図5を参照すると、対象プロパティ要素が、モジュールMod2のプロパティ要素“IP.Mod2”であれば、プロパティ値は“1.1.1.1”である。図6を参照すると、対象ポリシーが、ポリシーID：PP1のポリシー（IP形式であるかを検出するポリシー）であれば、対象プロパティ要素が対象ポリシーを満たす。このため、処理が、ステップ1529に移動することになる。しかし、対象ポリシーが、ポリシーID：PP2のポリシー（Port形式であるかを検出するポリシー、一例としては、ポート番号の最小値0からポート番号の最大値65535の間の数値に当てはまるか否かを検出する）であれば、対象プロパティ要素は対象ポリシーを満たさないため、処理が、ステップ1530に移動する。

30

【0160】

種別602が“プロパティ”のポリシーについて対象プロパティへの適用を行い、ステップ1526からのループ処理（ループ6）が終了する（ステップ1530）。

【0161】

プロパティテーブル135のプロパティ要素全てに対して処理が完了したら、ステップ1525からのループ処理（ループ5）が終了し（ステップ1531）、処理が、ステップ1532に移動する。

40

【0162】

次に、集約候補生成部122は、対象比較グループを構成する二つのモジュールがそれぞれ有する二つのプロパティから、二つのプロパティ要素の組み合わせを作成する（ステップ1527）。対象比較グループにおける一方のプロパティ要素は、一方のモジュールのプロパティにおけるいずれかのプロパティ要素である。対象比較グループにおける他方のプロパティ要素は、他方のモジュールのプロパティにおけるいずれかのプロパティ要素である。本実施例では、モジュールMod2とモジュールMod3の組み合わせの場合は、{検出パターン.Mod1,IP.Mod2},{検出パターン.Mod1,Port.Mod2},{検出パターン.Mod1,User.Mod2},{検出パターン.Mod1>Password.Mod2}というような形で組み合わせが生成される。なお、この後の処理量にプロパティの組み合わせの数が影響を与えるため、特定のプロ

50

パーティ同士は組み合わせをしないように設定しておいても良い。例えば、データ型が一致しないものは組み合わせとしては除外されても良い。

【0163】

次に、集約候補生成部122は、プロパティ要素の組み合わせ全てに対して処理が完了するまでループ処理（ループ7）を行う（ステップ1533）。以下、プロパティ要素の一個の組み合わせ（ペア）を例に取る（ループ7の説明において「対象組み合わせ」）。

【0164】

集約候補生成部122は、対象組み合わせに属する各プロパティ要素に対して、ポリシー管理テーブル132のうち種別602が“プロパティ比較”のポリシーを全て適用していくループ処理（ループ8）を開始する（ステップ1534）。以下、“プロパティ比較”のポリシーのうち

10

【0165】

集約候補生成部122は、対象ポリシーについて事前ポリシー605が設定されているか否かを判断する（ステップ1535）。ステップ1535の判断結果が真の場合は、処理がステップ1536に移動する。そうでない場合は、処理がステップ1538に移動する。

【0166】

ステップ1536では、集約候補生成部122は、対象組み合わせを構成する二個のプロパティのそれぞれについて事前ポリシーを満たしているか否かの情報を取得する。取得する情報は、プロパティテーブル135の事前ポリシー1002の情報である。

【0167】

次に、集約候補生成部122は、ポリシー管理テーブル132の事前ポリシー605（対象ポリシーに対応した事前ポリシー605）の情報とステップ1536で取得した事前ポリシー1002の情報を基に、次のステップでプロパティの組み合わせにポリシーを適用すべきか否かを判断する（ステップ1537）。ここでの処理は、対象ポリシーが、設定している前提条件を満たしているか否かを事前ポリシーの結果から判断する処理である。例えば、ポリシーID：PPC1のポリシー（IP形式の二個のプロパティ値が一致しているか否かを判断するポリシー）には、事前ポリシー605としてポリシーID：PP1（IP形式であるかを検出するポリシー）が設定されている。これは、ポリシーID：PPC1のポリシーを適用する対象組み合わせを構成する二個のプロパティ要素の各々は、ポリシーID：PP1を満たすプロパティ要素であるべきことを意味している。プロパティ要素の組み合わせ{検出パターン.Mod1, IP.Mod2}に対してポリシーID：PPC1のポリシーの適用を考えた場合のステップ1537の処理は、ポリシーID：PPC1の事前ポリシーにあたるポリシーID：PP1のポリシーの適用結果が、検出パターン.Mod1はポリシーを満たしていない（プロパティテーブル135の事前ポリシー1002が“PP1”を含んでいない）ため、ポリシーID：PPC1のポリシーの適用が行われない。仮に、対象組み合わせが{IP.Mod1, IP.Mod4}であり対象ポリシーがポリシーID：PPC1のポリシーの場合のステップ1537の処理は、ポリシーID：PPC1の事前ポリシー605にあたるポリシーID：PP1が、IP.Mod1とIP.Mod4の共にポリシーを満たしている（プロパティテーブル135の事前ポリシー1002が“PP1”を含んでいる）ため、ポリシーID：PPC1のポリシーの適用が行われる。

20

30

【0168】

次に、集約候補生成部122は、対象ポリシーを対象組み合わせが満たすか否かを判断する（ステップ1538）。ステップ1538の判断結果が真の場合は、処理がステップ1539に移動する。そうでない場合は、処理がステップ1540に移動する。例えば、対象ポリシーが、ポリシーID：PPC1のポリシー（IP形式の二個のプロパティ値が一致しているか否かを判断するポリシー）であり、対象組み合わせが、{IP.Mod2（プロパティ値が1.1.1.1）, IP.Mod3（プロパティ値が1.1.1.1）}である場合は（つまり、プロパティ値が同じである場合は）、対象組み合わせが対象ポリシーを満たすので、処理がステップ1539に移動する。仮に、対象ポリシーが、ポリシーID：PPC2のポリシー（Port形式の二個のプロパティ値が一致しているか否かを判断するポリシー）であり、対象組み合わせが、{Port.Mod2（プロパティ値が8000）, Port.Mod3（プロパティ値が8080）}である場合は、対象組み

40

50

合わせが対象ポリシーを満たさないため、処理がステップ1540に移動する。

【0169】

ステップ1538で対象ポリシーが満たされた場合は、集約候補生成部122は、対象比較グループのプロパティ関連ポイント905に、対象ポリシーのポイント（ポリシー管理テーブル132のポイント604の数値）を加算する（ステップ1539）。なお、対象比較グループのプロパティ関連ポイント905には、更に、対象ポリシーの事前ポリシー605としてのポリシーIDに対応したポイントも加算されても良い。

【0170】

種別602が“プロパティ比較”のポリシーを全てプロパティの組み合わせに適用が完了した場合、ステップ1535から開始したループ処理（ループ8）が終了する（ステップ1540）

