

(19) 日本国特許庁(JP)

(12) 特 許 公 報(B2)

(11) 特許番号

特許第6630442号
(P6630442)

(45) 発行日 令和2年1月15日(2020.1.15)

(24) 登録日 令和1年12月13日(2019.12.13)

(51) Int. Cl. F I
G 0 6 F 8/60 (2018.01) G O 6 F 8/60
G 0 6 F 9/50 (2006.01) G O 6 F 9/50 1 2 O A

請求項の数 13 (全 28 頁)

(21) 出願番号	特願2018-541627 (P2018-541627)	(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
(86) (22) 出願日	平成28年6月7日(2016.6.7)	(74) 代理人	110001689 青稜特許業務法人
(65) 公表番号	特表2019-506682 (P2019-506682A)	(72) 発明者	河野 泰隆 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) 公表日	平成31年3月7日(2019.3.7)	審査官	北元 健太
(86) 国際出願番号	PCT/US2016/036246		
(87) 国際公開番号	W02017/213634		
(87) 国際公開日	平成29年12月14日(2017.12.14)		
審査請求日	平成30年8月9日(2018.8.9)		

最終頁に続く

(54) 【発明の名称】 適切なITリソース上にアプリケーションを配備するための管理コンピュータ及び非一時的なコンピュータ可読媒体

(57) 【特許請求の範囲】

【請求項1】

情報技術（IT）リソースを管理するように構成された管理コンピュータであって、アプリケーション開発リポジトリに通信可能に連結されており、

前記アプリケーション開発リポジトリ内のアプリケーションを配備するための要求から、前記ITリソース上に配備されるアプリケーションを決定し、

前記アプリケーション開発リポジトリからの情報に基づいて、前記配備されるアプリケーションの完成度レベルを決定し、

前記配備されるアプリケーションの前記ITリソースに基づいて1つ以上の構成を生成することであって、前記1つ以上の構成は前記完成度レベルによって生成され、

前記配備されるアプリケーションの前記1つ以上の構成に対して指定された前記ITリソースを割り振り、

前記配備されるアプリケーションを前記割り振られたITリソース上に配備するように構成されたプロセッサを備え、

1つ以上のアプリケーションの完成度レベルと1つ以上のタイムアウトとの間の関係性を記憶するように構成されたメモリをさらに備え、

前記プロセッサが、

前記1つ以上の構成の各々の配備に要する時間の計算と、

前記配備されるアプリケーションの前記アプリケーションの完成度レベル及び、前記1つ以上のアプリケーションの完成度レベルと前記1つ以上のタイムアウトとの間の前記関

10

20

係性に基づく、前記1つ以上の構成に対する前記タイムアウトの決定と、によって、前記配備されるアプリケーションの前記ITリソースに基づいて、1つ以上の構成を生成するように構成され、

前記決定されたタイムアウトを超過して前記配備に要する時間中、前記プロセッサが、前記タイムアウトが超過しているという表示を提供するように構成され、

前記プロセッサが、前記配備に要する時間が前記決定されたタイムアウトを超過していないとき、前記配備されるアプリケーションの前記1つ以上の構成に対して指定された前記ITリソースを割り振るように構成される、

管理コンピュータ。

【請求項2】

情報技術(IT)リソースを管理するように構成された管理コンピュータであって、アプリケーション開発リポジトリに通信可能に連結されており、

前記アプリケーション開発リポジトリ内のアプリケーションを配備するための要求から、前記ITリソース上に配備されるアプリケーションを決定し、

前記アプリケーション開発リポジトリからの情報に基づいて、前記配備されるアプリケーションの完成度レベルを決定し、

前記配備されるアプリケーションの前記ITリソースに基づいて1つ以上の構成を生成することであって、前記1つ以上の構成は前記完成度レベルによって生成され、

前記配備されるアプリケーションの前記1つ以上の構成に対して指定された前記ITリソースを割り振り、

前記配備されるアプリケーションを前記割り振られたITリソース上に配備するように構成されたプロセッサを備え、

前記プロセッサが、

前記アプリケーション開発リポジトリから前記情報を検索することであって、前記アプリケーション開発リポジトリからの前記情報は前記配備されるアプリケーションの1つ以上のアプリケーションログを備えるような検索と、

前記アプリケーションが配備される期間を通してなされた変更及び前記アプリケーションが配備される期間中の変更の合計数を計算することと、

1つ以上の完成度モデルを、前記アプリケーションが配備される前記期間を通してなされた前記変更及び前記アプリケーションが配備される前記期間中の前記変更の合計数に適用することによって、前記アプリケーション開発リポジトリからの情報に基づいて、前記配備されるアプリケーションの完成度レベルを決定するように構成された、

管理コンピュータ。

【請求項3】

1つ以上のアプリケーションの完成度レベルに利用可能な前記ITリソースに関する情報を格納するように構成されたメモリをさらに備え、前記プロセッサが、前記1つ以上のアプリケーションの完成度レベルに利用可能な前記ITリソースに関する前記情報に基づいて、前記配備されるアプリケーションの前記1つ以上の構成に対して指定された前記ITリソースを割り振るように構成された、請求項1または2に記載の管理コンピュータ。

【請求項4】

前記プロセッサが、1つ以上のシステムテンプレートの使用を介して、前記ITリソースを割り振るように構成され、

前記プロセッサが、

前記1つ以上のシステムテンプレートに基づいて、前記1つ以上の構成を作成し、

前記1つ以上の構成にわたり、よく使用されるリソースを前記ITリソースから特定し、

前記よく使用されるリソースを前記ITリソースから作成するように構成された、請求項1または2に記載の管理コンピュータ。

【請求項5】

別のアプリケーションを削除するための要求の受け付けについて、前記プロセッサが、

10

20

30

40

50

前記 IT リソースが前記よく使用されるリソースではないという決定に対して、前記別のアプリケーションによって利用される前記 IT リソースを削除するように構成される、請求項 4 に記載の管理コンピュータ。

【請求項 6】

別のアプリケーションをサイズ変更するための要求の受け付けについて、前記プロセッサが、

前記別のアプリケーションの完成度レベル及び使用頻度を決定し、

前記配備されるアプリケーションの前記 IT リソースに基づいて 1 つ以上の構成を生成するように構成され、前記 1 つ以上の構成は前記完成度レベル及び前記使用頻度によって生成される、請求項 1 または 2 に記載の管理コンピュータ。

10

【請求項 7】

1 つ以上のアプリケーションの完成度レベルと、1 つ以上の使用頻度と、1 つ以上のタイムアウトとの間の関係性を記憶するように構成されたメモリをさらに備え、

前記プロセッサが、

前記 1 つ以上の構成の各々の配置に要する時間の計算と、

前記サイズ変更されるアプリケーションの前記アプリケーションの完成度レベル、前記サイズ変更されるアプリケーションの前記使用頻度、及び前記 1 つ以上のアプリケーションの完成度レベルと前記 1 つ以上の使用頻度と前記 1 つ以上のタイムアウトとの間の前記関係性に基づく、前記 1 つ以上の構成に対するタイムアウトの決定とによって、前記サイズ変更されるアプリケーションの前記 IT リソースに基づいて、1 つ以上の構成を生成するように構成され、

20

前記決定されたタイムアウトを超過して前記配備に要する時間中、前記プロセッサが、前記タイムアウトが超過しているという表示を提供するように構成され、

前記プロセッサが、前記配備に要する時間が前記決定されたタイムアウトを超過していないとき、前記サイズ変更されるアプリケーションの前記 1 つ以上の構成に対して指定された前記 IT リソースを割り振るように構成される、請求項 6 に記載の管理コンピュータ。

【請求項 8】

情報技術 (IT) リソースを管理するように構成された管理コンピュータの命令を格納する非一時的なコンピュータ可読媒体であって、前記管理コンピュータはアプリケーション開発リポジトリに通信可能に連結されており、前記命令が、

30

前記アプリケーション開発リポジトリ内のアプリケーションを配備するための要求から、前記 IT リソース上に配備されるアプリケーションを決定することと、

前記アプリケーション開発リポジトリからの情報に基づいて、前記配備されるアプリケーションの完成度レベルを決定することと、

前記配備されるアプリケーションの前記 IT リソースに基づいて 1 つ以上の構成を生成することであって、前記 1 つ以上の構成は前記完成度レベルによって生成されことと、

前記配備されるアプリケーションの前記 1 つ以上の構成に対して指定された前記 IT リソースを割り振ることと、

前記配備されるアプリケーションを前記割り振られた IT リソース上に配備することと、を備え、

40

前記アプリケーション開発リポジトリからの情報に基づいて、前記配備されるアプリケーションの完成度レベルの前記決定が、

前記アプリケーション開発リポジトリから前記情報を検索することであって、前記アプリケーション開発リポジトリからの前記情報は前記配備されるアプリケーションの 1 つ以上のアプリケーションログを備えるような検索と、

前記アプリケーションが配備される期間を通してなされた変更及び前記アプリケーションが配備される前記期間中の変更の合計数を計算することと、

1 つ以上の完成度モデルを、前記アプリケーションが配備される前記期間を通してなされた前記変更及び前記アプリケーションが配備される前記期間中の前記変更の合計数に適用

50

することと、を備える、

非一時的なコンピュータ可読媒体。

【請求項 9】

前記命令が、1つ以上のアプリケーションの完成度レベルに利用可能な前記ITリソースに関する情報を格納することをさらに備え、前記配備されるアプリケーションの前記1つ以上の構成に対して指定された前記ITリソースの割り振りが、前記1つ以上のアプリケーションの完成度レベルに利用可能な前記ITリソースに関する前記情報に基づく、請求項8に記載の非一時的なコンピュータ可読媒体。

【請求項 10】

前記命令が、1つ以上のアプリケーションの完成度レベルと1つ以上のタイムアウトとの間の関係性を格納することをさらに備え、

前記配備されるアプリケーションの前記ITリソースに基づく前記1つ以上の構成の前記生成が、

前記1つ以上の構成の各々の配置に要する時間を計算することと、

前記配備されるアプリケーションの前記アプリケーションの完成度レベル及び前記1つ以上のアプリケーションの完成度レベルと前記1つ以上のタイムアウトとの間の前記関係性に基づいて、前記1つ以上の構成の前記タイムアウトを決定することと、を備え、

前記決定されたタイムアウトを超過して前記配備に要する時間中、前記管理コンピュータが、前記タイムアウトが超過しているという表示を提供するように構成され、

前記配備されるアプリケーションの前記1つ以上の構成に対して指定された前記ITリソースの前記割り振りが、前記配備に要する時間が前記決定されたタイムアウトを超過していないときに遂行される、請求項8に記載の非一時的なコンピュータ可読媒体。

【請求項 11】

前記命令が、別のアプリケーションを削除するための要求の受け付けについて、前記ITリソースがよく使用されるリソースではないという決定に対して、前記別のアプリケーションによって利用される前記ITリソースを削除することをさらに含む、請求項8に記載の非一時的なコンピュータ可読媒体。

