

(19)日本国特許庁(JP)

## (12)特許公報(B2)

(11)特許番号  
特許第7598129号  
(P7598129)

(45)発行日 令和6年12月11日(2024.12.11)

(24)登録日 令和6年12月3日(2024.12.3)

(51)国際特許分類 F I  
G 0 6 Q 10/0633(2023.01) G 0 6 Q 10/0633

請求項の数 20 (全22頁)

(21)出願番号	特願2021-569980(P2021-569980)	(73)特許権者	520262319 ユーアイパス、インコーポレイテッド UiPath, Inc. アメリカ合衆国、ニューヨーク 100 17、ニューヨーク、ヴァンダービルト アベニュー 1、シックスティース フロア 1 Vanderbilt Avenue , 60th Floor, New Yo rk, NY 10017, United States of America
(86)(22)出願日	令和2年9月16日(2020.9.16)	(74)代理人	100180781 弁理士 安達 友和
(65)公表番号	特表2023-517404(P2023-517404 A)	(74)代理人	100182903 弁理士 福田 武慶
(43)公表日	令和5年4月26日(2023.4.26)	(72)発明者	コピエール、マルティン
(86)国際出願番号	PCT/US2020/050932		
(87)国際公開番号	WO2021/183180		
(87)国際公開日	令和3年9月16日(2021.9.16)		
審査請求日	令和5年9月14日(2023.9.14)		
(31)優先権主張番号	16/816,184		
(32)優先日	令和2年3月11日(2020.3.11)		
(33)優先権主張国・地域又は機関	米国(US)		

最終頁に続く

(54)【発明の名称】 プロセスのためのボトルネック検出

## (57)【特許請求の範囲】

## 【請求項1】

ボトルネックを識別するためにプロセスの複数の実行インスタンスについてイベントログを解析するためのコンピュータ実装方法であって、

前記複数の実行インスタンスのうちの1つまたは複数の間に実行されたセグメントを前記イベントログから識別するステップであって、前記セグメントの各々が前記プロセスの一对のアクティビティを表す、ステップと、

前記識別されたセグメントの特定のセグメントごとに、

前記イベントログに基づいて前記特定のセグメントの前記1つまたは複数の実行インスタンスの各々の性能の尺度を計算するステップと、

前記計算された性能の尺度に基づいて前記特定のセグメントの前記1つまたは複数の実行インスタンスの各々を分類するステップと、

前記特定のセグメントの前記分類された1つまたは複数の実行インスタンスに基づいて、前記特定のセグメントについての1つまたは複数のメトリックを計算するステップと、

ボトルネックを有する可能性が最も高い前記識別されたセグメントのうちの1つを識別するために、前記1つまたは複数のメトリックに基づいて前記識別されたセグメントを互いに比較するステップと、を含む、コンピュータ実装方法。

## 【請求項2】

前記一对のアクティビティは、送信元アクティビティおよび宛先アクティビティを含み、前記宛先アクティビティの実行が、前記イベントログにおける前記送信元アクティビテ

10

20

イの実行に直接続く、請求項 1 に記載のコンピュータ実装方法。

【請求項 3】

前記性能の尺度は、前記一对のアクティビティ間の時間間隔である、請求項 1 に記載のコンピュータ実装方法。

【請求項 4】

前記計算された性能の尺度に基づいて前記特定のセグメントの前記 1 つまたは複数の実行インスタンスの各々を分類するステップは、

J e n k s 自然分類最適化アルゴリズムを用いて、前記特定のセグメントの前記 1 つまたは複数の実行インスタンスの各々を複数のクラスのうちの一つに分類するステップと、を含む、請求項 1 に記載のコンピュータ実装方法。

10

【請求項 5】

前記特定のセグメントの前記分類された 1 つまたは複数の実行インスタンスに基づいて、前記特定のセグメントについての 1 つまたは複数のメトリックを計算するステップは、

1) 前記特定のセグメントのクラス間の平均差を表す前記特定のセグメントについての効果量メトリック、2) 前記特定のセグメントの最良性能クラスと他のクラスとの間の時間差を表す前記特定のセグメントについての損失時間メトリック、および 3) 前記特定のセグメントの全体性能に対する寄与を表す重み付き影響メトリックを計算するステップを含む、請求項 1 に記載のコンピュータ実装方法。

【請求項 6】

ボトルネックを有する可能性が最も高い前記識別されたセグメントのうちの一つを識別するために、前記 1 つまたは複数のメトリックに基づいて前記識別されたセグメントを互いに比較するステップは、

20

前記 1 つまたは複数のメトリックに基づいて前記識別されたセグメントをランク付けするステップを含む、請求項 5 に記載のコンピュータ実装方法。

【請求項 7】

前記 1 つまたは複数のメトリックに基づいて前記識別されたセグメントをランク付けするステップは、

前記効果量メトリック、前記損失時間メトリック、および前記重み付き影響メトリックの各々について前記識別されたセグメントの個々のランク付けを決定するステップと、

前記個々のランク付けに基づいて前記識別されたセグメントの総合ランク付けを決定するステップと、を含む、請求項 6 に記載のコンピュータ実装方法。

30

【請求項 8】

前記総合ランク付けに基づいて、前記ボトルネックを有する可能性が最も高い前記識別されたセグメントのうちの前記 1 つを識別するステップをさらに含む、

請求項 7 に記載のコンピュータ実装方法。

【請求項 9】

前記ボトルネックを識別するために、前記比較の結果を表示させるステップをさらに含む、

請求項 1 に記載のコンピュータ実装方法。

【請求項 10】

40

前記プロセスは、ロボティックプロセスオートメーションプロセスである、請求項 1 に記載のコンピュータ実装方法。

【請求項 11】

ボトルネックを識別するためにプロセスの複数の実行インスタンスについてイベントログを解析するためのコンピュータ命令を格納するメモリと、

前記コンピュータ命令を実行するように構成された少なくとも一つのプロセッサと、を含み、前記コンピュータ命令は、前記少なくとも一つのプロセッサに、

前記複数の実行インスタンスのうちの一つまたは複数の間に実行されたセグメントを前記イベントログから識別するステップであって、前記セグメントの各々が前記プロセスの一对のアクティビティを表す、ステップと、

50

前記識別されたセグメントの特定のセグメントごとに、  
 前記イベントログに基づいて前記特定のセグメントの前記1つまたは複数の実行インスタンスの各々の性能の尺度を計算するステップと、  
 前記計算された性能の尺度に基づいて前記特定のセグメントの前記1つまたは複数の実行インスタンスの各々を分類するステップと、  
 前記特定のセグメントの前記分類された1つまたは複数の実行インスタンスに基づいて、前記特定のセグメントについての1つまたは複数のメトリックを計算するステップと、  
 ボトルネックを有する可能性が最も高い前記識別されたセグメントのうちの1つを識別するために、前記1つまたは複数のメトリックに基づいて前記識別されたセグメントを互いに比較するステップと、

10

の動作を実行させるように構成される、装置。

【請求項12】

前記一对のアクティビティは、送信元アクティビティおよび宛先アクティビティを含み、前記宛先アクティビティの実行が、前記イベントログにおける前記送信元アクティビティの実行に直接続く、請求項11に記載の装置。