10

【0171】

プロパティ要素の組み合わせ全てに対して処理が完了した場合、ステップ1533から開始したループ処理（ループ7）が終了する（ステップ1541）。

【0172】

モジュール比較テーブル134の比較グループ全てに対して処理が完了した場合、ステップ1523から開始したループ処理（ループ4）が終了する（ステップ1542）。

【0173】

次に、集約候補生成部122は、表示グループ生成部123に処理開始を通知して、一連の処理を完了する（ステップ1543）。なお、ステップ1543（又はそれよも前の段階で）、集約候補生成部122は、比較グループについて、プロパティ要素の組み合わせのうち、適用された“プロパティ比較”のポリシーを満たした組み合わせを表す情報が、関連プロパティ904の少なくとも一部として、モジュール比較テーブル134に登録されて良い。

20

【0174】

図16は、表示グループ生成部123の一例を説明するフローチャートである。

【0175】

表示グループ生成部123は、ユーザが複数個のモジュールを集約して集約モジュールを生成する過程において、共通化できるプロパティ要素の候補（ビジュアルプログラミングツール106の画面表示部141が表示するGUIにおけるモジュールのグループ（表示グループ））を決定する。

30

【0176】

基本的には、モジュール関連ポイントが高いモジュールを可能な限り一個の画面に表示するようにすることでユーザが表示されたモジュールの中のプロパティの共通化の判断しやすいようにする。

【0177】

具体的には、表示グループ生成部123は、集約候補生成部122からの処理開始命令を検知して処理を開始する。（ステップ1601）。

【0178】

表示グループ生成部123は、表示条件テーブル136から表示条件（最大モジュール表示数）を取得する（ステップ1602）。表示グループ生成部123は、表示条件テーブル136に複数の表示条件が存在する場合は、ユーザから集約処理を開始する時点でユーザ所望の表示条件の選択を受け付けたり、デフォルトの表示条件を自動で選択したり、ユーザが利用する入力装置の環境情報を基に自動で表示条件を選択したりしても良い。

40

【0179】

表示グループ生成部123は、モジュール管理テーブル131から現在の集約処理に該当する選択範囲に属する各モジュールのモジュール情報（例えばエントリにおける全情報）を取得する（ステップ1603）。

【0180】

表示グループ生成部123は、モジュール比較テーブル134から現在の集約処理に該当する選択範囲に属する各比較グループの情報（例えばエントリにおける全情報）を取得する（

50

ステップ1604)。

【0181】

表示グループ生成部123は、現在の集約処理に該当する選択範囲に属する全てのモジュールに対してループ処理(ループ1)を開始する(ステップ1605)。ループ処理を行う際のモジュールの処理順番は、モジュール管理テーブル131の表示順510(番号が小さい順)で良い。以下、一つのモジュールを例に取る(ループ1の説明において「対象モジュール」)。

【0182】

表示グループ生成部123は、対象モジュールを構成要素として含む比較グループを、ステップ1604で取得した情報を基に抽出する(ステップ1606)。対象モジュールがモジュールMod1の場合は、比較グループは、[Mod1,Mod2]、[Mod1,Mod3]及び[Mod1,Mod4]である。表示グループ生成部123は、この時点で、例えばモジュール関連ポイントが一定値(例えば“0”)以下の[Mod1,Mod3]を、ステップ1603で抽出される比較グループから除外してもよい。つまり、表示グループ生成部123は、モジュールを同時に表示して集約処理を行う必要性が低い比較グループを除外しておいても良い。

【0183】

次に、表示グループ生成部123は、表示されるモジュールの数がステップ1602で取得した最大モジュール表示数以下になるように、比較グループから表示グループを生成する(ステップ1607)。ここで生成された表示グループを、この段落において「対象表示グループ」と言う。例えば、表示グループ生成部123は、最大モジュール表示数が“3”の場合は、対象モジュールMod1以外に二つのモジュールを表示する余裕があるため、モジュール関連ポイントが相対的に高い比較グループ[Mod1,Mod2]及び[Mod1,Mod4]を対象表示グループとして構成する。この際、表示グループ生成部123は、対象表示グループに入らない比較グループ[Mod1,Mod3]を、別の表示グループを作成してそこに格納しても良いし、関連性が低い比較グループとして表示グループを作成しなくても良い。ステップ1607において、表示グループ生成部123は、生成した表示グループに関する情報1201~1205を、表示グループテーブル137に登録する。表示グループID1201は、表示グループ生成部123により決定されたID(又はユーザから指定されたID)である。表示グループID1201は、対象表示グループについて表示グループ生成部123により決定されたID(又はユーザから指定されたID)を示す。比較グループ1202は、対象表示グループに含まれる各比較グループのIDを示す。共通化対象1203は、この時点では無効値“-”である。表示条件1204は、ステップ1604で取得された表示条件を示す。選択範囲ID 1206は、現在の集約処理に該当する選択範囲のIDを示す。尚、ステップ1607では、モジュール関連ポイントが一定値以上である比較グループが選択されても良いし、モジュール関連ポイントに加えてプロパティ関連ポイントが相対的に又は一定値以上に高い比較グループが選択されても良い。

【0184】

現在の集約処理に該当する選択範囲に属する全てのモジュールに対して処理が行われた場合、表示グループ生成部123は、ループ処理(ループ1)を完了する(ステップ1608)。

【0185】

次に、表示グループ生成部123は、集約実行部124に処理開始を通知する(ステップ1609)。

【0186】

図17は、集約実行部124の一例を説明するフローチャートである。

【0187】

集約実行部124は、ユーザに比較グループ(共通化するプロパティの候補)を、ビジュアルプログラミングツール106の画面表示部141により表示されるGUI(ワークスペース)上に表示することで、ユーザにプロパティ要素を共通化するか否かを判断させる。そして、ユーザからプロパティ要素の共通化の選択を受け付けた場合、集約実行部124は、当該共通化されるプロパティ要素を、集約モジュールについてのプロパティ要素として設定す

10

20

30

40

50

る。

【0188】

具体的には、集約実行部124は、集約実行部124からの処理開始を検知する（ステップ1701）。

【0189】

集約実行部124は、表示グループテーブル137から全ての表示グループの情報（例えばエントリにおける全ての情報）を取得する（ステップ1702）。

【0190】

集約実行部124は、全ての表示グループに対してループ処理（ループ1）を開始する（ステップ1703）。以下、一個の表示グループを例に取る（ループ1の説明において「対象表示グループ」）。