【請求項 12】

前記命令が、別のアプリケーションをサイズ変更するための要求の受け付けについて、前記別のアプリケーションの完成度レベル及び使用頻度を決定することと、

前記配備されるアプリケーションの前記ITリソースに基づいて1つ以上の構成を生成することとあって、前記1つ以上の構成は前記完成度レベル及び前記使用頻度によって生成されことと、をさらに含む、請求項8に記載の非一時的なコンピュータ可読媒体。

【請求項 13】

前記命令が、1つ以上のアプリケーションの完成度レベルと、1つ以上の使用頻度と、1つ以上のタイムアウトとの間の関係性を格納することと、

前記1つ以上の構成の各々の配置に要する時間の計算と、前記サイズ変更されるアプリケーションの前記アプリケーションの完成度レベル、前記サイズ変更されるアプリケーションの前記使用頻度、及び前記1つ以上のアプリケーションの完成度レベルと前記1つ以上の使用頻度と前記1つ以上のタイムアウトとの間の前記関係性に基づいて、前記1つ以上の構成に対する前記タイムアウトの決定とによって、前記サイズ変更されるアプリケーションの前記ITリソースに基づいて、1つ以上の構成を生成することと、をさらに含み、

前記決定されたタイムアウトを超過して前記配備に要する時間中、前記タイムアウトが超過しているという表示を提供し、

前記配備に要する時間が前記決定されたタイムアウトを超過していないとき、前記サイズ変更されるアプリケーションの前記1つ以上の構成に対して指定された前記ITリソースを割り振る、請求項12に記載の非一時的なコンピュータ可読媒体。

【発明の詳細な説明】

【技術分野】

10

20

30

40

50

【 0 0 0 1 】

本開示は、一般に、情報技術（IT）システムに関し、より具体的には、適切なITリソース上へのアプリケーションの配備に関する。

【背景技術】**【 0 0 0 2 】**

関連技術の実装では、即時のホスト構成のための方法及び装置が存在する。そのような関連技術の実装では、ネットワークでアクセス可能なサービスのリソースインスタンスに対して構成可能な複数のインスタンスホストと、インスタンスホストのリモート構成を管理するための制御サーバを含むシステムが存在する。そのような実装例では、選択された制御サーバは、クライアントからのインスタンス構成要求に回答して、1つ以上のコマンドのシーケンスを、選択されたインスタンスホストに伝送する。選択されたインスタンスホストは、リモートコマンド実行部をインスタンス化する。リモートコマンド実行部は、コマンドシーケンスに対応する構成の動作を開始して終結する。選択された制御サーバは、実行部によって開始された動作の結果の少なくとも部分的に基づいて、インスタンス構成要求への回答を提供する。そのような関連技術の実施例は、全ての目的のために、参照によりその全体が本明細書に組み込まれる米国特許公開第2014/0207918 A1号に記載されている。

10

【発明の概要】**【課題を解決するための手段】****【 0 0 0 3 】**

関連技術の実装は、ITインフラストラクチャ及びリソースが配備されるときに、アプリケーションの完成度及びアプリケーションの使用頻度を考慮しない。本明細書に記載されている実装例は、ITインフラストラクチャを管理し、かつそれらの上に、アプリケーションの完成度を考慮しながらアプリケーションを配備する管理プログラムを提供することを目的とする。さらにまた、実装例は、アプリケーションの完成度及び使用頻度を考慮してアプリケーションをサイズ変更するために、ITリソースの構成を変更する管理プログラムを含む。

20

【 0 0 0 4 】

本開示の態様は、ITリソースを管理するように構成された管理コンピュータを含み、この管理コンピュータは、アプリケーション開発リポジトリに通信可能に連結されている。管理コンピュータは、アプリケーション開発リポジトリ内のアプリケーションを配備するための要求から、ITリソース上に配備されるアプリケーションを決定し、アプリケーション開発リポジトリからの情報に基づいて、配備されるアプリケーションの完成度レベルを決定し、配備されるアプリケーションのITリソースに基づいて1つ以上の構成を生成し、この1つ以上の構成は完成度レベルによって生成され、配備されるアプリケーションの1つ以上の構成に対して指定されたITリソースを割り振り、配備されるアプリケーションを割り振られたITリソース上に配備するように構成されたプロセッサを含むことができる。

30

【 0 0 0 5 】

本開示の態様は、情報技術（IT）リソースを管理するように構成された管理コンピュータの命令を格納する非一時的なコンピュータ可読媒体を含むことができ、この管理コンピュータは、アプリケーション開発リポジトリに通信可能に連結されている。この命令は、アプリケーション開発リポジトリ内のアプリケーションを配備する要求から、ITリソース上に配備されるアプリケーションを決定することと、アプリケーション開発リポジトリからの情報に基づいて、配備されるアプリケーションの完成度レベルを決定することと、配備されるアプリケーションのITリソースに基づいて、1つ以上の構成を生成することを含むことができ、この1つ以上の構成は完成度レベルによって生成され、配備されるアプリケーションの1つ以上の構成に対して指定されたITリソースを割り振ることと、配備されるアプリケーションを、割り振られたITリソース上に配備することと、を含むことができる。

40

50

【 0 0 0 6 】

本開示の態様は、情報技術（IT）リソースを管理するように構成された管理コンピュータのための方法を含むことができ、この管理コンピュータは、アプリケーション開発リポジトリに通信可能に連結されている。本方法は、アプリケーション開発リポジトリ内のアプリケーションを配備するための要求から、ITリソース上に配備されるアプリケーションを決定することと、アプリケーション開発リポジトリからの情報に基づいて、配備されるアプリケーションの完成度レベルを決定することと、配備されるアプリケーションのITリソースに基づいて、1つ以上の構成を生成することを含むことができ、この1つ以上の構成は完成度レベルによって生成され、配備されるアプリケーションの1つ以上の構成に対して指定されたITリソースを割り振ることと、配備されるアプリケーションを割り振られたITリソース上に配備することと、を含むことができる。

10

【 図面の簡単な説明 】

【 0 0 0 7 】

【 図 1 A 】 実施形態の本方法及び装置が適用され得るシステムの論理構成の一例を例示する。

【 0 0 0 8 】

【 図 1 B 】 ITインフラストラクチャの例示的な論理構成を例示する。

【 0 0 0 9 】

【 図 1 C 】 第1の実施形態例のIT環境の例示的な物理構成を例示する。

【 0 0 1 0 】

【 図 2 】 実装形態例に従って、管理コンピュータの構成を例示する。

20

【 0 0 1 1 】

【 図 3 】 実装形態例に従って、システムテンプレートを例示する。

【 0 0 1 2 】

【 図 4 】 実装形態例に従って、テンプレートのサイズ定義を例示する。

【 0 0 1 3 】

【 図 5 】 実装形態例に従って、物理ストレージテーブルを例示する。

【 0 0 1 4 】

【 図 6 】 実装形態例に従って、ストレージボリュームテーブルを例示する。

【 0 0 1 5 】

【 図 7 】 実装形態例に従って、物理サーバテーブルを例示する。

30

【 0 0 1 6 】

【 図 8 】 実装形態例に従って、仮想サーバテーブルを例示する。

【 0 0 1 7 】

【 図 9 】 実装形態例に従って、マッピングテーブルを例示する。

【 0 0 1 8 】

【 図 1 0 】 実装形態例に従って、ストレージ性能テーブルを例示する。

【 0 0 1 9 】

【 図 1 1 】 実装形態例に従って、サーバ性能テーブルを例示する。

【 0 0 2 0 】

【 図 1 2 】 実装形態例に従って、リソースグループテーブルを例示する。

40

【 0 0 2 1 】

【 図 1 3 】 実装形態例に従って、タイムアウト定義テーブルを例示する。

【 0 0 2 2 】

【 図 1 4 】 実装形態例に従って、セルフサービスポータルグラフィカルユーザインターフェイス（GUI）を例示する。

【 0 0 2 3 】

【 図 1 5 】 実装形態例に従って、アプリケーションを配備するための管理プログラムのフローチャートを例示する。

【 0 0 2 4 】

50

【図16】実装形態例に従って、アプリケーションの完成度を計算するための管理プログラムのフローチャートを例示する。

【0025】

【図17】実装形態例に従って、アプリケーションの完成度モデルを例示する。

【0026】

【図18】実装形態例に従って、構成のバリエーションを作成するための管理プログラムのフローチャートを例示する。

【0027】

【図19】実装形態例に従って、ITリソースを準備するための管理プログラムのフローチャートを例示する。

10

【0028】

【図20】実装形態例に従って、ITリソースを削除するための管理プログラムのフローチャートを例示する。

【0029】

【図21】第2の実施形態例に従って、管理コンピュータの構成を例示する。

【0030】

【図22】実装形態例に従って、リソースグループテーブルを例示する。

【0031】

【図23】実装形態例に従って、タイムアウト定義テーブルを例示する。

【0032】

20

【図24】実装形態例に従って、セルフサービスポータル GUI を例示する。

【0033】

【図25】実装形態例に従って、アプリケーションをサイズ変更するための管理プログラムのフローチャートを例示する。

【0034】

【図26】実装形態例に従って、構成のバリエーションを作成するための管理プログラムのフローチャートを例示する。

【発明を実施するための形態】

【0035】

以下の詳細な説明は、本出願の図面及び実装形態例のさらなる詳細を提供する。図面間の冗長な要素の参照番号及び説明は、明確にするために省略されている。明細書全体にわたって使用されている用語は、例として提供されており、限定を意図するものではない。例えば、「自動」という用語の使用は、本出願の実装形態を履行する当業者の所望の実装形態に応じて、実装形態の特定の態様によるユーザまたは管理者の制御を含む完全自動または半自動の実装形態を含んでいてもよい。

30

【0036】

第1の実施形態例

【0037】

第1の実施形態例では、ITインフラストラクチャを管理して、アプリケーションの完成度を考慮に入れながらインフラストラクチャ上にアプリケーションを配備するための管理プログラムが存在する。

40

【0038】

図1Aは、実施形態の方法及び装置が適用され得るシステムの論理構成の実施例を例示する。IT環境1000は、管理プログラム1200、アプリケーション配備リポジトリ1100、セルフサービスポータル1300、ITインフラストラクチャ管理UI1400、及びITインフラストラクチャ1500を含むことができる。アプリケーション開発者1010は、アプリケーションを開発し、アプリケーションの要素（例えば、ソースコード、構成ファイル、及び他のドキュメントまたは画像）をアプリケーション開発リポジトリ1100に格納する。アプリケーション開発者1010及びアプリケーション管理者1020は、セルフサービスポータル1300を介してIT環境1000にアクセスす