【請求項13】

前記性能の尺度は、前記一对のアクティビティ間の時間間隔である、請求項11に記載の装置。

【請求項14】

前記計算された性能の尺度に基づいて前記特定のセグメントの前記1つまたは複数の実行インスタンスの各々を分類するステップは、

20

Jenks自然分類最適化アルゴリズムを用いて、前記特定のセグメントの前記1つまたは複数の実行インスタンスの各々を複数のクラスの中の1つに分類するステップと、を含む、請求項11に記載の装置。

【請求項15】

ボトルネックを識別するためにプロセスの複数の実行インスタンスについてイベントログを解析するための、非一時的コンピュータ可読媒体上に格納されたコンピュータプログラムであって、前記コンピュータプログラムは、少なくとも1つのプロセッサに、

前記複数の実行インスタンスのうちの1つまたは複数の間に実行されたセグメントを前記イベントログから識別するステップであって、前記セグメントの各々が前記プロセスの一对のアクティビティを表す、ステップと、

30

前記識別されたセグメントの特定のセグメントごとに、前記イベントログに基づいて前記特定のセグメントの前記1つまたは複数の実行インスタンスの各々の性能の尺度を計算するステップと、

前記計算された性能の尺度に基づいて前記特定のセグメントの前記1つまたは複数の実行インスタンスの各々を分類するステップと、

前記特定のセグメントの前記分類された1つまたは複数の実行インスタンスに基づいて、前記特定のセグメントについての1つまたは複数のメトリックを計算するステップと、

ボトルネックを有する可能性が最も高い前記識別されたセグメントのうちの1つを識別するために、前記1つまたは複数のメトリックに基づいて前記識別されたセグメントを互いに比較するステップと、

40

を含む動作を実行させるように構成される、コンピュータプログラム。

【請求項16】

前記特定のセグメントの前記分類された1つまたは複数の実行インスタンスに基づいて、前記特定のセグメントについての1つまたは複数のメトリックを計算するステップは、

1)前記特定のセグメントのクラス間の平均差を表す前記特定のセグメントについての効果量メトリック、2)前記特定のセグメントの最良性能クラスと他のクラスとの間の時間差を表す前記特定のセグメントについての損失時間メトリック、および3)前記特定のセグメントの全体性能に対する寄与を表す重み付き影響メトリックを計算するステップを含む、請求項15に記載のコンピュータプログラム。

50

## 【請求項 17】

ボトルネックを有する可能性が最も高い前記識別されたセグメントのうちの1つを識別するために、前記1つまたは複数のメトリックに基づいて前記識別されたセグメントを互いに比較するステップは、

前記1つまたは複数のメトリックに基づいて前記識別されたセグメントをランク付けするステップを含む、請求項16に記載のコンピュータプログラム。

## 【請求項 18】

前記1つまたは複数のメトリックに基づいて前記識別されたセグメントをランク付けするステップは、

前記効果量メトリック、前記損失時間メトリック、および前記重み付き影響メトリックの各々について前記識別されたセグメントの個々のランク付けを決定するステップと、

前記個々のランク付けに基づいて前記識別されたセグメントの総合ランク付けを決定するステップと、を含む、請求項17に記載のコンピュータプログラム。

10

## 【請求項 19】

前記動作は、

前記総合ランク付けに基づいて、前記ボトルネックを有する可能性が最も高い前記セグメントのうちの前記1つを識別するステップをさらに含む、請求項18に記載のコンピュータプログラム。

## 【請求項 20】

前記プロセスは、ロボティックプロセスオートメーションプロセスである、請求項15に記載のコンピュータプログラム。

20

## 【発明の詳細な説明】

## 【技術分野】

## 【0001】

(関連出願の相互参照)

本出願は、2020年3月11日に提出された米国実用特許出願第16/816,184号の優先権を主張し、その開示はその全体が参照により本明細書に組み込まれる。

## 【0002】

本発明は、一般に、プロセスマイニングに関し、より詳細には、プロセスのボトルネック検出に関する。

30

## 【背景技術】

## 【0003】

プロセスは、製品またはサービスを提供するために実行される一連のアクティビティである。プロセスマイニングでは、効率を改善し、プロセスのより良い理解を得るために、傾向、パターン、および他のプロセス解析尺度を識別するためにプロセスが解析される。しかしながら、現在のプロセスマイニング技術は、プロセスにおけるボトルネックの識別を可能にするためのプロセスの性能に関する十分な解析データを提供しない。そのようなボトルネックは、プロセスの性能を制限し、不必要な遅延、収益の損失、およびユーザの不満をもたらす可能性がある。

## 【発明の概要】

40

## 【0004】

1つまたは複数の実施形態によれば、ボトルネックを識別するためにプロセスを解析するためのシステムおよび方法が提供される。イベントログは、プロセスの複数の実行インスタンスの間に維持される。イベントログは、複数の実行インスタンスの間に実行されたセグメントを識別し、セグメントの各々はプロセスの一对のアクティビティを表す。識別されたセグメントの各々のメトリックが計算され、ボトルネックを有する可能性が最も高い識別されたセグメントのうちの1つが、メトリックに基づいて識別される。プロセスは、ロボティックプロセスオートメーションプロセスであってもよい。

## 【0005】

一実施形態では、プロセスの複数の実行インスタンスについてのイベントログが受信さ

50

れ、複数の実行インスタスのうちの1つまたは複数の間に実行されたセグメントがイベントログから識別される。セグメントは、プロセスの対のアクティビティを表し、対のアクティビティは、送信元アクティビティおよび宛先アクティビティを含み、宛先アクティビティの実行は、イベントログにおける送信元アクティビティの実行に直接続く。識別されたセグメントの各特定のセグメントについて、イベントログに基づいて特定のセグメントの1つまたは複数の実行インスタスの各々について性能の尺度が計算され、計算された性能の尺度に基づいて特定のセグメントの1つまたは複数の実行インスタスの各々が分類され、特定のセグメントの分類された1つまたは複数の実行インスタスに基づいて特定のセグメントについての1つまたは複数のメトリックが計算される。識別されたセグメントは、1つまたは複数のメトリックに基づいて互いに比較され、ボトルネックを有する可能性が最も高い識別されたセグメントのうちの1つを識別する。

10

**【0006】**

一実施形態では、性能の尺度は対のアクティビティ間の時間間隔である。

**【0007】**

一実施形態では、特定のセグメントの1つまたは複数の実行インスタスの各々は、Jenks自然分類最適化アルゴリズムを使用して複数のクラスのうちの1つに分類される。特定のセグメントの1つまたは複数のメトリックは、1)特定のセグメントのクラス間の平均差を表す特定のセグメントについての効果量メトリック、2)特定のセグメントの最良性能クラスと他のクラスとの間の時間差を表す特定のセグメントについての損失時間メトリック、および3)特定のセグメントの全体性能に対する寄与を表す重み付き影響メトリックを計算することによって計算される。