10

【0191】

集約実行部124は、GUI（ワークスペース）に、対象表示グループに含まれた各モジュールを表示する（ステップ1704）。ステップ1704の結果の一例を、図18が示す。なお、モジュールの表示とは、当該モジュールを表す表示オブジェクト（例えば角丸の矩形であるモジュールアイコン）の表示に加えて、当該モジュールに関する詳細情報の少なくとも一部の表示であっても良い。ここで言う「詳細情報の少なくとも一部」とは、当該モジュールについてモジュール関連テーブル131から取得される少なくとも一部の情報と、当該モジュールが属する比較グループについてモジュール比較テーブル134から取得される少なくとも一部の情報と、当該モジュール及び当該比較グループの各々が満たすポリシーに関する少なくとも一部の情報とのうちの少なくとも一部で良い。

20

【0192】

ユーザは、表示された比較グループ（プロパティ要素の組）に対して共通化するか否かを判断し、判断結果を、入力装置104を用いてGUIに対して入力する。この際、共通化すると判断された比較グループがあった場合、集約実行部124は、表示グループテーブル137の共通化対象1203としてその比較グループのIDを登録する（ステップ1705）。ユーザが、もし前の表示グループに遡って修正する場合は、前表示グループの表示画面に戻っても良い。

【0193】

集約実行部124は、全表示グループに対して処理が完了したらループ処理（ループ1）を終わりにする（ステップ1706）。

30

【0194】

集約実行部124は、図19のような画面（例えばGUI）、すなわち、集約モジュールと共通化するプロパティ要素との一覧としての確認画面を表示する（ステップ1707）。言い換えれば、集約実行部124は、ユーザに、集約モジュールに関する情報を確認させる。

【0195】

集約実行部124は、ユーザが集約モジュール及び共通化するプロパティの確認を終えたことを検知する（ステップ1708）。ステップ1708の検知は、例えば、ユーザから確認終了の入力の受け付けである。もし、ユーザが、共通化するプロパティ要素を変更したい場合は、表示グループを表示する画面に戻れるようにして、共通化の判断をやり直させが可能であっても良い。

40

【0196】

図18は、ユーザに対して共通化するプロパティを表示するGUIであるプロパティ共通化画面の一例を示している。

【0197】

プロパティ共通化画面1800は、ユーザが複数個のモジュールを集約する際、集約するモジュールのプロパティ要素に共通化できる可能性があるものを制御部120が検知した場合に、その情報を提示するために用いられる。また、提示した情報を基に共通化するプロパティ要素をユーザが選択するためにもプロパティ共通化画面1800は用いられる。

【0198】

50

プロパティ共通化画面1800は、ビジュアルプログラミングツール106の画面表示部141により表示される。

【0199】

プロパティ共通化画面1800は、共通化可能と制御部120により判断されたプロパティ要素の組を表示する。プロパティ共通化画面1800は、モジュールアイコン1801、ワイヤアイコン1802、プロパティ要素リスト1803、共通化レコメンデーション1804、説明ボックス1805、モジュール関連領域1806、及び、ボタン1807～1809を表示する。

【0200】

モジュールアイコン1801は、モジュールをビジュアルプログラミングツール上で表現する際のアイコンである。図示の例では、“エラーチケット登録”及び“ログ登録”の二つのモジュールのモジュールアイコン1801C及び1802Dが表示されている。これらはワイヤアイコン1802で接続されている。なお、図示の例では、二つのモジュールのモジュールアイコン1801C及び1802Dが表示されているが、採用されている表示条件（最大モジュール表示数）の範囲で三つ以上のモジュールが一画面に表示されても良い。また、一画面に表示されるモジュール（モジュールアイコン1801）の数は、表示倍率に応じて変更されても良い。

【0201】

ワイヤアイコン1802は、モジュール同士を接続するワイヤを示すアイコンであり、モジュール間関係を視覚的に判断できるようにするために用意されたGUI部品である。

【0202】

プロパティ要素リスト1803は、モジュール毎に表示され、当該モジュールに属するプロパティ要素のリストである。

【0203】

なお、モジュールアイコン1801及びワイヤアイコン1802は、モジュール管理テーブル131の各モジュールについての接続先ID505を基に表示されて良い。また、モジュールアイコン1801及びワイヤアイコン1802は、モジュール管理テーブル131の各モジュールについての接続先ID505（又は、ビジュアルプログラミングツール106が管理するフロー情報（各モジュールの接続関係に関する情報）を基にビジュアルプログラミングツール106により表示されて良い。

【0204】

共通化レコメンデーション1804は、共通化可能と制御部120により判断されたプロパティ要素の組の提示に相当する。図示の例では、共通化レコメンデーション1804は、プロパティ要素を囲むボックスとそれらを接続する線で構成される。また、図示の例では、プロパティ要素を囲むボックス（枠線）とそれらを接続する線として、実線と破線があるが、破線は、共通化可能とのレコメンデーションであり、実線は、ユーザにより承認されたレコメンデーション（つまり、ユーザから対象のプロパティ要素の組を共通化すると判断（選択）されたレコメンデーション）を示している。ユーザは、例えば、レコメンデーションとしてのプロパティ要素の組の指定（例えばクリック）を1回以上行うことで、実線（共通化する）と破線（共通化しない）を切り替える、つまり、共通化するか否かを指定（入力）する。

【0205】

説明ボックス1805は、共通化レコメンデーション1804の理由、すなわち、共通化可能と制御部120により判断された理由を表示する。ユーザは、説明ボックス1805に記述の説明を参考にして、共通化レコメンデーション1804の対象のプロパティ要素の組を共通化するか否かを判断できる。説明ボックス1805は、共通化レコメンデーション1804の対象の比較グループに対応したモジュール関連ポイント903が示す値を提示している。なお、図示の例では、説明ボックス1805は、予め共通化レコメンデーション1804毎に理由を示しているが、それに代えて、共通化レコメンデーション1804がマウスオーバーされた際に当該共通化レコメンデーション1804についての説明ボックス1805が表示されても良い。

。

10

20

30

40

50

【0206】

モジュール関連領域1806は、説明ボックス1805が説明する範囲を視覚的に示したものである。図示の例では、二つのモジュールが表示されているため、モジュール関連領域1806が無くても説明ボックス1805が説明している範囲のモジュールは判断できるが、三個以上のモジュールが画面1800に表示されている場合には、モジュール関連領域1806が無いと、説明ボックス1805がカバーする範囲の判断が困難になり得るため、モジュール関連領域1806は有用である。

【0207】

ボタン1807は、現在表示されているプロパティ要素リストについて共通化の判断が終わり、次のプロパティ共通化の判断に移る時にユーザが押下するボタンである。共通化の判断が必要なプロパティが存在しない場合は、図19の確認画面に遷移する。

10

【0208】

ボタン1808は、現在より前のプロパティ共通化の画面に戻る時にユーザが押下するボタンである。これはユーザが一度共通化しようとしたプロパティについて再考する際などに利用される。なお、最初のプロパティ共通化画面の場合は、前のプロパティ共通化画面が存在しないため押下しても画面が遷移しなくても良いし、ボタン1808が非表示とされても良い。