50

る。ITインフラストラクチャ管理者1030は、ITインフラストラクチャ管理UI1400を介してIT環境1000にアクセスする。

【0039】

図1Bは、ITインフラストラクチャ1500の論理構成例を示す。ITインフラストラクチャ1500は、1つ以上のサーバ及び/またはストレージレイを含む。図1Bの実施例では、2つのサーバ(1540と1550)及び2つのストレージレイ(1510と1520)が存在するが、本開示はそれに限定されず、かつ所望の実装形態によって任意の数のサーバ及びストレージレイを含むことができる。アプリケーション1544、OS(オペレーティングシステム)1543及びVM(仮想マシン)1542は、ハイパーバイザー1541上で実行される。ハイパーバイザー1541は、サーバ1540 10
上で実行される。OS1554、VM1552、及びVM1553は、ハイパーバイザー1551上で実行される。ハイパーバイザー1551は、サーバ上で実行される。アプリケーション1544は、ストレージレイ1510のストレージボリューム1511及びストレージボリューム1512を使用する。ストレージレイ1520のストレージボリューム1521は、VM1552及びOS1554に割り振られるが、図1Bの実施例では、OS1554上のアプリケーションはボリュームを使用していない。ストレージレイ1522のストレージボリューム1522は、VM1553に割り振られる。

【0040】

図1Cは、第1の実施形態例のためのIT環境の物理構成例を例示する。IT環境1000は、管理コンピュータ2000、サーバ3000、ストレージレイ4000、管理ネットワーク5000、及びデータネットワーク6000を含む。サーバ3000及び 20
ストレージレイ4000は、データネットワーク6000を介して接続される。このネットワークはLAN(ローカルエリアネットワーク)であってもよいが、所望の実装形態に応じて他のネットワークもまた利用されてもよい。管理コンピュータ2000、サーバ3000、及びストレージレイ4000は、管理ネットワーク5000を介して接続され、またLANまたは任意の他の所望のネットワークの実装形態としても実装され得る。この実施形態例では、管理ネットワーク5000及びデータネットワーク6000が分離されているが、それらは、所望の実装形態に応じて、単一の集束ネットワークで置き換えることができる。この実施形態例では、管理コンピュータ2000及びサーバ3000は分 30
離されるが、所望の実装形態によって他の方法で実装され得る。例えば、任意のサーバが管理プログラムをホストすることができる。さらにまた、サーバ3000及びストレージレイ4000は、この実施形態例では分離されているが、サーバ及びストレージレイを1つのシステムに集束させることができ、または他の所望の実装形態を利用することもできる。

【0041】

図2は、実装形態例に従って、管理コンピュータの構成を例示する。管理ネットワークインターフェース2100は、管理コンピュータ2000を管理ネットワーク5000に接続するために利用されるインターフェースである。入力及び出力デバイス2300は、モニター、キーボード、及びマウスなどのユーザインターフェースである。ローカルディスク2400は、管理プログラム2410、システムテンプレート2420、及びアプ 40
リケーションの完成度モデル2430を含む。管理プログラム2410は、メモリ2500にロードされ、1つ以上の物理的な中央処理装置(CPU)の形態とすることができるプロセッサ2200によって実施される。管理プログラム2410の手順の詳細は、本開示全体を通じて提供される。管理プログラム2410は、図1Aの管理プログラム1200のような同じエンティティであることができる。

【0042】

システムテンプレート2420及びアプリケーションの完成度モデル2430は、メモリ2500にロードされて管理プログラム2410によって使用される。これらのテンプレート及びモデルを以下に開示する。メモリ2500は、ストレージレイボリューム2510、ストレージボリュームテーブル2520、物理サーバテーブル2530、仮想 50

サーバテーブル 2540、マッピングテーブル 2550、ストレージ性能テーブル 2560、サーバ性能テーブル 2570、リソースグループテーブル 2580、及びタイムアウト定義テーブル 2590を含み、これらのさらなる詳細は、本開示全体を通じて提供される。

【0043】

実装形態例では、管理コンピュータ 2000は、サーバ 3000及びストレージレイノストレージシステム 4000などのITリソースを管理するように構成され、管理ネットワーク 5000を介してアプリケーション開発リポジトリ 1100に通信可能に連結され、管理ネットワーク 5000に接続されたデータベースの形態で実装され得る。

【0044】

管理プログラム 2410は、プロセッサ 2200内にロードされ得る命令の形態であってもよく、その結果、プロセッサ 2200は、アプリケーション開発リポジトリ内のアプリケーションを配備するための要求から、ITリソース上に配備されるアプリケーションを決定し、アプリケーション開発リポジトリからの情報に基づいて、配備されるアプリケーションの完成度レベルを決定し、配備されるアプリケーションのITリソースに基づいて、1つ以上の構成を生成し、この1つ以上の構成は完成度レベルによって生成され、配備されるアプリケーションの1つ以上の構成に対して指定されたITリソースを割り振り、図15にさらなる詳細が説明されるように、配備されるアプリケーションを、割り振られたITリソース上に配備するよう構成される。プロセッサ 2200は、例えば、図18に説明されているように、ITリソースからのストレージボリューム及び仮想VMのうちの少なくとも1つの作成により配備されるアプリケーションの1つ以上の構成に対して指定されるITリソースを割り振るよう構成されている。

【0045】

実装形態例では、リソースグループテーブル 2580は、1つ以上のアプリケーションの完成度レベルに利用可能なITリソースに関する情報を含み、1つ以上のアプリケーションの完成度レベルに利用可能なITリソースに関する情報に基づいて配備されるアプリケーションの1つ以上の構成に対して指定されたITリソースを割り振るために、プロセッサ 2200によって使用され得る。タイムアウト定義テーブル 2590はまた、1つ以上のアプリケーションの完成度レベルと1つ以上のタイムアウトとの間の関係性を表す情報も含むことができ、プロセッサ 2200は、1つ以上の構成の各々の配備に要する時間の計算によって、配備されるアプリケーションのITリソースに基づいて、1つ以上の構成を生成するよう構成され、配備されるアプリケーションの、アプリケーションの完成度及び1つ以上のアプリケーションの完成度レベルと1つ以上のタイムアウトとの間の関係性に基づいて、1つ以上の構成に対するタイムアウトの決定も含む。

決定されたタイムアウトを超過して配備に要する時間中、プロセッサ 2200は、タイムアウトが超過しているという表示を提供するよう構成され、プロセッサ 2200は、図13及び15に説明されているように、配備に要する時間が決定されたタイムアウトを超過していないとき、配備されるアプリケーションの1つ以上の構成に対して指定されるITリソースを割り振るよう構成されている。

【0046】

実装形態例では、例えば、図16及び17に説明されているように、プロセッサ 2200は、アプリケーション開発リポジトリから情報を検索することによって、アプリケーション開発リポジトリからの情報に基づいて、配備されるアプリケーションの完成度レベルを決定するよう構成され、アプリケーション開発リポジトリからの情報は配備されるアプリケーションの1つ以上のアプリケーションログと、アプリケーションが配備される期間を通してなされた変更及びアプリケーションが配備される期間中の変更の合計数を計算することと、1つ以上の完成度モデルを、アプリケーションが配備される期間を通してなされた変更及びアプリケーションが配備される期間中の変更の合計数に適用することと、を備える。

【0047】

10

20

30

40

50

実装形態例では、プロセッサ 2 2 0 0 は、例えば、図 1 9 に説明されているように、1 つ以上のシステムテンプレートの使用を介して IT リソースを割り振るように構成され、プロセッサ 2 2 0 0 は、1 つ以上のシステムテンプレートに基づいて 1 つ以上の構成を作成し、1 つ以上の構成にわたって IT リソースからよく使用されるリソースを特定し、よく使用されるリソースを IT リソースから作成するように構成されている。

【 0 0 4 8 】

別のアプリケーションを削除するための要求を受け付ける実装形態例では、プロセッサ 2 2 0 0 は、例えば、図 2 0 に説明されているように、IT リソースがよく使用されるリソースではないという決定に対して、別のアプリケーションによって利用される IT リソースを削除するように構成されている。

10

【 0 0 4 9 】

別のアプリケーションをサイズ変更するための要求を受け付ける実装形態例では、例えば、図 2 4 及び図 2 5 に説明されているように、プロセッサは、別のアプリケーションの完成度レベル及び使用頻度を決定し、配備されるアプリケーションの IT リソースに基づいて 1 つ以上の構成を生成するように構成され、この 1 つ以上の構成は完成度レベル及び使用頻度によって生成される。

【 0 0 5 0 】

図 3 は、実装形態例に従って、システムテンプレート 2 4 2 0 を例示する。これらのテンプレートは、ローカルディスク 2 4 0 0 からメモリ 2 5 0 0 にロードされ、管理コンピュータ 2 0 0 0 の使用されるプロセッサ 2 2 0 0。アプリケーション開発者 1 0 1 0 は、アプリケーションを作成し、かつこのアプリケーションを、セルフサービスポータル 1 3 0 0 を介してシステムテンプレートを選択することにより IT インフラストラクチャ 1 5 0 0 上に配備する。

20

【 0 0 5 1 】

具体的には、図 3 は、「モバイルアプリテンプレート # 1」と名付けられたシステムテンプレートの例を示す。このテンプレートは、一組の構成要素が配備されていることを示す配備ターゲット 2 4 2 1 を有する。このテンプレートの配備ターゲット 2 4 2 1 は、アプリケーション 2 4 2 2、及びアプリケーションサーバ 2 4 2 3、及び NoSQL DB (データベース) 2 4 2 4 を含む。このテンプレートを使用することによって作成されたシステムは、モバイルデバイス 2 4 2 5 上のウェブブラウザを含むことができるが、そのようなウェブブラウザは配備されていない。この例示的なテンプレートはまた、サイズ定義 2 4 2 6 も有する。

30

【 0 0 5 2 】

図 4 は、実装形態例に従って、テンプレートのサイズ定義 2 4 2 6 を例示する。サイズ定義は、配備されるシステムのサイズの利用可能なオプションを説明する。つまり、図 4 は、「モバイルアプリテンプレート # 1」と名付けられたシステムテンプレートのサイズ定義の実施例を示す。実装形態例では、このシステムに対して小サイズまたは中サイズが選択され得る。

【 0 0 5 3 】

図 5 は、実装形態例に従って、物理ストレージテーブル 2 5 1 0 を例示する。このテーブルは、管理プログラム 2 4 1 0 によってメモリ 2 5 0 0 内に作成される。列 2 5 1 1 は、ストレージレイの ID を示す。列 2 5 1 2 は、ストレージレイのプロセッサを示す。列 2 5 1 3 は、ストレージレイのポートを示す。列 2 5 1 4 は、ストレージレイのキャッシュリソースを示す。列 2 5 1 5 は、ストレージレイのリソースのプール (通常は容量) を示す。各行 (2 5 1 A、2 5 1 B、2 5 1 C、2 5 1 D) は、各物理ストレージレイの構成を示す。例えば、行 2 5 1 A 及び 2 5 1 B は、ストレージレイ 0 1 の構成を示す。ストレージレイは、各々 8 個のコアを備えた 2 つのプロセッサ、ポート A、B、C、及び D の 8 G b p s、キャッシュ C - 0 1 の 1 6 0 G B 及びキャッシュ C - 0 2 の 1 2 8 G B、プール P o o l - 0 1 及び P o o l - 0 2 の 3 0 0 T B、及びプール P o o l - 0 3 及び P o o l - 0 4 の 5 0 0 T B を有する。