20

**【0008】**

一実施形態では、識別されたセグメントは、1つまたは複数のメトリックに基づいて識別されたセグメントをランク付けすることによって互いに比較される。識別されたセグメントは、効果量メトリック、損失時間メトリック、および重み付き影響メトリックの各々について識別されたセグメントの個々のランク付けを決定し、個々のランク付けを組み合わせることによって識別されたセグメントの総合ランク付けを決定することによってランク付けされ得る。ボトルネックを有する可能性が最も高いセグメントの1つが、総合ランク付けに基づいて識別され得る。

**【0009】**

一実施形態では、比較の結果が表示される。

30

**【0010】**

本発明のこれらおよび他の利点は、以下の詳細な説明および添付の図面を参照することによって当業者に明らかになるであろう。

**【図面の簡単な説明】****【0011】**

【図1】本発明の1つまたは複数の実施形態による、ボトルネック検出のために解析され得る例示的なプロセスを示す図である。

**【0012】**

【図2】本発明の1つまたは複数の実施形態による、ボトルネック検出のためのプロセスを解析するための方法を示す図である。

40

**【0013】**

【図3】本発明の1つまたは複数の実施形態による、図1のプロセスの例示的なイベントログを示す図である。

**【0014】**

【図4】本発明の1つまたは複数の実施形態による、セグメントをランク付けするための例示的な表を示す図である。

**【0015】**

【図5】本発明の1つまたは複数の実施形態による、セグメント概要のダッシュボードを示す図である。

50

【 0 0 1 6 】

【 図 6 】 本発明の 1 つまたは複数の実施形態による、プロセスの解析の時間的観点のダッシュボードを示す図である。

【 0 0 1 7 】

【 図 7 】 本発明の 1 つまたは複数の実施形態による、プロセスの解析の事例的観点のダッシュボードを示す図である。

【 0 0 1 8 】

【 図 8 】 本発明の 1 つまたは複数の実施形態による、プロセスの解析の組織的観点のダッシュボードを示す図である。

【 0 0 1 9 】

【 図 9 】 1 つまたは複数の実施形態による、プロセスの解析の制御フロー的観点のダッシュボードを示す図である。

10

【 0 0 2 0 】

【 図 1 0 】 1 つまたは複数の実施形態による、プロセスの解析の制御フロー的観点からエッジ横断率を示すプロセスの結合ビューのダッシュボードを示す図である。

【 0 0 2 1 】

【 図 1 1 】 1 つまたは複数の実施形態による、プロセスの解析の制御フロー的観点からの平均スループット時間を示すプロセスの結合ビューのダッシュボードを示す図である。

【 0 0 2 2 】

【 図 1 2 】 本発明の一実施形態によるコンピューティングシステムのブロック図である。

20

【 発明を実施するための形態 】

【 0 0 2 3 】

プロセスは、いくつかの異なるアプリケーションのための製品またはサービスを提供するために利用され得る。そのようなプロセスの例示的なアプリケーションには、管理アプリケーション（例えば、新しい従業員の新人研修）、調達支払アプリケーション（例えば、購入、請求書管理、および支払いの円滑化）、および情報技術アプリケーション（例えば、発券システム）が含まれる。例示的なプロセス 1 0 0 が図 1 に示されている。プロセス 1 0 0 は、請求書を処理および支払いするためのビジネスプロセスである。一実施形態では、プロセス 1 0 0 は、1 つまたは複数の R P A ロボットを使用してタスクを自動的に実行するためのロボティックプロセスオートメーション（R P A）ワークフローとして実施されてもよい。

30

【 0 0 2 4 】

プロセス 1 0 0 は、プロセス 1 0 0 における所定のシーケンスのステップを表すアクティビティ 1 0 2 ~ 1 1 4 を含む。図 1 に示すように、プロセス 1 0 0 は、各アクティビティ 1 0 2 ~ 1 1 4 がノードとして表され、アクティビティ 1 0 2 ~ 1 1 4 間の各遷移がノードをリンクするエッジとして表される有向グラフとしてモデル化されている。アクティビティ間の遷移は、送信元アクティビティから宛先アクティビティへのプロセス 1 0 0 の実行を表す。プロセス 1 0 0 は、請求書受信アクティビティ 1 0 2 で開始し、受信請求書確認アクティビティ 1 0 4 に進む。受信した請求書が、受信請求書確認アクティビティ 1 0 4 において情報が欠落していると判定された場合には、プロセス 1 0 0 は、請求書最終確認アクティビティ 1 1 0 に進む前に、データ要求アクティビティ 1 0 6 および契約条件確認アクティビティ 1 0 8 に進む。受信した請求書が、受信請求書確認アクティビティ 1 0 4 において情報が欠落していないと判定された場合には、プロセス 1 0 0 は、請求書最終確認アクティビティ 1 1 0 に直接進む。次に、プロセス 1 0 0 は、請求書承認アクティビティ 1 1 2 および請求書支払アクティビティ 1 1 4 に進む。プロセス 1 0 0 の実行は、イベントログの形態で記録される。

40

【 0 0 2 5 】

時には、プロセス 1 0 0 の実行がボトルネックに起因して妨げられることがある。本明細書で使用される場合、ボトルネックは、プロセス 1 0 0 の実行に悪影響を及ぼすプロセス 1 0 0 の一組の 1 つまたは複数のアクティビティを指す。プロセス 1 0 0 の実行は、例

50

例えば、スループット（すなわち、実行）時間、リソース使用量、収益、サポート解決時間、または任意の他の性能インジケータなどの任意の適切なメトリックに基づいて悪影響を受ける可能性がある。例えば、プロセス100の実行は、最大スループット時間、最小リソース使用量、最小収益成長率、収益成長率、サポート解決時間などの任意のメトリックに基づいて悪影響を受ける可能性がある。そのようなボトルネックは、プロセス100の実行を制限し、不必要な遅延、収益の損失、およびユーザの不満をもたらす可能性がある。

#### 【0026】

本発明の実施形態によれば、プロセス（例えば、プロセス100）は、プロセスにおけるボトルネックを識別するために解析される。ボトルネックの根本原因の特定を容易にするために、解析の様々な態様を1つまたは複数のダッシュボードで視覚化することができる。有利には、本発明の実施形態によるボトルネックを識別するためのプロセスの解析は、そのようなボトルネックの根本原因を軽減または排除することを可能にし、それによってプロセスの効率および性能を改善する。

#### 【0027】

図2は、1つまたは複数の実施形態による、ボトルネック識別のためのプロセスを解析するための方法200を示す。図2は、引き続き図1のプロセス100を参照して説明される。一実施形態では、方法200のステップは、ボトルネック検出のためにプロセス100を解析するために実行される。方法200のステップは、例えば図12のコンピュータ1200などの任意の適切なコンピューティングデバイスによって実行することができる。