【0209】

ボタン1809は、集約処理を途中でキャンセルするためにユーザが押下するボタンである。これによりユーザは途中でモジュールの集約を中止することができる。

20

【0210】

また、本実施例では、画面を区切って複数の画面を切り替えるように記載しているが、例えば、集約する選択範囲全体を表示しておいて、一部の箇所を拡大するようにしておくようなGUIが採用されも良い。その場合は、ボタン1807やボタン1808での画面切り替えではなく、ユーザがマウスポインタでドラックするなどして画面の切り替えが行われて良い。

【0211】

図19は、集約モジュールのプロパティ要素の一覧を表示するGUIである確認画面の一例を示している。

【0212】

確認画面1901は、ユーザが複数個のモジュールを集約する際、集約するモジュールのプロパティの共通化の判断を全て終えた後の最終確認のための画面である。確認画面1901は、ビジュアルプログラミングツール106の画面表示部141により表示される。

30

【0213】

確認画面1901は、図18の画面1800でプロパティ要素の共通化の判断を行った結果の一覧である。

【0214】

確認画面1901は、一覧ボックス1902を有する。一覧ボックス1902は、テキストボックス1903、共通プロパティ要素リスト1904及び個別プロパティ要素リスト1905を有する。また、確認画面1901は、ボタン1906～1908を有する。

【0215】

テキストボックス1903は、集約モジュールのモジュール名を表示する。テキストボックス1903経由で集約モジュールのモジュール名が変更できて良いし、テキストボックス1903は最初空白でありここで初めてモジュール名がユーザにより入力されても良い。

40

【0216】

共通プロパティ要素リスト1904は、ユーザが共通化すると判断したプロパティ要素のリストである。共通化したプロパティ要素は、共通化する前のプロパティ要素が属していたモジュールが判断できるようカテゴライズされても良い。本実施例では、共通化するプロパティ要素が“エラーチケット登録”のモジュールと“ログ登録”のモジュールの各々に属している場合は、共通化するプロパティ要素は、“[共通]エラーチケット登録/ログ登録”といったカテゴライズがされて良い。また、本実施例では、共通化するプロパティ要素におけ

50

るプロパティ値（IPプロパティの場合は、例えば“1.1.1.1”）も一緒に表示されているが、プロパティ要素のうち、プロパティ値は表示されず、共通化するプロパティ要素におけるプロパティ項目が表示されても良い。

【0217】

個別プロパティ要素リスト1905は、プロパティ要素を共通化する処理の中で、共通化すると判断されたプロパティ要素以外のプロパティ要素のリストである。モジュールに個別に属するため、モジュールごとに個別プロパティ要素リスト1905が作成される。本実施例では、代表として“エラーチケット登録”のモジュールと“ログ登録”のモジュールについて、それぞれ共通化されなかったプロパティ要素をモジュールごとにリストで表示される。

【0218】

ボタン1906は、表示されているモジュール及びプロパティの設定で集約モジュールを生成して良いとユーザが判断した際に押下するボタンである。

【0219】

ボタン1907は、集約処理を途中でキャンセルするために利用するボタンである。これによりユーザは途中でモジュールの集約を中止することができる。

【0220】

ボタン1908は、図18のプロパティ共通化画面1800に戻る時にユーザが押下するボタンである。

【0221】

実施例1によれば、前述した処理を実行することによって、ユーザは集約モジュールのプロパティの数を削減することができ、結果として、ユーザの集約モジュールに対する設定工数が減少する。なお、集約モジュールは、一個のモジュール相当であるため、制御部120により、集約モジュールの情報は、モジュール管理テーブル131に登録される。また、制御部120により、集約モジュールについて共通化されたプロパティ要素の情報は、モジュール比較テーブル134に集約906として登録され、集約モジュールについて共通化されなかったプロパティ要素の情報は、モジュール比較テーブル134に非集約907として登録される。

【実施例2】

【0222】

実施例2を説明する。その際、実施例1との相違点を主に説明し、実施例1との共通点については説明を省略又は簡略する。

【0223】

実施例2では、集約モジュールを構成するモジュールの仕様変更などを行う必要が生じた場合、集約モジュールのプロパティをユーザに必要最低限の判断で再設定させることができる。

【0224】

図20は、本発明の実施例2における変更実行部の一例を説明するフローチャートである。

【0225】

変更実行部2000は、制御部120に追加された機能であり、変更する必要が生じたモジュールの情報を基に集約モジュールのプロパティを変更するための表示グループを特定する。

【0226】

具体的には、変更実行部2000は、ユーザから集約モジュールとそれを構成するモジュールのうち仕様変更するモジュールとの選択（指定）を受け付け、当該集約モジュールの情報と当該仕様変更のモジュールの情報とをモジュール管理テーブル131から取得する（ステップ2001）。

【0227】

変更実行部2000は、選択範囲管理テーブル133から、選択された集約モジュールに対応する選択範囲ID 801と仕様変更するモジュールのIDとを特定する（ステップ2002）。

【0228】

変更実行部2000は、モジュール比較テーブル134を参照し、特定された選択範囲ID 801

10

20

30

40

50

と一致する選択範囲ID 908し、かつ仕様変更するモジュールIDを有する比較グループ902を特定する(ステップ2003)。

【0229】

変更実行部2000は、表示グループテーブル137を参照し、ステップ2003で特定した比較グループID 902と一致する比較グループIDを含む比較グループ1202に対応した表示グループを特定する(ステップ2004)。

【0230】

変更実行部2000は、集約実行部124に、特定した表示グループの表示グループIDを通知し(ステップ2005)、処理を完了する。

【0231】

ステップ2005で集約実行部124に通知されたIDに対応する表示グループは、仕様変更されるモジュールを含んだ比較グループを有するため、この表示グループについての集約実行部124のステップ1703以降の処理に適用することで、集約モジュールのプロパティ要素一覧をユーザに必要な最低限の判断で再設定させることができる。

【0232】

ちなみに、ユーザがモジュールの仕様変更を知った上で開始するのではなく、例えば制御部120がモジュールの仕様変更を検知しても良いし、またはモジュールの処理が失敗したことを制御部120や管理サーバ101が連携する監視システムなどが検知しても良い。また、それらの検知を契機に集約モジュールのプロパティの変更処理を開始の契機としても良い。

【0233】

以上、幾つかの実施例を説明したが、以上の説明を、例えば下記のように総括できる。なお、下記総括は、上述の説明に無い事項を含んでいても良い。

【0234】

管理サーバ101(プログラム作成支援装置の一例)において、集約候補生成部122が、ビジュアルプログラミングツール106を用いて作成されるプログラムを構成するN個のモジュール(Nは2以上の整数)のうち選択されたM個のモジュール(Mは、2以上且つN以下の整数)であるサブフローのうち、P個のプロパティをそれぞれ有するP個のモジュール(Pは、2以上且つM以下)を決定する。管理サーバ101において、集約実行部124が、一個のプロパティ項目として共通化可能であり当該一個のプロパティ項目の更新が反映されるQ個のプロパティ項目(Qは2以上且つP以下)であるプロパティ項目群がP個のプロパティに一個以上あれば、当該一個以上のプロパティ項目群の各々について、共通化可能であることを提示する。これにより、サブフローのプロパティについてユーザの設定工数が減少する。