40

50

【 0 0 5 4 】

図 6 は、実装形態例に従って、ストレージボリュームテーブル 2 5 2 0 を例示する。このテーブルは、管理プログラム 2 4 1 0 によってメモリ 2 5 0 0 内に作成される。列 2 5 2 1 は、ストレージボリュームを所有するストレージアレイの ID を示す。列 2 5 2 2 は、ストレージボリュームの ID を示す。列 2 5 2 3 は、各ストレージボリューム容量を示す。列 2 5 2 4 は、ストレージボリュームが湾曲しているプールの ID を示す。列 2 5 2 5 は、各ボリュームが特定のユーザまたは特定のシステムのために予約されているか否かを表すフラグを示す。列 2 5 2 6 は、ストレージボリュームのメディアタイプを示す。

【 0 0 5 5 】

各行 (2 5 2 A、2 5 2 B、2 5 2 C、2 5 2 D、2 5 2 E) は、各ストレージボリュームの構成を示す。例えば、行 2 5 2 A は、ストレージアレイ 0 1 のストレージボリューム 0 1 の構成を示す。このストレージボリュームは、1 0 T B の容量を有し、Pool - 0 1 から湾曲し、特定のユーザまたは特定のシステムのために予約され、そのメディアタイプは半導体ドライブ (S S D) である。

10

【 0 0 5 6 】

図 7 は、実装形態例に従って、物理サーバテーブル 2 5 3 0 を例示する。このテーブルは、管理プログラム 2 4 1 0 によってメモリ 2 5 0 0 内に作成される。列 2 5 3 1 は、物理サーバの ID を示す。列 2 5 3 2 は、各物理サーバの CPU のコアの数及びタイプを示す。列 2 5 3 3 は、各物理サーバのメモリリソースの容量を示す。列 2 5 3 4 は、各物理サーバのポートを示す。

20

【 0 0 5 7 】

各行 (2 5 3 A、2 5 3 B、2 5 3 C、2 5 3 D) は、各物理サーバの構成を示す。例えば、行 2 5 3 A は、物理サーバ 0 1 の構成を示す。物理サーバは、正常な CPU の 1 2 個のコア、3 2 G B のメモリ、及び 4 G b p s のポート A 及び B を有する。

【 0 0 5 8 】

図 8 は、実装形態例に従って、仮想サーバテーブル 2 5 4 0 を例示する。このテーブルは、管理プログラム 2 4 1 0 によってメモリ 2 5 0 0 内に作成される。列 2 5 4 1 は、仮想サーバの ID を示す。列 2 5 4 2 は、仮想サーバが動作している物理サーバの ID を示す。列 2 5 4 3 は、各仮想サーバに割り当てられた CPU コアの数を示す。列 2 5 4 4 は、各仮想サーバに割り当てられたメモリリソースの容量を示す。列 2 5 4 5 は、各仮想サーバに割り当てられたポートを示す。列 2 5 4 6 は、各仮想サーバが特定のユーザまたは特定にシステムのために予約されているか否かを表すフラグを示す。列 2 5 4 7 は、各仮想サーバに割り振られた記憶容量を示す。

30

【 0 0 5 9 】

各行 (2 5 4 A、2 5 4 B、2 5 4 C、2 5 4 D) は、各仮想サーバの構成を示す。例えば、行 2 5 4 A は、仮想サーバ 0 1 の構成を示す。この仮想サーバは、物理サーバ 0 1 でホストされ、2 つの CPU コア、4 G B のメモリ及び 4 G b p s のポート A、5 0 0 G B の記憶容量を有し、かつ特定のユーザまたは特定のシステムのために予約されている。

【 0 0 6 0 】

図 9 は、実装形態例に従って、マッピングテーブル 2 5 5 0 を例示する。このテーブルは、管理プログラム 2 4 1 0 によってメモリ 2 5 0 0 内に作成される。列 2 5 5 1 は、アプリケーションの ID を示す。列 2 5 5 2 は、アプリケーション名を示す。この名前は、アプリケーション開発者 1 0 1 0 によってセルフサービスポータル 1 6 0 0 の GUI 1 6 0 0 - A のアプリケーション名フィールド 1 6 2 0 - A で指定されている。

40

【 0 0 6 1 】

列 2 5 5 3 は、テンプレートの ID を示す。テンプレートのこの ID は、アプリケーション開発者によってセルフサービスポータルの GUI 1 6 0 0 - A のテンプレートドロップダウンリスト 1 6 3 0 - A で指定されている。列 2 5 5 4 は、アプリケーションが動作している仮想サーバの ID を示す。列 2 5 5 5 は、仮想サーバ名を示す。この実装形

50

態例では、これらの名前は管理プログラムによるアプリケーション名に基づいて自動的に作成されるが、所望の実装形態に応じて他の実装が可能である。例えば、アプリケーション管理者が各仮想サーバ名を指定することができる。列 2 5 5 6 は、仮想サーバのポートの ID を示す。列 2 5 5 7 は、ストレージレイの ID を示す。列 2 5 5 8 は、ストレージレイのポートの ID を示す。列 2 5 5 9 は、ストレージボリュームの ID を示す。

【 0 0 6 2 】

各行 (2 5 5 A、2 5 5 B、2 5 5 C、2 5 5 D、2 5 5 E、2 5 5 F、2 5 5 G) は、アプリケーションとストレージボリュームとの間のエンドツーエンドのマッピングを示す。例えば、行 2 5 5 B は、アプリケーション 2 が「Web - C」という名前を有し、テンプレート 4 から作成され、「Web DB」という名前の仮想サーバ 0 3 上で動作し、さらに 2 つのストレージボリューム 0 5 2 及び 0 5 5 がアプリケーションの仮想サーバ 0 3 に割り当てられていることを示す。ストレージレイ 0 2 のストレージボリューム 0 5 2 は、ストレージレイのポート B 及び仮想サーバポート A を介して仮想サーバ 0 3 に割り当てられる。ストレージレイ 0 1 のストレージボリューム 0 5 5 は、ストレージレイのポート A 及び仮想サーバポート A を介して仮想サーバ 0 3 に割り当てられる。

【 0 0 6 3 】

図 1 0 は、実装形態例に従って、ストレージ性能テーブル 2 5 6 0 を例示する。このテーブルは管理プログラム 2 4 1 0 によってメモリ 2 5 0 0 内に作成される。列 2 5 6 1 は、ストレージレイの ID を示す。列 2 5 6 2 は、ストレージボリュームの ID を示す。列 2 5 6 3 は、ストレージボリュームの性能データの履歴の ID を示す。タイムスタンプが履歴 ID 2 5 6 3 として使用され得る。列 2 5 6 4 は、プロセッサの使用率を示す。列 2 5 6 5 は、ストレージボリュームに割り当てられたキャッシュリソースの使用率を示す。列 2 5 6 6 は、ストレージボリュームが湾曲しているプールの使用率を示す。列 2 5 6 7 は、ストレージボリュームに割り当てられたポートの使用率を示す。各行 (2 5 6 A、2 5 6 B) は、各ストレージボリュームの性能データの履歴を示す。例えば、列 2 5 6 A は、履歴データのうちの少なくとも 3 つの記録 (0 ~ 2) を有するストレージレイ 0 1 のストレージボリューム 0 1 の性能データを示す。

【 0 0 6 4 】

図 1 1 は、実装形態例に従って、サーバ性能テーブル 2 5 7 0 を例示する。このテーブルは、管理プログラム 2 4 1 0 によってメモリ 2 5 0 0 内に作成される。列 2 5 7 1 は、物理及び/または仮想サーバの ID を示す。列 2 5 7 2 は、サーバが物理サーバであるか否かを表すフラグを示す。値が「はい」の場合は、そのサーバは物理サーバである。値が「いいえ」の場合は、そのサーバは仮想サーバである。列 2 5 7 3 は、サーバの性能データの履歴の ID を示す。タイムスタンプが履歴 ID 2 5 7 3 として使用され得る。列 2 5 7 4 は、サーバの CPU の使用率を示す。列 2 5 7 5 は、サーバのメモリリソースの使用率を示す。列 2 5 7 6 は、サーバのディスクの使用率を示す。列 2 5 7 7 は、サーバのポートの使用率を示す。

【 0 0 6 5 】

各行は、各サーバの性能データの履歴を示す。例えば、行 2 5 7 A は、物理サーバであり、少なくとも 3 つの履歴データ (0 ~ 2) を有するサーバ 0 1 の性能データを示す。

【 0 0 6 6 】

図 1 2 は、実装形態例に従って、リソースグループテーブル 2 5 8 0 を例示する。このテーブルは、管理プログラム 2 4 1 0 によってメモリ 2 5 0 0 内に作成される。列 2 5 8 1 は、リソースのタイプを示す。列 2 5 8 2 は、リソースの ID を示す。列 2 5 8 3 は、対応するアプリケーションの完成度を示す。

【 0 0 6 7 】

各行 (2 5 8 A、2 5 8 B、2 5 8 C、2 5 8 D、2 5 8 E、2 5 8 F、2 5 8 G) は、完成度の定義に割り当てられたリソースを示す。例えば、行 2 5 8 A は、ストレージレイ 0 1 が「高い」完成度に割り当てられていることを示す。これは、ストレージレイ 0 1 が高い完成度を有するアプリケーションにのみ使用され得ることを意味する。図 1

10

20

30

40

50

2 に示すリソースグループテーブルの実施例は、ストレージレイ及び物理サーバのみを含むが、このテーブルは所望の実装形態に応じて他のタイプのリソースを含んでいてもよい。

【 0 0 6 8 】

図 1 3 は、実装形態例に従って、タイムアウト定義テーブル 2 5 9 0 を例示する。このテーブルは、管理プログラム 2 4 1 0 によってメモリ 2 5 0 0 内に作成される。列 2 5 9 1 は、アプリケーションの完成度レベルを示す。列 2 5 9 2 は、アプリケーションの配備プロセス中に構成のバリエーションを作成するためのタイムアウトを示す。各行 (2 5 9 A、2 5 9 B) は、各完成度に定義されたタイムアウト値を示す。例えば、行 2 5 9 A は、低い完成度のアプリケーションに定義された 6 0 秒のタイムアウト値を示す。タイムアウト値は、他の方法で表示され得る。例えば、タイムアウトは、作成される構成のバリエーションの数として、または所望の実装形態による他の方法として表示され得る。

10

【 0 0 6 9 】

図 1 4 は、実装形態例に従って、セルフサービスポータル 1 6 0 0 の G U I 1 6 0 0 - A を例示する。この G U I は、アプリケーション開発者 1 0 1 0 がアプリケーションを配備するときに使用される。アプリケーション開発者は、アプリケーション 1 6 1 0 - A、システムテンプレート 1 6 3 0 - A 及びサイズ 1 6 4 0 - A を選択し、かつアプリケーション名 1 6 2 0 - A を入力する。アプリケーションフィールド 1 6 1 0 - A に示す I D は、アプリケーション開発リポジトリ 1 1 0 0 内のアプリケーションの I D である。