#### 【0028】

ステップ202において、プロセスの複数の実行インスタンスについてのイベントログが受信される。イベントログは、プロセスの複数の実行インスタンスの間に発生するイベントを記録することによって、プロセスの複数の実行インスタンスの間に維持されてもよい。イベントは、特定の時間および特定の事例におけるアクティビティの実行を指す。事例は、プロセスの単一の実行インスタンスに対応し、事例識別子（ID）によって識別される。一実施形態では、各イベントは、実行されたアクティビティのラベル、アクティビティの実行のタイムスタンプ、および実行されたアクティビティの実行インスタンスを識別する事例IDを含むタプルとして表すことができる。

#### 【0029】

図3は、1つまたは複数の実施形態による、図1のプロセス100の例示的なイベントログ300を示す。イベントログ300は、イベントログ300内の事例ID1および事例ID2に対応する、プロセス100の2つの実行インスタンス中に発生したイベントを記録する。図3に示すように、イベントログ300は、各々がイベントに対応する行302と、各々が行302と列304とが交差するセルにおけるイベントの属性を識別する列304と、を有する表としてフォーマットされる。具体的には、各行302は、アクティビティ102~114の実行を表すイベント（列304-Bで識別される）、アクティビティ102~114の実行のタイムスタンプ（列304-Cで識別される）、および実行されたアクティビティ102~114の実行インスタンスを識別する事例ID（列304-Aで識別される）に関連する。一実施形態では、列304-Cで識別されるアクティビティ102-114の実行のタイムスタンプは、アクティビティ102-114の実行が完了した時刻を指すが、あるいはアクティビティ104-114の実行が開始された時刻を指してもよい。一実施形態では、イベントログ300は、図3に示すように、各イベントについて、実行されたアクティビティのラベル、アクティビティの実行のタイムスタンプ、および実行されたアクティビティの実行インスタンスを識別する事例IDのみを識別する。しかしながら、イベントログ300は、任意の適切なフォーマットであってもよく、イベントの他の属性を識別する追加の列304を含んでもよいことを理解されたい。

#### 【0030】

ステップ204において、プロセスの複数の実行インスタンスのうちの1つまたは複数の間に実行されたセグメントがイベントログから識別される。各セグメントは、プロセス

10

20

30

40

50

における一対のアクティビティを表す。一対のアクティビティは、本明細書では<送信元アクティビティ、宛先アクティビティ>として示される送信元アクティビティおよび宛先アクティビティを含み、宛先アクティビティの実行はイベントログ内の送信元アクティビティの実行に直接続く。セグメントは、一対のアクティビティを識別するためにイベントログにおける各実行インスタンスをトラバースすることによって識別される。一例では、図3のイベントログ300において、以下のセグメントが識別され得る。すなわち、<請求書受信、受信請求書確認>、<受信請求書確認、請求書最終確認>、<受信請求書確認、データ要求>、<データ要求、契約条件確認>、<契約条件確認、請求書最終確認>、<請求書最終確認、請求書承認>、および<請求書承認、請求書支払>である。

#### 【0031】

ステップ206において、イベントログに基づいて、識別されたセグメントのうちの特定のセグメントの1つまたは複数の実行インスタンスの各々について、性能の尺度が計算される。一実施形態では、特定のセグメントの特定の実行インスタンスの性能の尺度は、その特定の実行インスタンスのセグメントの送信元アクティビティと宛先アクティビティとの間の時間間隔である。時間間隔は、イベントログから特定の実行インスタンスについての送信元アクティビティの実行のタイムスタンプおよび宛先アクティビティの実行のタイムスタンプを抽出し、宛先アクティビティの実行のタイムスタンプと送信元アクティビティの実行のタイムスタンプとの間の差を判定することによって計算され得る。例えば、事例ID1の実行インスタンスのセグメント<受信請求書確認、請求書最終確認>に関連する時間間隔は、図3のイベントログ300から2日、3時間、および10秒として計算され得る。別の例では、事例ID1の実行インスタンスのセグメント<請求書最終確認、請求書承認>に関連する時間間隔は、図3のイベントログ300から1日、20時間、35分、および3秒として計算され得る。時間間隔は、例えば、秒、分、時間、日、月、および/または年などの任意の適切なフォーマットであってもよい。例えば、リソース使用量、収益値、占有率、あるいは数値に基づいており、送信元アクティビティおよび宛先アクティビティに対して個別に計算/測定することができる任意の他の適切な尺度などの、他の性能の尺度も使用することができる。

#### 【0032】

ステップ208において、特定のセグメントの1つまたは複数の実行インスタンスの各々は、計算された性能の尺度に基づいて複数のクラスのうちの一つに分類される。一実施形態では、特定のセグメントの1つまたは複数の実行インスタンスの各々は、各クラス内の変動を最小化することによって機能する周知のJenks自然分類最適化アルゴリズムを使用して、複数のクラスのうちの一つに分類される。Jenks自然分類最適化アルゴリズムは、George F. Jenksによる「Optimal Data Classification for Choropleth Maps」に記載されている。任意の他の適切なデータクラスタリングアルゴリズムを使用することもできる。

#### 【0033】

Jenks自然分類最適化アルゴリズムは、特定のセグメントの1つまたは複数の実行インスタンスの性能の尺度を含むデータセットQを、クラスを描写する異なるブレイクポイントを使用してデータセットを反復的に分割することによって、同様の性能のn個のクラス $Q_i$ に分割する。例えば、7600、3480、220、500、1500、および1700の時間間隔を有する6つの実行インスタンスの間に実行されるセグメント<請求書受信、受信請求書確認>を考える。所与のデータセット

$Q = [7600, 3480, 220, 500, 1500, 1700]$ の場合、Jenks自然分類最適化アルゴリズムは、Qのn個の所定数のクラスへの最適分割を作成しようと試みる。この例では、 $n = 3$ である。Jenks自然分類最適化アルゴリズムは、以下のようにステップ1~5を適用する。

#### 【0034】

ステップ1：データセットQをソートし、任意に（例えば、ランダムに）ブレイクポイントを選択する。したがって、ソートされたデータセット $Q = [220, 500, 150$

10

20

30

40

50

0, 1700, 3480, 7600]

任意に選択されたブレイクポイントにより、クラス  $Q_{bp} = [220]$ 、 $[500, 1500, 1700, 3480]$ 、 $[7600]$  が得られる。

【0035】

ステップ2：各クラス  $Q_{bp}$  についての配列平均 (SDAM) についての偏差の二乗和を計算する。SDAMは、式1に従って計算することができる。

【数1】

$$SDAM = \sum_{i=1}^N (Q_i - \mu)^2, \text{ ここで、 } \mu = \text{mean}(Q). \quad (\text{式1})$$

10

ソートされたデータセット  $Q$  に式1を適用すると、以下が得られる。

$$SDAM = (220 - 2500)^2 + (500 - 2500)^2 + (1500 - 2500)^2 + (1700 - 2500)^2 + (3480 - 2500)^2 + (7600 - 2500)^2 = 37,808,800$$

【0036】

ステップ3：各クラス  $Q_{bp}$  についてクラス間の偏差の二乗和 (SDBC) を計算する。SDBCは、式2に従ってクラス  $Q_{bp}$  ごとに計算することができる。

【数2】

$$SDBC = \sum_{i=1}^N (Q_{bpi} - \mu)^2, \text{ ここで、 } \mu = \text{mean}(Q_{bp}). \quad (\text{式2})$$