【0235】

P個のモジュールの各々は、それぞれが複数個のモジュールを含む複数個のモジュールグループのうちモジュール関連ポイントが相対的に又は一定値以上に高いモジュールグループに属するモジュールである。各モジュールグループについて、モジュール関連ポイントは、それぞれがモジュール間の関係性の評価のために用意された一又は複数個のポリシーである一又は複数個のモジュール比較ポリシー(例えば種別602が“モジュール比較”のポリシー)のうち当該モジュールグループが満たす一個以上のモジュール比較ポリシーに基づき算出された評価値である。モジュール関連ポイントを基にP個のモジュールが決定されているため、関係性が低いモジュール間でプロパティ要素が同一又は類似であってもプロパティ項目が共通化されることが回避される。

【0236】

なお、「モジュールグループ」の一例は、上述の表示グループであっても比較グループであっても良い。具体的には、上述の実施例では、一個以上の比較グループが表示グループであり、比較グループ毎にモジュール関連ポイントが算出されるが、比較グループや表示グループのように複数種類のグループが用意されることに代えて、モジュールグループという種類のグループが用意され、モジュールグループ毎に、モジュール関連ポイントが

10

20

30

40

50

算出されても良い。

【0237】

また、「モジュール関連ポイントが相対的に又は一定値以上に高い」とは、モジュール間の関係性に関して実質的に評価値が高いという意味でも良く、具体的には、例えば、一又は複数個のモジュール比較ポリシーのうち所定の一個以上のモジュール比較ポリシーを満たした、ということでも良い。

【0238】

複数個のモジュール比較ポリシーは、モジュール間のホップ数に関するポリシーである接続関係ポリシー（例えばポリシーID：MP1のポリシー）と、ビジュアルプログラミングツール106が提供するワークスペースにおけるモジュール位置に関するポリシーである位置ポリシー（例えばポリシーID：MP2のポリシー）とのうちの少なくとも一個を含む。各モジュールグループについて、モジュール関連ポイントの算出には、接続関係ポリシー及び位置ポリシーのうち少なくとも一個が、それら以外のモジュール比較ポリシーに比べて優先的に使用される。ビジュアルプログラミングによれば、処理フローにおいてモジュール間の距離が近い程、又は、ワークスペースにおいて距離が近い程、モジュール間の関係性が高いと考えられ、そのような特性を満たすモジュールグループ程、高いモジュール関連ポイントが得られることが期待できる。なお、「接続関係ポリシー及び位置ポリシーのうち少なくとも一個が、それら以外のモジュール比較ポリシーに比べて優先的に使用」は、上述の実施例では、接続関係ポリシー及び位置ポリシーのうち少なくとも一個に関連付けられているポイント604が他のモジュール比較ポリシーのポイント604よりも高いことで実現されるが、他の方法で実現されても良い。

【0239】

複数個のモジュール比較ポリシーは、モジュールが配備される物理的又は仮想的な計算機リソースであるITインフラに関するポリシーであるITインフラポリシーを含む。これにより、モジュールが配備されるITインフラの観点で、P個のモジュールに含まれるか否かの基となるモジュール関連ポイントに影響を与えることができる。

【0240】

上述の一個以上のプロパティ項目群の各々は、それぞれがプロパティ要素間の関係性の評価のために用意された1又は複数個のポリシーである1又は複数個のプロパティ比較ポリシー（例えば、種別602が“プロパティ比較”のポリシー）のうち当該プロパティ項目群に適用されたプロパティ比較ポリシーを満たしたプロパティ項目群である。これにより、適切な二個以上のプロパティ項目を共通化可能なプロパティ項目として提示することができる。言い換えれば、プロパティ項目やプロパティ要素が同一であっても共通化に不適切なプロパティ項目を共通化可能として提示してしまうことを避けることができる。

【0241】

なお、上述の一個以上のプロパティ項目群の各々について、上記「適用されたプロパティ比較ポリシー」とは、下記(x)が下記(y)に適合したプロパティ比較ポリシーである。これにより、各プロパティ項目群について、当該プロパティ項目群に適切なプロパティ比較ポリシーを適用することが期待できる。

(x)それぞれがプロパティ要素に関する評価のために用意された1又は複数個のポリシーである1又は複数個のプロパティポリシー（例えば、種別602が“プロパティ”のポリシー）のうち、当該プロパティ項目群に属するプロパティ項目が満たすプロパティポリシー（例えば、ステップ1529において事前ポリシー1002として設定されたプロパティポリシー）。

(y)当該プロパティ比較ポリシーに予め関連付けられていたプロパティポリシー（例えば、プロパティ比較ポリシーについて事前ポリシー605として予め設定されているプロパティポリシー）。

【0242】

P個のモジュールの各々は、モジュール関連ポイントが相対的に又は一定値以上に高いモジュールグループのうち、プロパティ関連ポイントが相対的に又は一定値以上に高いモ

10

20

30

40

50

ジュールグループに属するモジュールでも良い。各ジュールグループについて、プロパティ関連ポイントは、一又は複数個のプロパティ比較ポリシーのうち当該ジュールグループが満たす一個以上のプロパティ比較ポリシーに基づき算出された評価値である。これにより、モジュール関連ポイントに加えてプロパティ関連ポイントを基に、共通化されるプロパティ項目を持つモジュールを絞り込むこと（例えば、モジュール間の関係性だけでなくプロパティ間の関係性も高いモジュールを絞り込むこと）ができる。

【0243】

サブフローのうちのいずれかのモジュールが変更された場合に、集約実行部124が、当該変更後のモジュールを含むP個のモジュール（例えば、変更実行部2000によって特定された比較グループに属するモジュール）について共通化可能の提示（例えば、ステップ1903以降）を実行する。これにより、サブフロー中のモジュールが変更されても変更後のモジュールを含むサブフロー（変更後のサブフロー）についてモジュール項目の共通化を維持できる。

10

【0244】

複数個のモジュール比較ポリシーは、ソースコード管理システムに対してジュールグループに属する全モジュールのソースコードが同時期に新規に作成されたことを定義したポリシー（例えば、ポリシーID：MP4のポリシー）と、ソースコード管理システムに対してジュールグループに属する全モジュールのソースコードが同時期に更新されたことを定義したポリシー（例えば、ポリシーID：MP5のポリシー）との少なくとも1個のポリシーを含む。ソースコード管理システムにより管理されている作成日時や更新日時が同時期（例えば、作成日時又は更新日時の差が一定値以下であること）であるモジュール間の関係性は高いと考えられるが、このような観点から、P個のモジュールに含まれるか否かの基となるモジュール関連ポイントに影響を与えることができる。