【 0 0 7 0 】

配備ボタン 1 6 5 0 - A がクリックされると、管理プログラム 2 4 1 0 は、指定されたシステムテンプレートを使用することにより、指定されたアプリケーションを配備する。取り消しボタン 1 6 6 0 - A がクリックされると、管理プログラム 2 4 1 0 はアプリケーションの配備プロセスを取り消す。

20

【 0 0 7 1 】

図 1 5 は、実装形態例に従って、アプリケーションを配備するための管理プログラム 2 4 1 0 のフローチャートを示す。このフローは、手順が開始される 1 0 0 1 0 で始まる。1 0 0 2 0 において、管理プログラム 2 4 1 0 はアプリケーションを配備するための要求をセルフサービスポータル 1 6 0 0 から受け付ける。図 1 4 に示すパラメータは管理プログラム 2 4 1 0 に渡される。1 0 0 3 0 において、管理プログラム 2 4 1 0 は、以下でさらに詳細に開示される「アプリケーションの完成度の計算」のサブシーケンスを呼び出す。1 0 0 4 0 において、管理プログラム 2 4 1 0 は、以下でさらに詳細に開示される「構成のバリエーションの作成」のサブシーケンスを呼び出す。1 0 0 5 0 において、1 0 0 4 0 のフローで作成された各構成のバリエーションについて、管理プログラム 2 4 1 0 は、作成されなければならないリソースを特定して、作成に要する時間 (「 t 1 」 と呼ばれる) を推定する。例えば、構成のバリエーションが 4 つの C P U コア、8 1 9 2 G B の R A M と 5 0 0 G B の H D D、及び 4 T B の S S D ストレージボリュームを有する V M を含むことができ、要件を満たすストレージボリュームが存在する場合、次に管理プログラム 1 4 1 0 は V M のみが作成されなければならないと判断する。管理プログラム 1 4 1 0 は、図 6 のストレージボリュームテーブル 2 5 2 0 または図 8 の仮想サーバテーブル 2 5 1 0 を参照することによって、そのようなストレージボリュームまたは V M が存在するかどうかを判断することができる。つまり、管理プログラム 2 4 1 0 は、4 T B の容量、予約したフラグの 0 値及び S S D メディアタイプを有するストレージボリュームがストレージボリュームテーブル 2 5 2 0 内に存在するかどうかを確認する。行 2 5 2 D に示すストレージボリュームは要件と一致するので、管理プログラム 2 4 1 0 はその容量が使用できると判断する。管理プログラム 2 4 1 0 はまた、4 つの C P U コア、8 1 9 2 G B の R A M、5 0 0 G B の H D D 及び 0 値の予約されたフラグを有する V M が、図 8 の仮想サーバテーブル 2 5 4 0 内に存在するかどうかを確認する。図 8 の仮想サーバテーブル 2 5 4 0 には要件を満たす V M を有していないので、管理プログラム 2 4 1 0 は、V M が作成されなければならないと判断する。管理プログラム 2 4 1 0 は、所定の作成時間 (例えば、1 V M の

30

40

50

作成に5分かかる)または以前に行われた平均作成時間に基づいて、 t_1 を推定することができる。上記の実施例では、管理プログラム2410は、例えば、 t_1 を5分と推定することができる。

【0072】

10060において、管理プログラム2410は、指定されたシステムテンプレートに定義された構成要素全てを配置するのに要する時間を推定する。その時間は「 t_2 」と呼ばれる。管理プログラム2410は、所定の配備時間(例えば、アプリケーションサーバの配備は1分かかる)または以前に行われた平均配備時間に基づいて、 t_2 を推定することができる。10070において、管理プログラム2410は、 $t_1 + t_2$ が所定のタイムアウトよりも短い構成のバリエーションが存在するか否かを判断する。その結果が「いいえ」の場合、次にプロセスは10080のフローに進む。その結果が「はい」の場合、次にプロセスは10090のフローに進む。

10

【0073】

10080において、管理プログラム2410は、構成のバリエーション及び合計時間(すなわち、 $t_1 + t_2$)をユーザに通知して、彼らに構成のバリエーションを1つ選択させる。10090において、管理プログラム2410は、構成のバリエーションに基づいてITリソースを作成して、アプリケーションを配備する。管理プログラム2410は、ストレージアレイテーブル2510、ストレージボリュームテーブル2520、物理サーバテーブル2530、仮想サーバテーブル2540、及びマッピングテーブル2550内のアプリケーションからITリソースまでのマッピング情報で作成されたITリソースの記録を作成する。10100において、管理プログラム2410は、アプリケーションの配備プロセスを修了する。

20

【0074】

図16は、実装形態例に従って、アプリケーションの完成度を計算するための管理プログラム2410のフローチャートを例示する。手順が開始されると、フローが11010で始まる。11020において、管理プログラム2410は、アプリケーション開発リポジトリ1100からログを検索する。このログは、ファイル(例えば、ソースコード、バイナリまたは構成ファイル)をリポジトリにコミットするログ、アプリケーションのID及び各コミットのタイムスタンプを含むことができる。11030において、管理プログラム2410は、1日当たりの変更数及び1日当たりの全変更量を計算する。前者はコミットの数に基づいて計算され、後者はソースコードの変更された行の数に基づいて計算される。11040において、管理プログラム2410は、計算された1日当たりの変更数と、1日当たりの全変更量を比較することによって、アプリケーションの完成度をアプリケーションの完成度モデルで推定する。11050において、管理プログラム2410は、アプリケーションの完成度が高いか低いかを判断する。これは、11040のフローで推定された完成度を所定の閾値と比較することによって行われ得る。11060において、管理プログラム2410はプロセスを修了する。

30

【0075】

図17は、実装形態例に従って、アプリケーションの完成度モデルを例示する。アプリケーションの完成度モデルの2つの実施例が図17にある。各モデルは、1日当たりの変更数及び1日当たりの全変更量の組み合わせと、アプリケーションの完成度レベルとの間の関係性を示す。

40

【0076】

図18は、実装形態例に従って、構成のバリエーションを作成するための管理プログラム2410のフローチャートを例示する。手順が始まると、フローが12010で開始される。12020において、管理プログラム2410は指定されたサイズのサイズ定義を取得する。12030において、管理プログラム2410はリソースグループテーブル2580を参照することにより、アプリケーションの完成度に基づいて、利用可能なリソースを特定する。図12に示すリソースグループテーブル2580の実施例に基づいて、アプリケーションの完成度が高い場合、その結果ストレージアレイ01、02及び03、

50

及び物理サーバ02及び03が利用可能となる。管理プログラム2410は、ストレージアレイテーブル2510及びストレージボリュームテーブル2520から、利用可能なストレージアレイ及びこれらのストレージアレイによって保持されたストレージボリュームの詳細と、物理サーバテーブル2530及び仮想サーバテーブル2540から、利用可能な物理サーバ及びこれらの物理サーバ上で動作している仮想サーバの詳細とを取得することができる。

【0077】

12040において、管理プログラム2410はアプリケーションの完成度によって、タイムアウト定義テーブル2590からタイムアウト値を取得する。図13に示すタイムアウト定義テーブル2590の実施例に基づいて、アプリケーションの完成度が高い場合、その結果タイムアウト値は120秒に構成される。12050において、管理プログラム2410はタイムアウト内で指定されたサイズに必要なリソース全てをセットアップするために構成のバリエーションを作成する。管理プログラム2410は、サーバ及びストレージアレイの性能情報に基づいて構成のバリエーションを作成して、各サーバ及びストレージアレイの負担のバランスを取ることができる。管理プログラム2410は、記憶メディアのタイプまたはサーバあるいはストレージアレイの容量に基づいて構成のバリエーションを作成してもよい。

10

【0078】

例えば、管理プログラム2410は、図10のストレージ性能テーブル2560から最低平均プロセッサの使用量を有する1つ以上のストレージアレイ、及び図11のサーバ性能テーブル2570から最低平均CPUの使用量を有する1つ以上の物理サーバを取り上げる。

20

【0079】

管理プログラム2410は、ストレージアレイ01及び02と、物理サーバ01を取得すると仮定する。次いで、管理プログラム2410は、1つ以上のプロセッサ、キャッシュ、平均使用量に基づいて選択されたストレージアレイ(上記の実施例では01及び02)のプール及びポートを取得する。管理プログラム2410は、プロセッサ、キャッシュ、選択されたストレージアレイ及び選択された物理サーバのプール及びポートの組み合わせ全てに基づいて、構成のバリエーションを作成する。

30

【0080】

構成のバリエーション例は、以下のようなものであってもよい。

・構成のバリエーション#1

物理サーバ01上の(1CPUコア及び2048GB RAMを備えた)VMは、ポートAを介してストレージアレイ01の(100GBの容量を備えた)ストレージボリュームに接続され、かつプロセッサP-02は、VMからストレージボリュームへのI/Oを制御する。ストレージボリュームはPool-01から湾曲し、かつキャッシュC-01はストレージボリュームに割り振られる。

・構成のバリエーション#2

物理サーバ01上の(1CPUコア及び2048GB RAMを備えた)VMは、ポートCを介してストレージアレイ01の(100GBの容量を備えた)ストレージボリュームに接続され、かつプロセッサP-03は、VMからストレージボリュームへのI/Oを制御する。ストレージボリュームはPool-02から湾曲し、かつキャッシュC-02はストレージボリュームに割り振られる。

40

・構成のバリエーション#3

物理サーバ01上の(1CPUコア及び2048GB RAMを備えた)VMは、ポートBを介してストレージアレイ02の(100GBの容量を備えた)ストレージボリュームへの接続。かつプロセッサP-01は、VMからストレージボリュームへのI/Oを制御する。ストレージボリュームはPool-01から湾曲し、かつキャッシュC-01はストレージボリュームに割り振られる。

【0081】

50

12060において、管理プログラム2410はプロセスを終了する。

【0082】

図19は、実装形態例に従って、ITリソースを準備するための管理プログラム2410のフローチャートを例示する。このプロセスは、図15のアプリケーションのシーケンスを配備することとは無関係に、かつ並行して実行され得る。フローは1310において開始され、手順が始まる。13020において、管理プログラム2410はシステムテンプレート全てを読み出す。13030において、管理プログラム2410は管理プログラム2410によって確認されていないシステムテンプレートが残っているか否かを判断する。その結果が「はい」の場合、次にプロセスは13040のフローに進む。その結果が「いいえ」の場合、次にプロセスは13080のフローに進む。

10

【0083】

13040において、管理プログラム2410は未確認のシステムテンプレートを選択して、確認したときに印を付ける。13050において、システムテンプレートに定義された各サイズについて、管理プログラム2410は構成のバリエーションを作成するサブシーケンスを呼び出す。このフローでは、管理プログラム2410はアプリケーションの完成度が低いと仮定する。13060において、システムテンプレートに定義された各サイズについて、管理プログラム2410は構成のバリエーションを作成するサブシーケンスを呼び出す。このフローでは、管理プログラム2410はアプリケーションの完成度が高いと仮定する。13070において、各構成のバリエーションについて、管理プログラム2410は作成されなければならないリソースを特定して作成に要する時間(t_1)を推定する。作成されなければならないリソースを特定し、かつ t_1 を推定するための方法は、図15の10050におけるフローと同じである。