20

式2を各クラス  $Q_{bp}$  に適用すると、以下が得られる。

$$SDBC = (220 - 220)^2 + \{ (500 - 1795)^2 + (1500 - 1795)^2 + (1700 - 1795)^2 + (3480 - 1795)^2 \} + (7600 - 7600)^2 = 4,612,300$$

【0037】

ステップ4：各クラス  $Q_{bp}$  についてクラス平均 (SDCM) の偏差の二乗和を計算する。SDCMは、式3に従ってクラス  $Q_{bp}$  ごとに計算することができる。

$$SDCM = SDAM - SDBC. \quad (\text{式3})$$

式3を適用すると、 $SDCM = 37,808,800 - 4,612,300 = 33,196,500$  となる。

【0038】

ステップ5：すべての可能なブレイクポイントの組み合わせについてステップ2～4を繰り返す。SDCMが最も高いブレイクポイントの組み合わせが選択される。したがって、特定のセグメントの1つまたは複数の実行インスタンスの各々は、最も高いSDCMとのブレイクポイントの組み合わせによって定義される複数のクラスのうちの1つに分類される。

30

40

【0039】

一実施形態では、クラスの最適な数  $n$  は、異なる数のクラスのデータセット  $Q$  に対して Jenks 自然分類最適化アルゴリズムを繰り返し実行することによって決定することができる。例えば、Jenks 自然分類最適化アルゴリズムは、 $n = 1, \dots, x$  のいくつかのクラスに対して繰り返し実行されてもよく、 $x$  はデータセット  $Q$  内の項目の数である。この例では、 $x = 6$  である。Jenks 自然分類最適化アルゴリズムは、クラスの数ごとに適合度 (GVF) 尺度を提供する。GVF測定値は、 $GVF = (SDAM - SDCM) / SDAM$  として計算することができ、0のGVF尺度は適合なしを示し、1のGVF尺度は完全な適合を示す。GVFを最大化すると、データセット  $Q$  内の項目の数と等し

50

いクラスの最適な数が常にもたらされる（すなわち、GVF = 1）。したがって、n個のクラスのGVFとn + 1個のクラスのGVFとの間の変化率を示す適合度変化（RGVFC）尺度が計算される。n個のクラスのRGVFCは、 $RGVFC = (GVF_{n+1} - GVF_n) / GVF_n$ として計算される。最適なクラス数は、最大GVF値を有するがRGVFCしきい値を超えないRGVFC値を有するクラス数nとして選択される。一実施形態では、RGVFCしきい値は2.5% ~ 10%であるが、任意の適切な値が用いられてもよい。

【0040】

ステップ210において、分類された特定のセグメントの1つまたは複数の実行インスタンスに基づいて、特定のセグメントについて1つまたは複数のメトリックが計算される。1つまたは複数のメトリックは、特定のセグメントを特徴付ける任意の適切なメトリックを含むことができる。一実施形態では、1つまたは複数のメトリックは、効果量、損失時間、および重み付き影響を含む。例えば、クラスサイズ、クラス方向（クラスの増加/減少）、クラス分布などの任意の他の適切なメトリックも使用することができる。

10

【0041】

効果量：効果量は、特定のセグメントにおける2つのクラス間の差の定量的尺度である。一実施形態では、効果量は、周知のCohenのd法に従って計算されるが、任意の他の適切な手法を使用してもよい。Cohenのd法は、Jacob Cohenによる「Statistical Power Analysis for the Behavioral Sciences」に記載されている。クラスX<sub>1</sub>およびX<sub>2</sub>のCohenのd値は、式4に従って計算される。

20

【数3】

$$d = \frac{\bar{x}_1 - \bar{x}_2}{s} = \frac{\mu_1 - \mu_2}{s} \quad (\text{式4})$$

ここで、μ<sub>1</sub>およびμ<sub>2</sub>はそれぞれX<sub>1</sub>およびX<sub>2</sub>の平均であり、sはプールされた標準偏差である。プールされた標準偏差sは、式5で定義される。

【数4】

30

$$s = \sqrt{\frac{(n_1 - 1)s_1^2 + (n_2 - 1)s_2^2}{n_1 + n_2 - 2}} \quad (\text{式5})$$

ここで、n<sub>1</sub>およびn<sub>2</sub>はそれぞれX<sub>1</sub>およびX<sub>2</sub>のサイズであり、

$$s_1^2$$

40

および

$$s_2^2$$

はそれぞれX<sub>1</sub>およびX<sub>2</sub>の変動であり、

$$s_1^2$$

および

50

$s_2^2$ 

はそれぞれ式 6 および式 7 で定義される。

【数 5】

$$s_1^2 = \frac{1}{n_1 - 1} \sum_{i=1}^{n_1} (x_{1,i} - \bar{x}_1)^2 \quad (\text{式 6})$$

$$s_2^2 = \frac{1}{n_2 - 1} \sum_{i=1}^{n_2} (x_{2,i} - \bar{x}_2)^2 \quad (\text{式 7})$$

10

【0042】

Cohen の d 値は、2つのクラスが異なる標準偏差の数を表す。例えば、1の Cohen の d 値は、2つのクラスが1標準偏差だけ異なることを示し、2の Cohen の d 値は、2つのクラスが2標準偏差だけ異なることを示すなどである。

【0043】

Cohen の d 法は、一度に2つのクラス間の効果量（すなわち、Cohen の d 値）のみを計算する。3つ以上のクラスを有し得る特定のセグメントの Cohen の d 値を計算するために、クラスのペアのすべての組み合わせの効果量が平均化される。特に、まず、N個のクラスを有する特定のセグメントについて、クラスのすべてのペア（ $T_i$ 、 $T_j$ ）が識別され、ここで  $i < j$  および  $i, j \leq N$  である。次に、クラスのすべてのペア（ $T_i$ 、 $T_j$ ）について Cohen の d 値が計算される。最後に、クラスのペア（ $T_i$ 、 $T_j$ ）ごとの Cohen の d 値の合計をクラスのペアの総数で除算して、特定のセグメントの Cohen の d 値を提供する。

20

【0044】

損失時間：損失時間は、特定のセグメント内の最良性能（例えば、最も速い）クラスと他のクラスとの間の時間差を表す。特定のセグメントの損失時間は、最初に特定のセグメント内の各クラス  $C_i$  の中央値を計算することによって計算され、 $i$  はクラス番号である。第2に、クラス1の中央値を最適時間（Jenks 自然分類最適化アルゴリズムの性質により、クラス1は常に最小中央値（すなわち、最も速い時間）を有する）とする。第3に、特定のセグメント内の各クラス  $C_i$  の損失時間は、 $C_i$  損失時間 = (中央値( $C_i$ ) - 最適時間) × サイズ( $C$ ) として計算される。最後に、各クラス  $C_i$  の損失時間は、