20

【0245】

複数個のモジュール比較ポリシーは、ジュールグループに属する全モジュールが同一の開発者によって開発されたことを定義するポリシーを含む。同一の開発者により開発されたモジュール間の関係性は高いと考えられるが、このような観点から、P個のモジュールに含まれるか否かの基となるモジュール関連ポイントに影響を与えることができる。

【0246】

複数個のモジュール比較ポリシーは、ジュールグループに属する全モジュールについて、基となるモジュール又はモジュール雛形のカテゴリが同一であることを定義するポリシーを含む。基となるモジュール又はモジュール雛形のカテゴリが同一であるモジュール間の関係性は高いと考えられるが、このような観点から、P個のモジュールに含まれるか否かの基となるモジュール関連ポイントに影響を与えることができる。

30

【0247】

以上、幾つかの実施例を説明したが、これらは本発明の説明のための例示であって、本発明の範囲をこれらの実施例にのみ限定する趣旨ではない。本発明は、他の種々の形態でも実行することが可能である。例えば、ビジュアルプログラミングツールは、「モデル開発環境」と呼ばれてもよい。プログラム（ソフトウェア）の構成要素や処理単位であるモジュールは「ノード」と呼ばれても良く、ノード同士の結線であるワイヤは「エッジ」と呼ばれても良い。ノードとエッジは有向グラフにおける用語であり、Node-REDにおけるフローダイアグラムも有向グラフとして捉えることができる。フローダイアグラムとブロックダイアグラムと有向グラフは同義とし、フローダイアグラムは単に「フロー」と呼ぶことができ、「フロー」と「グラフ」は同義とできる。フローを構成する二個以上のモジュールの集約モジュールが「サブフロー」に相当する。

40

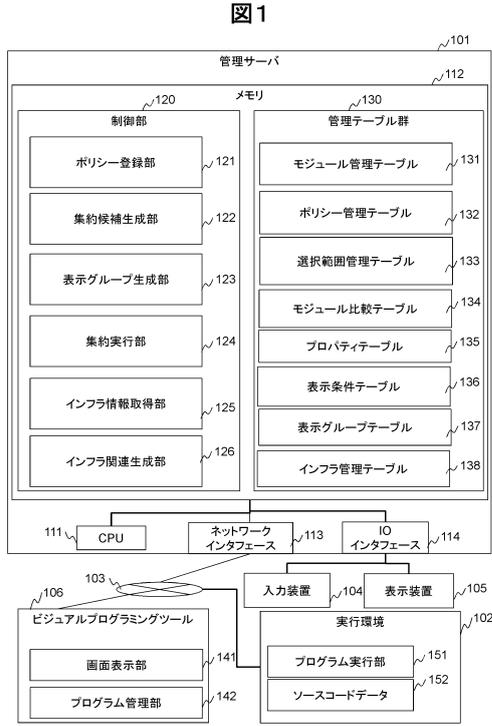
【符号の説明】

【0248】

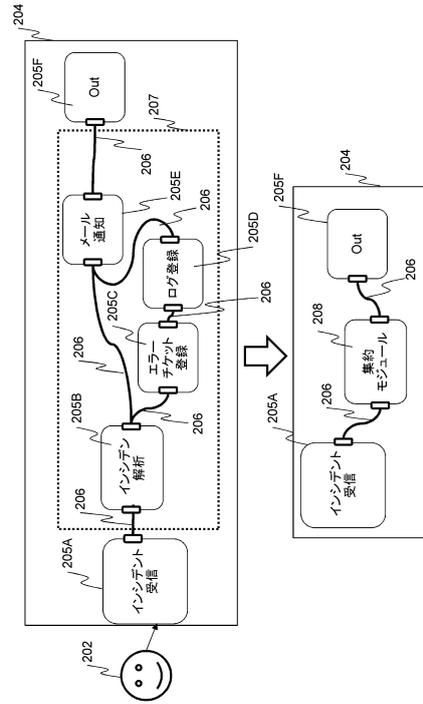
101：管理サーバ

50

【図面】
【図 1】



【図 2】

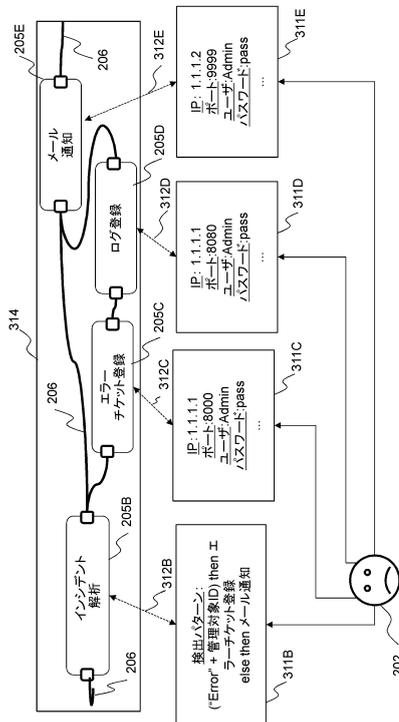


10

20

【図 3】

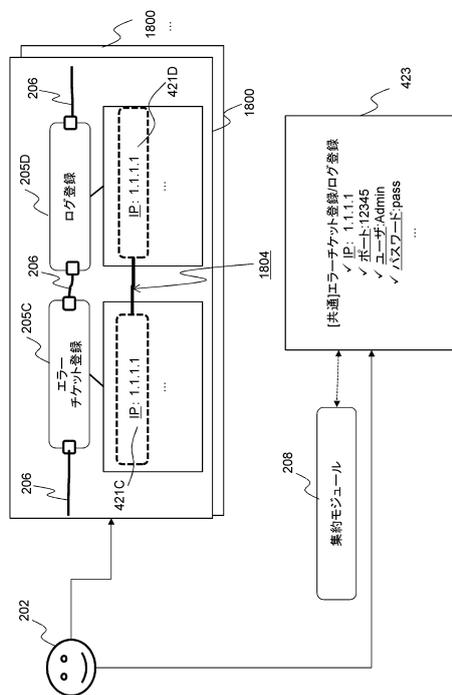
図3
比較例



30

【図 4】

図4



40

50

【 図 5 】

図5

モジュール管理テーブル
131

501	502	503	504	505	506	507	508	509	510
モジュールID	モジュール名	X座標	Y座標	接続先ID	プロパティ	生成コメント	更新コメント	カテゴリ	表示順
Mod1	エラーハンドリング	218	63	[Mod2, Mod4]	[抽出パターン{ value: ("Error" + 管理対象ID) then エラーチケット登録 else then メール通知, Type:String },...]	abcd123 efght23		Category1	1
Mod2	エラーチケット登録	392	117	[Mod3]	[{IP:(value.1.1.1.type.String)}, {Port:(value.8000.type.int)}, {UserID:(value.Admin.type.String)}, Password:(value.pass.type.String)}, ...]	abcd234 efght234		Category2	2
Mod3	ログ登録	542	161	[Mod4]	[{IP:(value.1.1.1.type.String)}, {Port:(value.8000.type.int)}, {UserID:(value.Admin.type.String)}, Password:(value.pass.type.String)}, ...]	abcd234 efght345		Category3	3
Mod4	メール通知	565	60	[Mod5]	[{IP:(value.1.1.1.type.String)}, {Port:(value.9000.type.int)}, {UserID:(value.Admin.type.String)}, Password:(value.pass.type.String)}, ...]	abcd345 efght345		Category4	4