20

【0084】

13080において、管理プログラム2410は構成のバリエーション全ての結合を計算して、よく使用されるリソースを特定する。管理プログラム2410は、リソースが所定の数よりも多いバリエーションに出現する場合、リソースがよく使用されると判定することができる。管理プログラム2410は、アプリケーション開発リポジトリ1100に格納されているアプリケーション全ての完成度に基づいて、リソースの出現回数を重み付けすることができる。低い完成度のアプリケーションが10個、高い完成度のアプリケーションが5個ある場合、次に管理プログラム2410は、13050のフローで作成された構成のバリエーションに出現するリソースの数を2倍にカウントすることができる。

30

【0085】

構成のバリエーション例は、以下のようなものであってもよい。

・構成のバリエーション#1

物理サーバ01上の(1CPUコア及び2048GB RAMを備えた)VMは、ポートAを介してストレージレイ01の(100GBの容量を備えた)ストレージボリュームに接続され、かつプロセッサP-02は、VMからストレージボリュームへのI/Oを制御する。ストレージボリュームはPool-01から湾曲し、かつキャッシュC-01はストレージボリュームに割り振られる。

・構成のバリエーション#2

40

物理サーバ01上の(1CPUコア及び2048GB RAMを備えた)VMは、ポートCを介してストレージレイ01の(100GBの容量を備えた)ストレージボリュームに接続され、かつプロセッサP-03は、VMからストレージボリュームへのI/Oを制御する。ストレージボリュームはPool-02から湾曲し、かつキャッシュC-02はストレージボリュームに割り振られる。

・構成のバリエーション#3

物理サーバ01上の(2CPUコア及び4096GB RAMを備えた)VMは、ポートBを介してストレージレイ02の(200GBの容量を備えた)ストレージボリュームへの接続。かつプロセッサP-01は、VMからストレージボリュームへのI/Oを制御する。ストレージボリュームはPool-01から湾曲し、かつキャッシュC-0

50

1 はストレージボリュームに割り振られる。

【 0 0 8 6 】

上記の例によれば、管理プログラム 2 4 1 0 は、所定の出現の数が 2 である場合、よく使用されるリソースとして「物理サーバ 0 1 上の (1 C P U コア及び 2 0 4 8 G B R A M を備えた) V M 」及び「ストレージレイ 0 1 の (1 0 0 G B の容量を備えた) ストレージボリューム」を決定してもよい。

【 0 0 8 7 】

1 3 0 9 0 において、よく使用される各リソースについて、管理プログラム 2 4 1 0 は t 1 が所定の値よりも大きい場合、リソースを作成して、作成されたリソースの予約されたフラグを 0 に構成する。これは、作成されたリソースが特定のユーザまたは特定のシステム用に予約されていないため、あらゆるユーザが作成されたリソースを使用することができることを表す。作成されたリソースがユーザによって使用されるようになると、予約されたフラグは 1 に構成される必要がある。1 3 1 0 0 において、管理プログラム 2 4 1 0 はプロセスを終了する。

10

【 0 0 8 8 】

図 2 0 は、実装形態例に従って、IT リソースを削除するための管理プログラム 2 4 1 0 のフローチャートを例示する。セルフサービスポータル 1 3 0 0 は、アプリケーション及び基礎となる IT リソースを削除するための GUI を有することができる。フローは 1 4 0 1 0 で開始され、手順が始まる。1 4 0 2 0 において、管理プログラム 2 4 1 0 はアプリケーションを削除するための要求をセルフサービスポータル 1 3 0 0 から受け付ける。1 4 0 3 0 において、管理プログラム 2 4 1 0 は指定されたアプリケーション及びアプリケーションを配備するために使用されるシステムテンプレート内に含まれる構成要素全てを削除する。1 4 0 4 0 において、管理プログラム 2 4 1 0 は指定されたアプリケーションに使用されるリソース全て及びシステムテンプレート内に含まれる構成要素全てをリストアップする。このプロセスは、マッピングテーブル 2 5 5 0 を参照することによって行われる。図 9 の行 2 5 5 B に示すアプリケーションが削除指定されている場合、管理プログラム 2 4 1 0 は指定されたアプリケーションに使用されるリソースとして、仮想サーバポート A、ストレージレイ 0 2 のストレージポート B、ストレージレイ 0 1 のストレージポート A、ストレージレイ 0 2 のストレージボリューム 0 5 2 及びストレージレイ 0 1 のストレージボリューム 0 5 5 をリストアップする。

20

30

【 0 0 8 9 】

1 4 0 5 0 において、管理プログラム 2 4 1 0 は未確認のリソースが残っているか否かを判断する。その結果が「はい」の場合、次にプロセスは 1 4 0 6 0 のフローに進む。その結果が「いいえ」の場合、次にプロセスは 1 4 0 9 0 のフローに進む。

【 0 0 9 0 】

1 4 0 6 0 において、管理プログラム 2 4 1 0 は未確認のリソースを選択して確認したときに印を付ける。1 4 0 7 0 において、管理プログラム 2 4 1 0 はそのリソースがよく使用されるか否かを判断する。このプロセスは、1 4 0 4 0 のフローでリスト化されたリソースの情報を図 1 9 の 1 3 0 8 0 のフローで決定されたよく使用されるリソースの情報と比較することによって行われ得る。1 4 0 4 0 のフローでリスト化されたリソースは、ストレージレイテーブル 2 5 1 0、ストレージボリュームテーブル 2 5 2 0、物理サーバテーブル 2 5 3 0、及び仮想サーバテーブル 2 5 4 0 から取得されてもよい。

40

【 0 0 9 1 】

例えば、管理プログラム 2 4 1 0 が「物理サーバ 0 1 上の (1 C P U コア及び 2 0 4 8 G B R A M を備えた) V M 」及び「ストレージレイ 0 1 の (1 0 0 G B の容量を備えた) ストレージボリューム」を、図 1 9 の 1 3 0 8 0 におけるフローでよく使用されるリソースとして決定し、かつステップ 1 4 0 4 0 でリストアップされたリソースが、ストレージレイ 0 1 の (1 0 0 G B の容量を備えた) ストレージボリュームを含む場合、管理プログラム 2 4 1 0 はそのストレージボリュームがよく使用されると判断する。

【 0 0 9 2 】

50

その結果が「はい」の場合、次にプロセスは14050のフローに進む。その結果が「いいえ」の場合、次にプロセスは14080のフローに進む。14080において、管理プログラム2410はリソースを削除する。14090において、管理プログラム2410はプロセスを終了する。

【0093】

この第1の実施形態例では、管理プログラムはITリソースを作成して、アプリケーションを、そのアプリケーションの完成度レベルに基づいてITリソース上に配備する。管理プログラムはまた、よく使用されるITリソースを準備して、さらにあまり使用されないリソースのみを削除する。このような実装形態によって、管理プログラムは、アプリケーションをアジャイル手法で適切なITリソース上に配備することができる。

10

【0094】

第2の実施形態例では、管理プログラムは、アプリケーションの完成度及び使用頻度を考慮に入れながらアプリケーションのサイズを変更するために、ITリソースの構成を変更する。

【0095】

図21は、第2の実施形態例に従って、管理コンピュータ2000'の構成を例示する。管理コンピュータ2000'の構成は、図2の管理コンピュータ2000の構成に類似しているが、以下に示す差異がある。

【0096】

管理コンピュータ2000'は、第1の実施形態例からのリソースグループテーブル2580とは異なる構造を有するリソースグループテーブル2580'、及び第1の実施形態例のタイムアウト定義テーブル2590とは異なる構造を有するタイムアウト定義テーブル2590'を有する。管理コンピュータ2000'は、アプリケーションをサイズ変更するために、アプリケーションの使用量の監視及びITリソースの構成の変更を含む追加機能を有する管理プログラム2410'を有する。アプリケーションの使用量の監視については、既存の技術が使用可能である。

20

【0097】

例示的な実装形態では、1つ以上のアプリケーションの完成度レベルと、1つ以上の使用頻度と、1つ以上のタイムアウトとの関係性は、タイムアウト定義テーブル2590'で概要が説明されており、プロセッサ2200は、1つ以上の構成の各々の配備に要する時間の計算によって、配備されるまたはサイズ変更されるアプリケーションに対するITリソースに基づく1つ以上の構成、及び配備されるまたはサイズ変更されるアプリケーションの、アプリケーションの完成度レベル、配備されるまたはサイズ変更されるアプリケーションの1つ以上の使用頻度、及び1つ以上のアプリケーションの完成度レベルと1つ以上のタイムアウトとの間の関係性に基づく1つ以上の構成に対するタイムアウトの決定を生成するように構成されている。決定されたタイムアウトを超過して配備に要する時間中、プロセッサ2200は、タイムアウトが超過しているという表示を提供するように構成され、プロセッサ2200は、配備に要する時間が以下に説明するように決定されたタイムアウトを超過していないとき、配備されるまたはサイズ変更されるアプリケーションの1つ以上の構成に対して指定されたITリソースを割り振るように構成されている。

30

40

【0098】

図22は、実装形態例に従って、リソースグループテーブル2580'を例示する。このテーブルは、管理プログラム2410によってメモリ2500内に作成される。列2581'はリソースタイプを示す。列2582'は、リソースのIDを示す。列2583'は、対応するアプリケーションの完成度を示す。列2584'は、対応するアプリケーションの使用頻度を示す。

【0099】

各行(258A'、258B'、258C'、258D'、258E'、258F'、258G')は、完成度の定義及び使用頻度に割り当てられたリソースを示す。例えば、行258A'は、ストレージアレイ01が「高い」完成度及び「高い」使用頻度に割り

50

当てられていることを示す。これは、ストレージレイ01が高い完成度及び高い使用頻度を有するアプリケーションにのみ使用され得ることを意味する。図22に示すリソースグループテーブルの実施例は、ストレージレイ及び物理サーバのみを含むが、このテーブルは、所望の実装形態に応じて他のタイプのリソースを含んでいてもよい。

【0100】

図23は、実装形態例に従って、タイムアウト定義テーブル2590'を例示する。このテーブルは、管理プログラム2410によってメモリ2500内に作成される。列2591'は、アプリケーションの完成度レベルを示す。列2592'は、アプリケーションの使用頻度を示す。列2593'は、アプリケーションの配備プロセス中に構成のバリエーションを作成するためのタイムアウトを示す。各行(259A'、259B'、259C'、259D')は、各組の完成度レベル及び使用頻度に定義されているタイムアウト値を示す。例えば、行259A'は、低い完成度及び低い使用頻度を有するアプリケーションに定義された60秒のタイムアウト値を示す。このタイムアウト値は、所望の実装形態に応じて、他の方法で表示され得る。これらは、作成される構成のバリエーションの数として表示され得る。