30

【数 6】

$$\sum_{i=1}^N C_i \text{ 損失時間}$$

として合計され、ここで N は特定のセグメント内のクラスの数であり、特定のセグメントの損失時間を提供する。

40

【0045】

クラスの中央値は、以下の2つの理由のために損失時間を計算するために使用される：1) Jenks 自然分類最適化アルゴリズムの性質のために、クラス1は常に最低の中央値を有する、および2) クラスの中央値は、クラスの平均と比較して不均衡なデータによる歪みが少ない。損失時間は、特定のセグメントの1つまたは複数の実行インスタンスが特定のセグメントにおける可能な最速時間よりも遅い大きさを表す。したがって、セグメント B と比較して損失時間が比較的小さいセグメント A は、セグメント B の改善に集中する方が収益性が高いことを示す。

【0046】

50

重み付き影響：重み付き影響は、特定のセグメントのセグメント性能によって引き起こされる特定のセグメントの全体性能の寄与率を表す。セグメント S の重み付き影響 (TWI) は、式 8 のように計算される。

【数 7】

$$TWI(S) = \sum_{i=1}^C \frac{SP(S_i) |S_i|}{OP(S_i) |S|} \quad (\text{式 8})$$

ここで、C はセグメント S 内のクラスの数であり、SP (S<sub>i</sub>) はセグメント S 内のクラス S<sub>i</sub> のセグメント性能であり、OP (S<sub>i</sub>) はセグメント S 内のクラス S<sub>i</sub> の全体性能であり、|S| はセグメント S を通過する事例の数を表し、|S<sub>i</sub>| はクラス S<sub>i</sub> 内の事例の数を表す。セグメント性能は、セグメント S の性能 (例えば、総時間間隔) である。全体性能は、セグメント S を有する事例の総スループット時間である。

【0047】

ステップ 212 で、識別されたセグメントの残りのセグメントがあるかどうか判定される。残りのセグメントがあると判定された場合には、方法 200 はステップ 206 に戻り、次の残りのセグメントを特定のセグメントとして使用してステップ 206 ~ 210 が繰り返される。したがって、ステップ 206 ~ 210 は、識別されたセグメントの各セグメントに対して実行される。残りのセグメントがないと判定された場合には、方法 200

【0048】

ステップ 214 において、識別されたセグメントは、識別されたセグメントのうちの少なくとも 1 つにおけるボトルネックを識別するために、1 つまたは複数のメトリックに基づいて互いに比較される。一実施形態では、識別されたセグメントは、1 つまたは複数のメトリックに基づいて識別されたセグメントの各々をランク付けすることによって互いに比較される。一実施形態では、識別されたセグメントの個々のランク付けは、1 つまたは複数のメトリックのそれぞれについて最初に決定される。例えば、識別されたセグメントの効果量ランク付けは、効果量メトリックに対して決定されてもよく、識別されたセグメントの損失時間ランク付けは、潜在的な損失時間メトリックに対して決定されてもよく、識別されたセグメントの重み付けされた影響ランク付けは、総重み付き影響メトリックに対して決定されてもよい。次いで、総合ランク付けが、1 つまたは複数のメトリックの個々のランク付けに基づいて計算される。例えば、総合ランク付けは、個々のランク付けの累積ランク付け、個々のランク付けの重み付けランク付け、または個々のランク付けを集約する任意の他の適切なランク付けを含むことができる。一実施形態では、累積ランク付けは、1 つまたは複数のメトリックの個々のランク付けの合計または平均として計算することができる。総合ランク付けは、セグメントが他のセグメントと比較してボトルネックを有する相対的尤度を表す。総合ランク付けが最も高いセグメントは、他のセグメントと比較してボトルネックを有する可能性が最も高いセグメントであり、累積ランク付けが最も低いセグメントは、他のセグメントと比較してボトルネックを有する可能性が最も低いセグメントである。識別されたセグメントを互いに比較することにより、ユーザによる識別されたセグメントにおけるボトルネックの識別が容易になる。例えば、最も高い総合ランク付けを有するセグメントは、ボトルネックを有する可能性が最も高いセグメントであり、これは、セグメントがボトルネックを有するかどうかを判定するためにユーザによるさらなる解析が推奨されることを示すことができる。いくつかの実施形態では、ランク付けはまた、運用コストに基づくことができ、ユーザは、潜在的なリスクが最も高いセグメントを見ることができる。

【0049】

図 4 は、1 つまたは複数の実施形態による、セグメントをランク付けするための例示的な表 400 を示す。表 400 の列 402 に示すように、イベントログでは、7 つのセグメ

ントが識別される。効果量、潜在的な損失時間、および総重み付き影響メトリックは、それぞれ列 4 0 4 , 4 0 6 および 4 0 8 に示すように、セグメントごとに計算される。セグメントの個々のランクは、それぞれ列 4 1 0 , 4 1 2 および 4 1 6 に示すように、効果量、潜在的な損失時間、および総重み付き影響メトリックのそれぞれについて決定される。セグメントの個々のランクは 1 ~ 7 の範囲であり、個々のランクが 1 のセグメントが最も低いランクのセグメントであり、個々のランクが 7 のセグメントが最も高いランクのセグメントである。例えば、セグメント < C , E > は、列 4 0 4 において最も低い効果量を有し、したがって、列 4 1 0 において最も低い ( 1 ) にランク付けされ、セグメント < A , B > は、列 4 0 4 において最も高い効果量を有し、したがって、列 4 1 0 において最も高い ( 7 ) にランク付けされる。

10

セグメント < D , E > は、列 4 0 6 において最も低い潜在的な損失時間を有し、したがって、列 4 1 2 において最も低い ( 1 ) にランク付けされ、セグメント < A , B > は、列 4 0 6 において最も高い潜在的な損失時間を有し、したがって、列 4 1 2 において最も高い ( 7 ) にランク付けされる。

セグメント < A , C > は、列 4 0 8 において最も低い総重み付き影響を有し、したがって、列 4 1 4 において最も低い ( 1 ) にランク付けされ、セグメント < B , C > は、列 4 0 8 において最も高い総重み付き影響を有し、したがって列 4 1 4 において最も高い ( 7 ) にランク付けされる。

列 4 1 6 に示すように、各メトリックの列 4 1 0 ~ 1 4 1 の個々のランク付けは、セグメントごとに加算されて累積ランク付けを計算する。

20

セグメント < A , B > は、累積ランク付けが最も高く、ボトルネックを有する可能性が最も高いセグメントとして識別される。いくつかの実施形態では、例えば、最も高い効果量、最も高い潜在的損失時間、および最も高い総重み付き影響を有するセグメントが最も低くランク付けされる場合、累積ランク付けが最も低いセグメントは、ボトルネックを有する可能性が最も高いセグメントであり得る。

#### 【 0 0 5 0 】

ステップ 2 1 6 において、ステップ 2 1 4 の比較の結果が出力される。例えば、比較の結果をコンピュータシステムのディスプレイデバイス上に表示することによって、比較の結果をコンピュータシステムのメモリまたはストレージ上に記憶することによって、または比較の結果をリモートコンピュータシステムに送信することによって、比較の結果を出力することができる。