【 図 6 】

図6

ポリシー管理テーブル
132

601	602	603	604	605
ポリシーID	種別	ポリシー内容	ポイント	事前ポリシー
PPC1	プロパティ比較	(\$ IP1 == \$IP2)?	5	PP1
PPC2	プロパティ比較	(\$ Port1 == \$ Port2)?	3	PP2
PPC3	プロパティ比較	(\$ UserID1 == \$ UserID2)?	5	PP5
PPC4	プロパティ比較	(\$ Password1 == \$ Password2)?	3	PP7
PPC5	プロパティ比較	(\$ APIToken1 == \$ APIToken2)?	5	-
PP1	プロパティ	\$IP.matches ("IPアドレス検出正規表現")	-	-
PP2	プロパティ	\$Port.matches ("Port検出正規表現")	-	-
PP3	プロパティ	\$URL.matches("先頭がhttp://または https:// を検査する正規表現")	-	-
PP4	プロパティ	データ型: Int型	-	-
PP5	プロパティ	データ型: String型	-	-
PP6	プロパティ	入力制限: Integer型(0<100)	-	-
PP7	プロパティ	入力条件: 英数字混在8文字以上	-	-
MP1	モジュール比較	(モジュール間ホップ数) == 1?	5	-
MP2	モジュール比較	(モジュール間距離) < 200pixel?	4	-
MP3	モジュール比較	同一ホスト上にデプロイされている?	3	-
MP4	モジュール比較	作成日時が同一コミット?	2	-
MP5	モジュール比較	前回更新日時が同一コミット?	2	-
MP6	モジュール比較	Author同一?	1	-
MP7	モジュール比較	Categoryが同一?	1	-

10

20

【 図 7 】

図7

インフラ管理テーブル
138

701	702	703	704	705	706
インフラID	管理IPアドレス	認証情報	OS	構成	関連モジュール
サーバ1	1.1.1.11	UserID:Admin Password:AAAA	W社 Server	-	Mod10
SaaS1	1.1.1.12	UserID:Admin Password:BBB APIkey:AAAAAA	-	-	Mod4
仮想サーバ1	1.1.1.1	UserID:Admin Password: Password	R社 Enterprise Server	Container 基盤ソフトD	[Mod2, Mod3]

【 図 8 】

図8

選択範囲管理テーブル
133

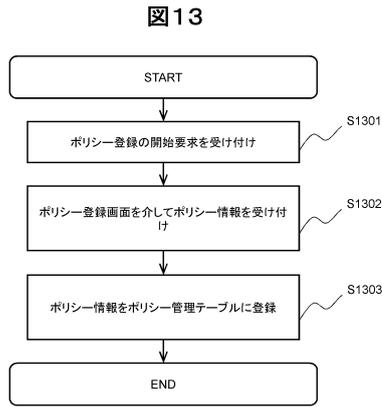
801	802	803
選択範囲ID	モジュールID	登録日時
SG1	[Mod1, Mod2, Mod3, Mod4]	2018/6/17 14:30