【0101】

図24は、実装形態例に従って、セルフサービスポータル1600のGUI1600-Bを例示する。このGUIは、アプリケーション開発者1010またはアプリケーション管理者1020がアプリケーションのサイズを変更したときに使用される。アプリケーション開発者またはアプリケーション管理者は、アプリケーションリスト1610-Bからアプリケーションを選択する。選択されたアプリケーションの配備に使用されるID、名前、テンプレート及びサイズを1620-B、1630-B、1640-B及び1650-Bでそれぞれ示す。アプリケーション開発者またはアプリケーション管理者は、新たなサイズをドロップダウンリスト1650-Bから選択する。図24は、「モバイルアプリA」という名前のアプリケーションが選択されてそのサイズが「小」から「中」に変更されている場合の実施例を示す。配備ボタン1660-Bがクリックされると、管理プログラム2410'は指定されたアプリケーションのサイズ変更をする。取り消しボタン1670-Bがクリックされると、管理プログラム2410'はアプリケーションのサイズ変更プロセスを取り消す。

【0102】

図25は、実装形態例に従って、アプリケーションをサイズ変更するための管理プログラム2410'のフローチャートを例示する。このフローは、20010で開始されて、手順が始まる。20020において、管理プログラム2410'はアプリケーションをサイズ変更するための要求をセルフサービスポータル1600から受け付ける。図24に示すパラメータが管理プログラム2410'に渡される。20030において、管理プログラム2410'は、アプリケーションが配備されたときに計算されるアプリケーションの完成度を取得する。管理プログラム2410'はまた、それ自身の監視機能を介して、アプリケーションの使用頻度を取得する。20040において、管理プログラム2410'は以下により詳細に開示される「構成のバリエーションの作成」のサブシーケンスを呼び出す。20050、20060、20070、20080、20090及び20100のフローは、図15の10050、10060、10070、10080、10090及び10100のフローとそれぞれ同様である。

【0103】

図26は、実装形態例に従って、構成のバリエーションを作成するための管理プログラム2410'のフローチャートを例示する。図26に示すフローチャートは、図18に示すフローチャートと類似している。これら2つの違いは22030及び22040のプロセスである。

【0104】

22010、22020、22060及び22060のフローは、図18の12010、12020、12050及び12060のフローとそれぞれ同様である。22030にお

10

20

30

40

50

いて、管理プログラム 2410' はリソースグループテーブル 2580' を参照することによってアプリケーションの完成度レベル及び使用頻度に基づいて利用可能なリソースを識別する。22040において、管理プログラム 2410' は、選択されたアプリケーションの完成度レベル及び使用頻度によってタイムアウト定義テーブルからタイムアウト値を取得する。

【0105】

したがって、第2の実施形態例では、管理プログラムは、アプリケーションの完成度及び使用頻度を考慮してアプリケーションをサイズ変更するために、構成のバリエーションを作成する。このような実装形態によって、管理プログラムは、アプリケーションをアジャイル手法で、サイズを適切なITリソース上で変更することができる。

10

【0106】

詳細な記述の一部は、コンピュータ内の動作のアルゴリズム及び記号表現の観点から提示されている。これらのアルゴリズム的記述及び記号表現は、データ処理技術の当業者が彼らの技術革新の神髄を他の当業者に伝えるために使用される手段である。アルゴリズムは、所望の終了状態または結果をもたらす一連の定義されたステップである。実装形態例では、具体的な結果を達成するための具体的な量の物理的操作を必要とする。

【0107】

特に断りのない限り、説明から明らかなように、この説明を通して、「処理」、「演算」、「計算」、「決定」、「表示」などの用語を用いる説明は、コンピュータシステムまたはコンピュータシステムのレジスタ及びメモリ内で物理（電子）量として表示されるデータを操作して、コンピュータシステムのメモリまたはレジスタあるいは他の情報記憶機器、変換機器または表示機器内で物理量として同じように表示される他のデータに変換する他の情報処理機器のアクション及びプロセスを含み得ることが理解される。

20

【0108】

実装形態例はまた、本明細書の動作を行うための装置に関連していてもよい。この装置は、必要な目的のために特別に構築されてもよく、または1つ以上のコンピュータプログラムによって選択的に起動または再構成された1つ以上の汎用コンピュータを含んでもよい。このようなコンピュータプログラムは、コンピュータ可読記憶媒体またはコンピュータ可読信号媒体などのコンピュータ可読媒体内に格納され得る。コンピュータ可読記憶媒体は、光ディスク、磁気ディスク、読み出し専用メモリ、ランダムアクセスメモリ、固体デバイス及びドライブなどの有形的表現媒体、または電子情報を格納するのに適したあらゆる他のタイプの有形的または非一時的媒体を含んでもよいが、その限りではない。コンピュータ可読信号媒体は、搬送波などの媒体を含むことができる。本明細書に提示されたアルゴリズム及びディスプレイは、特定のコンピュータまたは他の装置に本質的に関連するものではない。コンピュータプログラムは、所望の実装の動作を行う命令を含む純粋なソフトウェアの実装を含むことができる。

30

【0109】

様々な汎用システムは、本明細書の実施例に従ってプログラム及びモジュールと共に使用されてもよく、または、所望の方法ステップを行うためにより専門化された装置を構築するのに好都合であることが判明し得る。加えて、実装形態例は、特定のプログラム言語を参照して記載されていない。本明細書に記載された実装形態例の教示を実施するために様々なプログラム言語を使用され得ることが理解されるであろう。プログラム言語の命令は、1つ以上の処理機器、例えば、中央処理装置（CPU）、プロセッサ、またはコントローラによって実施され得る。

40

【0110】

当技術分野で知られているように、上述の動作は、ハードウェア、ソフトウェア、またはソフトウェアとハードウェアのいくつかの組み合わせによって行うことができる。実装例の様々な態様は、回路及び論理デバイス（ハードウェア）を用いて実装することができるが、他の態様はプロセッサによって実施される場合、本出願の実装を実行するための

50

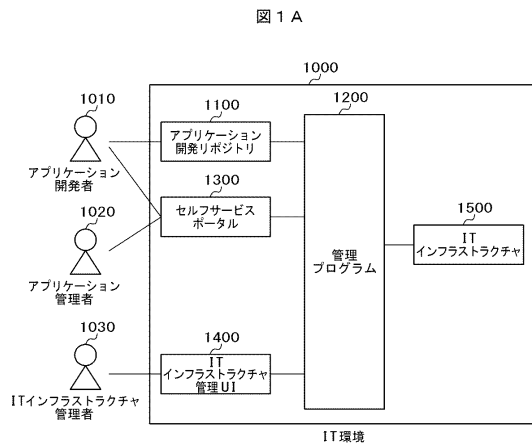
方法を当該プロセッサに行わせることになる、機械可読媒体に格納された命令を用いて実装されてもよい。さらにまた、本出願のいくつかの実装例は、ハードウェアでのみ実行されてもよく、他の実装例は、ソフトウェアでのみ実行されてもよい。そのうえ、説明された様々な機能は、単一のユニット内で行うことができ、またはいろいろな方法で複数の構成要素にわたって分散されることができる。ソフトウェアによって行われるとき、本方法は、コンピュータ可読媒体に格納されている命令に基づいて、汎用コンピュータなどのプロセッサによって実施することができる。必要に応じて、この命令は、圧縮された及び/または暗号化されたフォーマットで媒体に格納することができる。

【0111】

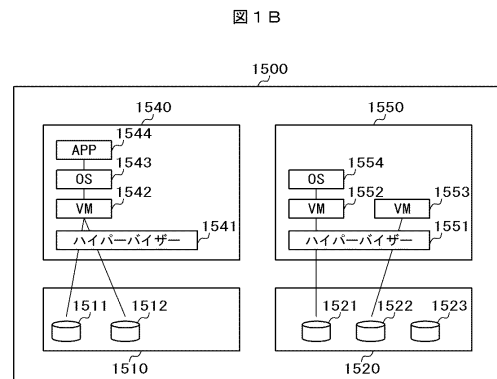
さらに、本出願の他の実装形態は、本明細書の考察及び本出願の教示の履行から当業者には明らかであろう。説明された実装例の様々な態様及び/または構成要素は、単独で、または任意の組み合わせで使用され得る。本明細書及び実装例は、単なる例として考えられるものであり、本出願の真の範囲及び精神は、以下の特許請求の範囲によって表されることを意図している。

10

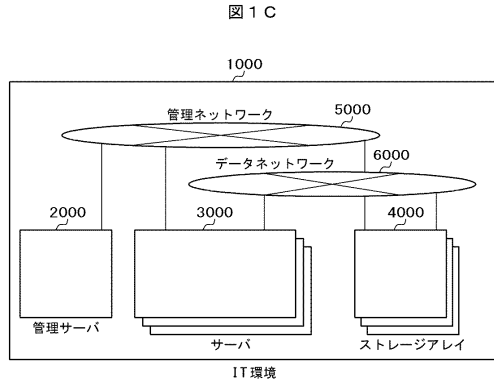
【図1A】



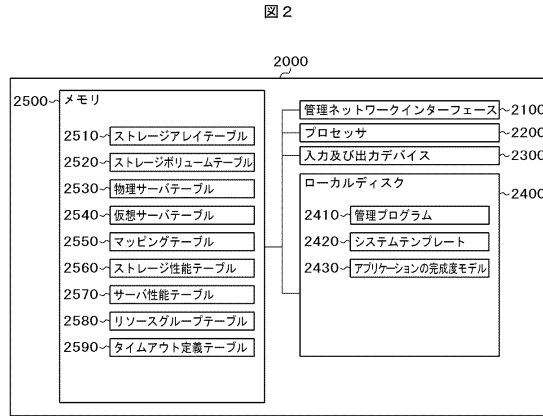
【図1B】



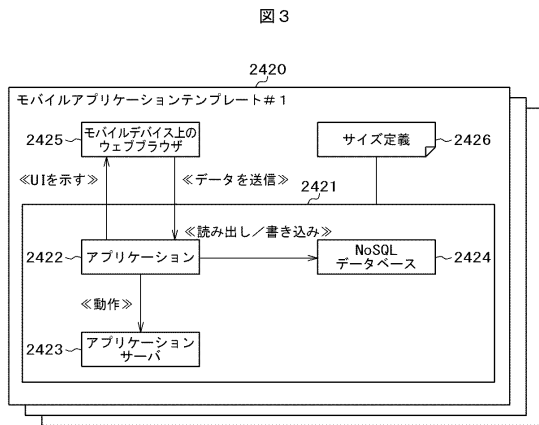
【図1C】



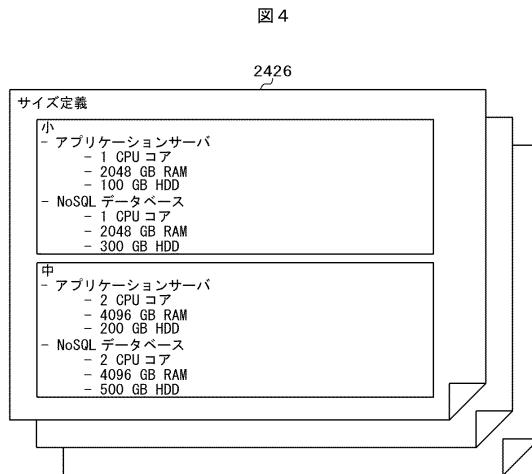
【図2】



【図3】



【図4】



【図 1 2】

図 1 2

	2581	2582	2583
	リソースタイプ	リソース ID	アプリケーションの完成度
258A	ストレージアレイ	01	高い
258B	ストレージアレイ	02	高い
258C	ストレージアレイ	03	高い
258D	物理サーバ	01	低い
258E	物理サーバ	02	低い/高い
258F	物理サーバ	03	低い/高い
258G	物理サーバ	04	低い
	⋮	⋮	⋮