30

#### 【 0 0 5 1 】

いくつかの実施形態では、比較の結果は、例えば、図 5 ~ 図 1 1 に示すダッシュボードなどの 1 つまたは複数のユーザインターフェースを介してディスプレイデバイスに識別されたセグメントの比較の結果を表示することによって出力されてもよい。

ダッシュボードは、例えば、時間的観点、事例的観点、組織的観点、および制御フロー的観点を含む、いくつかの観点からの結果を視覚化する。そのようなダッシュボードは、根本原因のボトルネック解析のためのセグメントの解析を容易にするための解析ツールをユーザに提供する。

#### 【 0 0 5 2 】

40

図 5 は、1 つまたは複数の実施形態による、セグメント概要のためのダッシュボード 5 0 0 を示す。ダッシュボード 5 0 0 は、各セグメントおよびそのメトリックの概要を領域 1 5 0 2 内のユーザに提示する。

領域 1 5 0 2 の概要は表形式で示されており、各行はセグメントに対応し、各列は属性に対応する。以下の属性、すなわち、セグメントの名称、そのセグメントにおける事例数、そのセグメントにおけるクラス数、効果量のランク付け、潜在的損失時間のランク付け、総重み付き影響のランク付け、および累積ランク付けが示される。領域 1 5 0 2 内のセグメントは、累積ランク付けに基づいて、最高から最低まで順序付けられている。しかしながら、領域 1 5 0 2 と対話するユーザは、(例えば、列をクリックすることにより) 任意の列の属性に基づいて表をソートすることができる。領域 2 5 0 4 は、(例え

50

ば、ホバリングすると)各メトリックがどのように計算されるかの説明を提供するアイコンを示す。

#### 【0053】

図6は、1つまたは複数の実施形態による、時間的観点からのダッシュボード600を示す。時間的観点は、イベントのタイミングおよび頻度に関する。タイミング情報を使用して、ボトルネックの発見、サービスレベルの測定などを行うことができる。ダッシュボード600は、選択されたセグメントの詳細なビューを提示する。領域1602は、ユーザがセグメントを選択することができるセレクタを示す。領域2604は積み重ねられたバーを示し、各セクション610、612、および614は、それぞれ効果量メトリック、損失時間メトリック、および重み付き影響メトリックの値を表す。領域2604は、選択されたセグメントが他のセグメントに対してどのようにランク付けされたかを迅速に確認し、選択されたセグメントに関する一般的な洞察を得る能力をユーザに提供する。領域3606は、クラスごとの、選択されたセグメントの各事例の総スループット時間の分布を示す。領域3606は、ユーザがクラスを視覚的に比較して、各クラスの全体性能の理解を生み出すことを可能にする。領域4608は、各クラスのスループット時間の中央値と、スループット時間がどのように細分化されるかを示す。特に、領域4608は、経過時間(事例の開始からセグメントに到達するまでの時間)、セグメント時間(事例がセグメントに費やされた時間)、および残り時間(セグメントを出るときと事例の終了時との間に事例が費やす時間)を示す。領域4608は、セグメント性能が全体性能にどのように関連するかを示すので、重み付き影響メトリックを視覚化し、それによってセグメントがプロセス全体に及ぼす影響に関する洞察をユーザに提供する。

10

20

#### 【0054】

図7は、1つまたは複数の実施形態による、事例の観点のためのダッシュボード700を示す。ダッシュボード700は、事例属性の解析を提示し、それにより、ユーザが、例えば、最も速いクラスと比較して最も遅いクラスの事例属性間の不一致を識別することを可能にし、それにより、不一致が存在する理由を明らかにすることができる。領域1702は、ユーザによって解析されるセグメントおよび解析されるクラスのセットを選択するためのセレクタを示す。領域2704は、解析が実行される事例属性を選択するための事例属性セレクタと、以前の研究に基づいて予想されたものとは著しく異なる属性のみを示すための「有意のみを示す」オプションと、を示す。領域3706は、線710が選択された事例属性の予想される頻度を表し、棒が選択された事例属性の観測された頻度を表すグラフを示す。領域4708は、選択された事例属性がすべてのクラスにどのように分布するかを示す表を示す。領域4708は、分布をパーセンテージまたは数(カウント)として表示するためのパーセンテージ/カウントセレクタを含む。

30

#### 【0055】

図8は、1つまたは複数の実施形態による、組織的観点からのダッシュボード800を示す。組織的観点は、リソース、すなわちどの当事者(例えば、人々、システム、役割、部門)が関与し、それらがどのように関連しているかに関する。領域1802は、解析するセグメントを選択するためのセレクタを示す。領域2804には、解析対象のイベント属性を選択するためのイベント属性セレクタと、分布をパーセンテージまたは数(カウント)として表示するためのパーセンテージ/カウントセレクタと、が示されている。領域3806は、セグメントアクティビティに基づくクラスごとの選択されたイベント属性の分布を示す表を示す。「パーセンテージ」が選択された場合、各行の合計は100%になり、ユーザはセグメントのイベント属性頻度を検査し、異なるクラス間のイベント属性頻度を比較することができる。領域4808は、プロセス全体のクラスごとの選択されたイベント属性の分布を示す表を示す。

40

#### 【0056】

図9は、1つまたは複数の実施形態による、制御フロー的観点からのダッシュボード900を示す。ダッシュボード900は、ユーザが、あるクラスのフローと他のクラスとの間の違いを発見することを可能にする。制御フロー的観点は、アクティビティの順序付け

50

に関する。領域 1 902 は、解析するセグメントを選択するためのセレクトアを示す。領域 2 904 は、プロセスフロー解析ツールを示す。特に、領域 2 904 は、クラスを選択するためのセレクトアを示す領域 2 a 906 と、すべてのエッジがそれを横切る事例の割合を示す選択されたクラスのプロセスフローを示す領域 2 b 908 と、プロセス利用率、事例数を説明する凡例を示す領域 2 c 910 と、を含み、まれなアクティビティおよびエッジを非表示または表示するためのしきい値を設定するためのユーザ調整可能スライダを含む。領域 3 912 は、別のプロセスフロー解析ツールを示す。領域 3 912 は、領域 3 a 914、領域 3 b 916、および領域 3 c 918 を含み、これらは領域 2 a 906、領域 2 b 908、および領域 2 c 910 と同様であるが、別のプロセス用である。領域 4 920 は、ダッシュボード 900 に示すように、プロセスを並べて表示するか、または組み合わせる表示するかを選択するためのボタンを示す。図 10 および図 11 は、1 つまたは複数の実施形態による、プロセスの合成ビューを示す制御フロー的観点からのダッシュボード 1000 およびダッシュボード 1100 をそれぞれ示す。プロセスは、プロセス A、プロセス B、および両方のプロセスが異なる色で表されるように、ダッシュボード 1000 およびダッシュボード 1100 に色で示されている。ダッシュボード 1000 内の各エッジは、色によって各プロセスのエッジ横断率を表すパーセンテージを含む。ダッシュボード 1100 内の各エッジは、色による各プロセスの平均スループット時間を含む。他の統計もまた、エッジ上に表され得る。