30

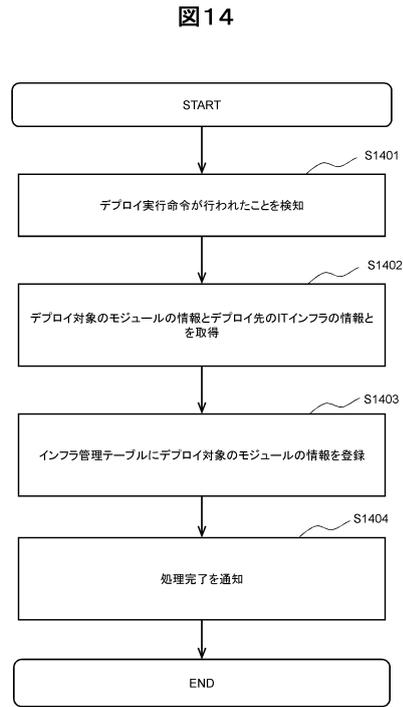
40

50

【 図 1 3 】



【 図 1 4 】

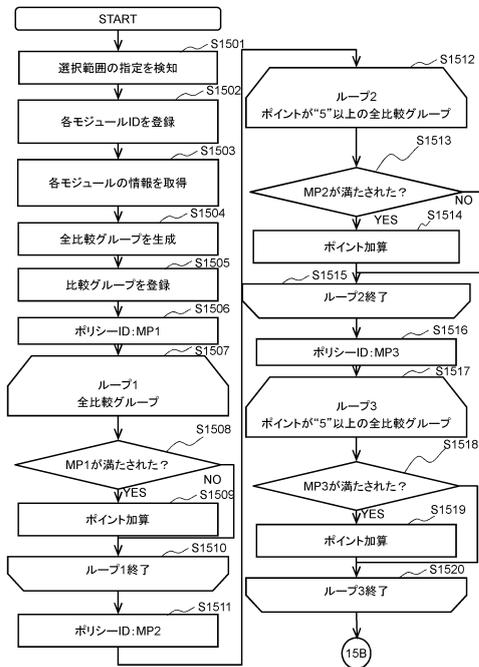


10

20

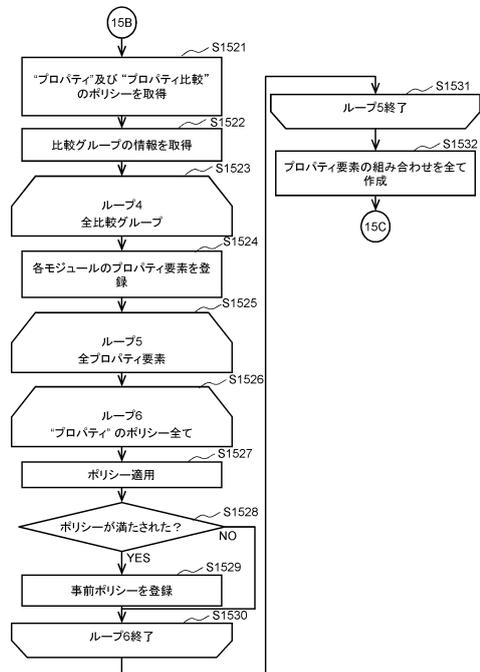
【 図 1 5 A 】

図15A



【 図 1 5 B 】

図15B



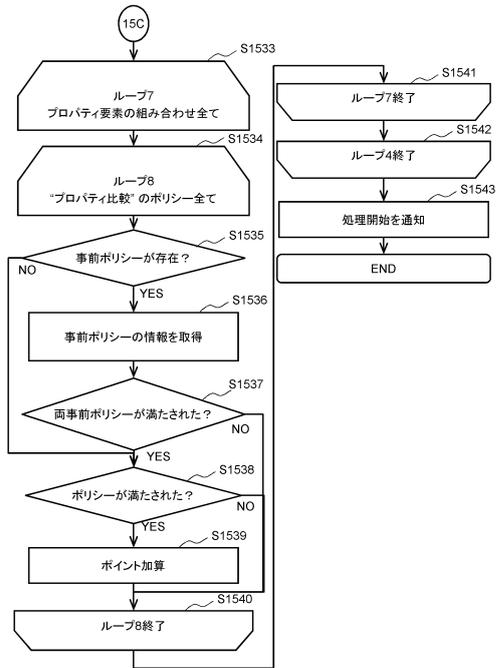
30

40

50

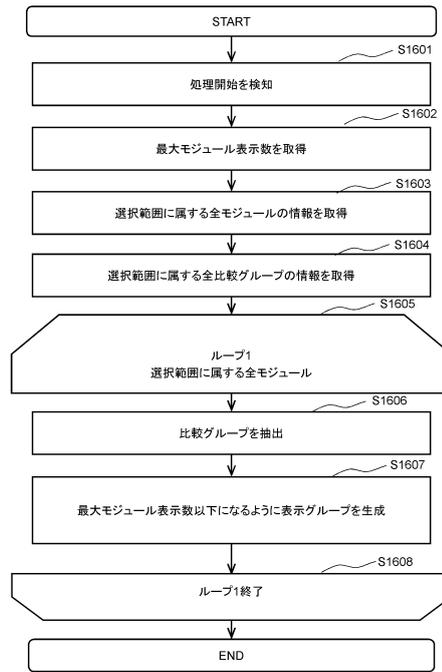
【 図 1 5 C 】

図15C



【 図 1 6 】

図16

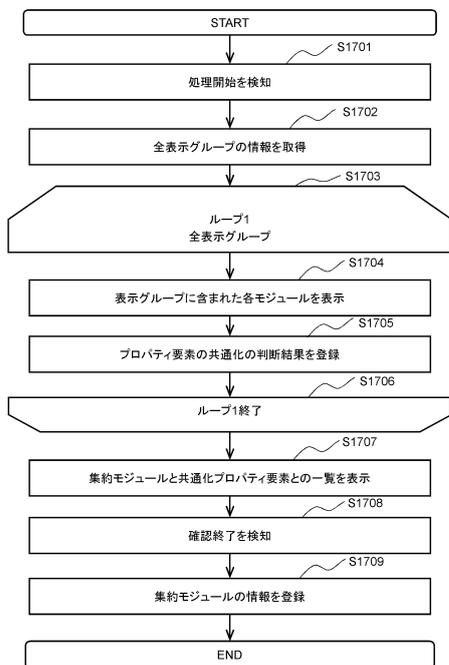


10

20

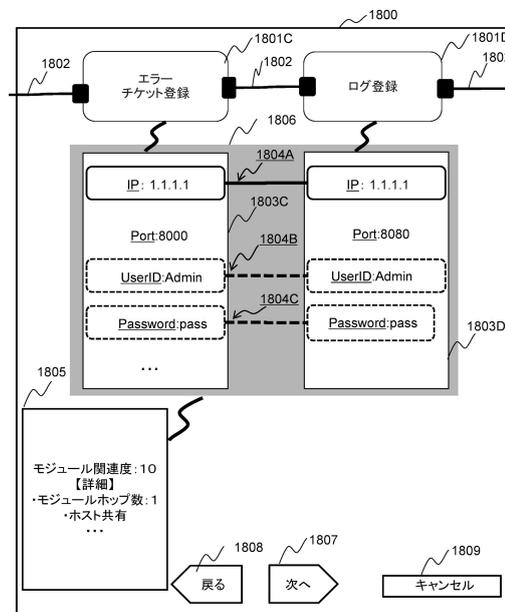
【 図 1 7 】

図17



【 図 1 8 】

図18



30

40

50

【 図 19 】

【 図 20 】

図19

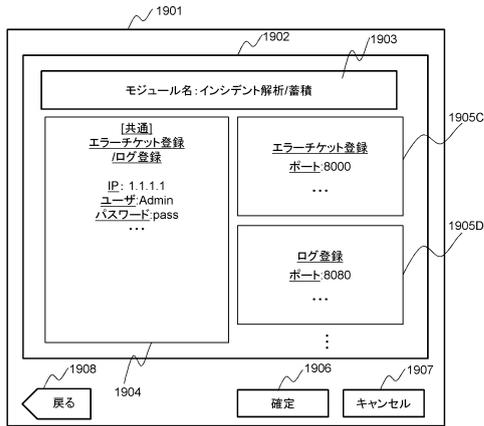
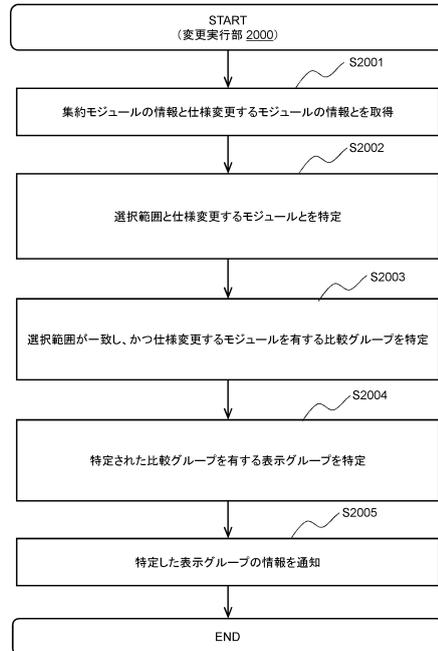


図20



10

20

30

40

50

フロントページの続き

- (72)発明者 守屋 哲
東京都千代田区丸の内一丁目6番6号 株式会社日立製作所内
- (72)発明者 西田 寿雄
東京都千代田区丸の内一丁目6番6号 株式会社日立製作所内
- 審査官 小林 秀和
- (56)参考文献 特開2014-164724(JP,A)
特開2002-366356(JP,A)
特開2000-215045(JP,A)
特表2007-537512(JP,A)
国際公開第2015/145556(WO,A1)
米国特許出願公開第2015/0169193(US,A1)
- (58)調査した分野 (Int.Cl., DB名)
G06F 8/34