【図 1 3】

図 1 3

	2591	2592
	アプリケーションの完成度	タイムアウト
259A	低い	60
259B	高い	120

【図 1 4】

図 1 4

1600-A

1610-A アプリケーション:

1620-A アプリケーション名:

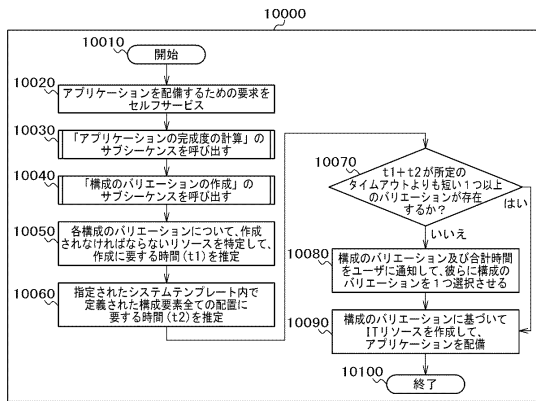
1630-A テンプレート:

1640-A サイズ:

1650-A 1660-A

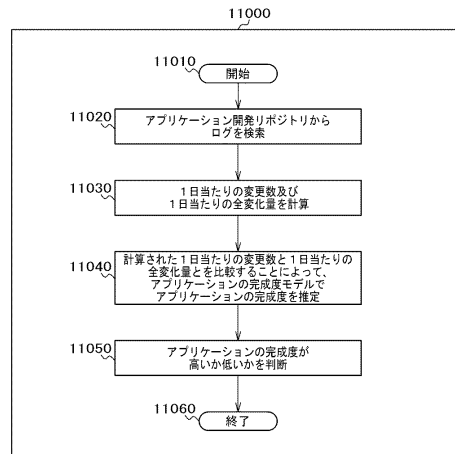
【図 1 5】

図 1 5



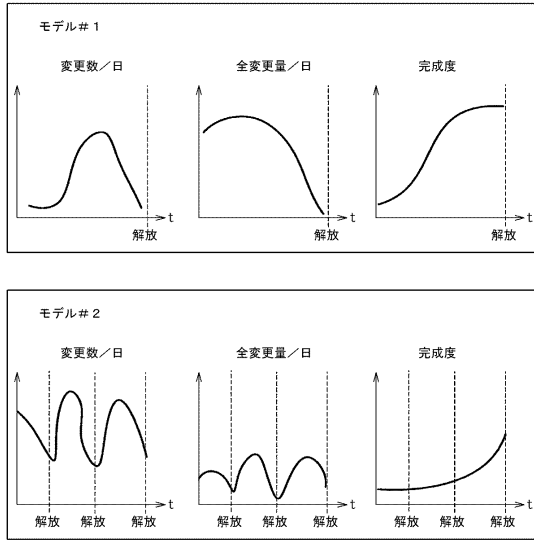
【図 1 6】

図 1 6



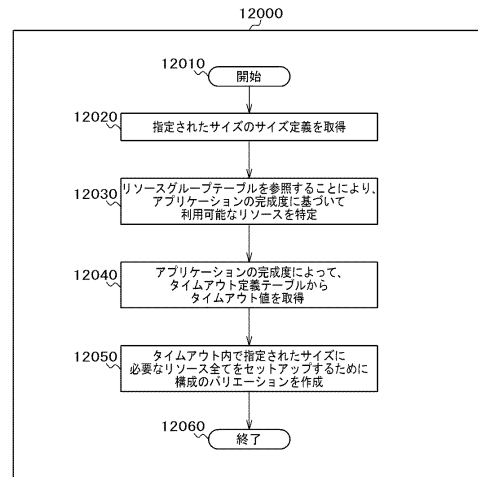
【図17】

図17



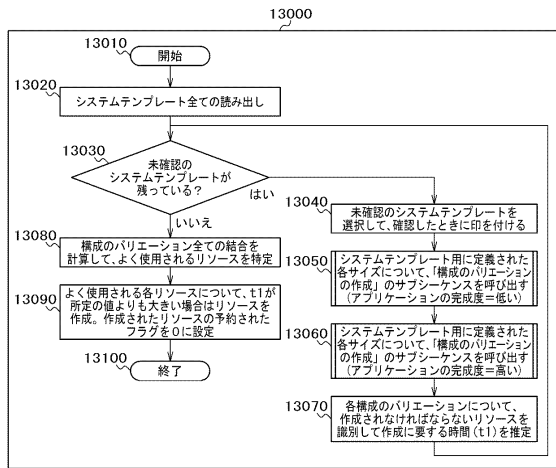
【図18】

図18



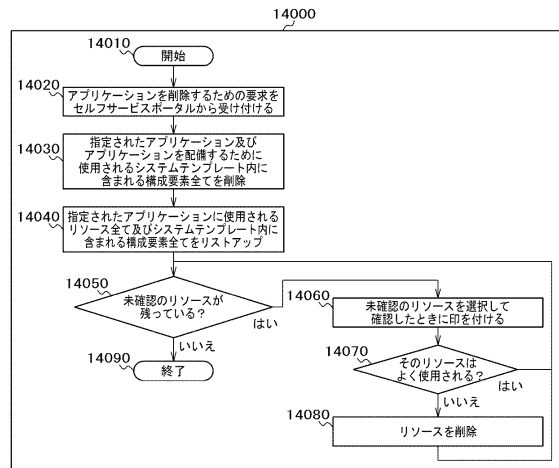
【図19】

図19

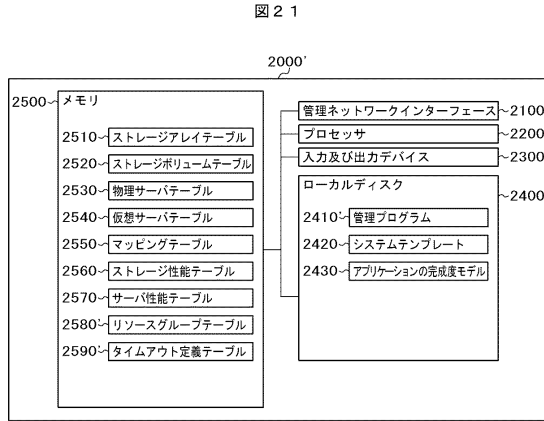


【図20】

図20



【図 2 1】



【図 2 2】

図 2 2

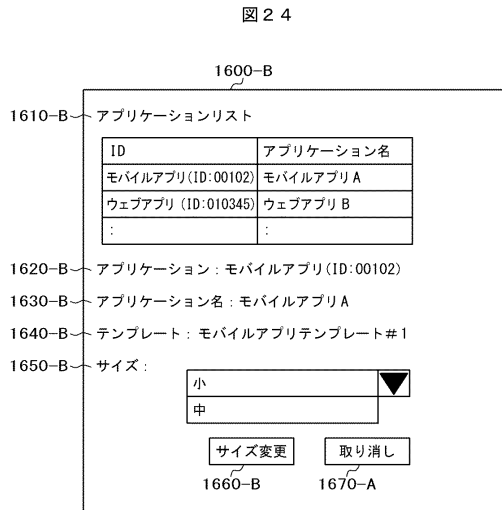
	2581'	2582'	2583'	2584'
	リソースタイプ	リソース ID	アプリケーションの完成度	使用頻度
258A'	ストレージレイ	01	高い	高い
258B'	ストレージレイ	02	高い	高い
258C'	ストレージレイ	03	高い	低い/高い
258D'	物理サーバ	01	低い	低い
258E'	物理サーバ	02	低い/高い	低い
258F'	物理サーバ	03	低い/高い	低い/高い
258G'	物理サーバ	04	低い	低い
	:	:	:	:

【図 2 3】

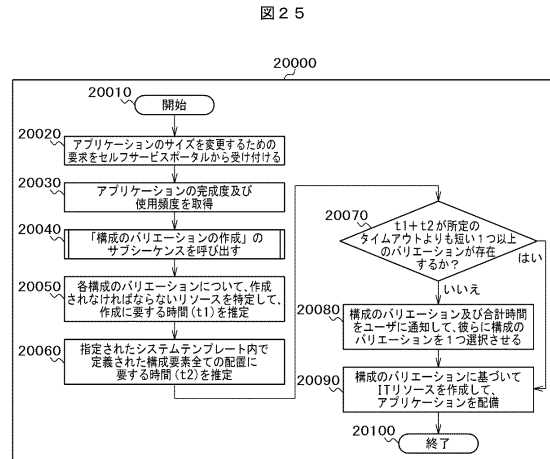
図 2 3

	2591'	2592'	2593'
	アプリケーションの完成度	使用頻度	タイムアウト
259A'	低い	低い	60
259B'	低い	高い	120
259C'	高い	低い	60
259D'	高い	高い	120

【図 2 4】

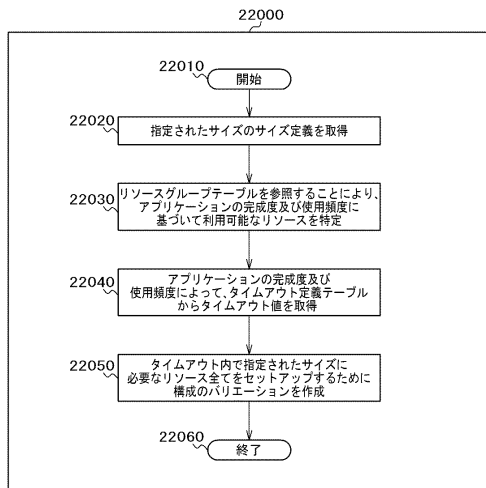


【図 2 5】



【 図 2 6 】

図 2 6



フロントページの続き

- (56)参考文献 国際公開第2014/002421(WO, A1)
特開2011-180751(JP, A)
特開2011-70394(JP, A)
特開2010-113381(JP, A)
国際公開第2016/016975(WO, A1)
国際公開第2007/096963(WO, A1)
特表2015-529918(JP, A)
国際公開第2013/018914(WO, A1)
宮川伸也、外3名、ビジネスグリッド技術解説、情報処理、日本、社団法人情報処理学会、2006年9月15日、第47巻、第9号、pp.953-961
三浦謙一、サイエンスグリッドの動向、情報処理、日本、社団法人情報処理学会、2010年2月15日、第51巻、第2号、pp.114-119

(58)調査した分野(Int.Cl., DB名)

G06F 8/00 - 8/77
G06F 9/44 - 9/54