10

**【0057】**

図 2 の方法 200 に戻って参照すると、一実施形態では、ステップ 202 で受信されたイベントログは、図 3 のイベントログ 300 に示すように、実行されたアクティビティ、アクティビティの実行の単一のタイムスタンプ、および事例 ID のみを識別する標準フォーマットである。

20

しかしながら、他の実施形態では、ステップ 202 で受信されたイベントログは、非標準フォーマットであり、標準フォーマットに変換される。

**【0058】**

非標準フォーマットのイベントログの一例は、各イベントに対応する行と、各イベントの実行開始時刻のタイムスタンプおよび実行完了時刻のタイムスタンプを識別する列と、を含むイベントログである。このような非標準フォーマットのイベントログは、各イベントを実行開始時刻に対応する第 1 のイベントと実行完了時刻に対応する第 2 のイベントとの 2 つの別々のイベント（すなわち、2 つの別々の行）に分割することで、標準フォーマットに変換される。この例では、実行開始時刻と実行完了時刻とに基づいて、解析対象のセグメントのタイプを区別することができる。例えば、特定のアクティビティの実行開始時間に対応するイベントから、同じアクティビティの実行完了時間に対応するイベントまでのセグメントは、処理時間を表すことができ、アクティビティの実行完了時間に対応するイベントから、次のアクティビティの実行開始時間に対応するイベントまでのセグメントは、待ち時間を表すことができる。したがって、検出されたボトルネックは、タイプ（すなわち、処理時間に起因するか、待ち時間に起因するか）によって区別され得る。

30

**【0059】**

非標準フォーマットのイベントログの他の例は、各イベントに対応する行と、各イベントに対するアクティビティライフサイクル情報を特定する列と、を含むイベントログである。アクティビティライフサイクル情報は、イベントに関連するアクティビティのステータスを表す。ステータスは、例えば、開始、完了、中断、または再開であってもよい。このような非標準フォーマットのイベントログは、ライフサイクル情報を含むようにアクティビティのラベル（すなわち、名称）を変更し、アクティビティライフサイクル情報を特定する列を削除することで、標準フォーマットに変換される。例えば、「請求書受信」というアクティビティラベルを識別する列と「開始」というアクティビティライフサイクル情報を特定する列とを有する非標準イベントログは、「請求書受信 + 開始」というアクティビティラベルを識別する列を有するように変換され、アクティビティライフサイクル情報を特定する列を削除することができる。したがって、アクティビティライフサイクル情

40

50

報に基づいてセグメントのタイプを区別することができる。例えば、アクティビティの開始実行に対応するイベントからアクティビティの中断実行に対応するイベントまでのセグメントは処理時間を表してもよく、アクティビティの中断実行に対応するイベントからアクティビティの再開実行に対応するイベントまでは待ち時間を表してもよく、アクティビティの再開実行に対応するイベントからアクティビティの完全な実行に対応するイベントまでは処理時間を表してもよい。したがって、検出されたボトルネックは、タイプ（すなわち、処理時間に起因するか、待ち時間に起因するか）によって区別され得る。

#### 【0060】

図12は、本発明の一実施形態による、図1～図2を含む、本明細書に記載の方法、ワークフロー、およびプロセスを実行するように構成されたコンピューティングシステム1200を示すブロック図である。いくつかの実施形態では、コンピューティングシステム1200は、本明細書に図示および/または記載されたコンピューティングシステムのうちの1つまたは複数であってもよい。コンピューティングシステム1200は、情報を通信するためのバス1202または他の通信機構と、情報を処理するためにバス1202に結合されたプロセッサ1204と、を含む。プロセッサ1204は、中央処理装置（CPU）、特定用途集積回路（ASIC）、フィールドプログラマブルゲートアレイ（FPGA）、グラフィックスプロセッシングユニット（GPU）、それらの複数のインスタンス、および/またはそれらの任意の組み合わせを含む、任意のタイプの汎用または専用プロセッサであってもよい。プロセッサ1204はまた、複数の処理コアを有してもよく、コアの少なくともいくつかは、特定の機能を実行するように構成されてもよい。いくつかの実施形態では、複数並列処理が使用されてもよい。

#### 【0061】

コンピューティングシステム1200は、プロセッサ1204によって実行される情報および命令を格納するためのメモリ1206をさらに含む。

メモリ1206は、ランダムアクセスメモリ（RAM）、読み出し専用メモリ（ROM）、フラッシュメモリ、キャッシュ、磁気もしくは光ディスクなどの静的ストレージ、または任意の他のタイプの非一時的コンピュータ可読媒体、またはそれらの組み合わせの任意の組み合わせで構成することができる。非一時的コンピュータ可読媒体は、プロセッサ1204によってアクセス可能な任意の利用可能な媒体であってもよく、揮発性媒体、不揮発性媒体、またはその両方を含んでもよい。媒体はまた、取り外し可能、取り外し不能、またはその両方であってもよい。

#### 【0062】

さらに、コンピューティングシステム1200は、任意の現在存在する、または将来実施される通信規格および/またはプロトコルに従って、無線および/または有線接続を介して通信ネットワークへのアクセスを提供するためのトランシーバなどの通信デバイス1208を含む。

#### 【0063】

プロセッサ1204は、バス1202を介して、ユーザに情報を表示するのに適したディスプレイ1210にさらに結合される。ディスプレイ1210はまた、タッチディスプレイおよび/または任意の適切な触覚I/Oデバイスとして構成されてもよい。

#### 【0064】

キーボード1212およびコンピュータマウス、タッチパッドなどのカーソル制御デバイス1214は、ユーザがコンピューティングシステムとインターフェースすることを可能にするためにバス1202にさらに結合される。しかしながら、特定の実施形態では、物理的なキーボードおよびマウスが存在しなくてもよく、ユーザは、ディスプレイ1210および/またはタッチパッド（図示せず）のみを介してデバイスと対話することができる。入力デバイスの任意のタイプおよび組み合わせを、設計上の選択事項として使用することができる。特定の実施形態では、物理的な入力デバイスおよび/またはディスプレイは存在しない。例えば、ユーザは、それと通信する別のコンピューティングシステムを介してコンピューティングシステム1200と遠隔で対話することができ、またはコンピュー

ティングシステム 1 2 0 0 は自律的に動作することができる。

【 0 0 6 5 】

メモリ 1 2 0 6 は、プロセッサ 1 2 0 4 によって実行されると機能を提供するソフトウェアモジュールを格納する。

モジュールは、コンピューティングシステム 1 2 0 0 のためのオペレーティングシステム 1 2 1 6 と、本明細書に記載のプロセスまたはその派生物の全部または一部を実行するように構成された 1 つまたは複数の追加の機能モジュール 1 2 1 8 と、を含む。

【 0 0 6 6 】

当業者は、「システム」が、本発明の範囲から逸脱することなく、サーバ、組み込みコンピューティングシステム、パーソナルコンピュータ、コンソール、パーソナルデジタルアシスタント ( P D A )、携帯電話、タブレットコンピューティングデバイス、量子コンピューティングシステム、もしくは任意の他の適切なコンピューティングデバイス、またはデバイスの組み合わせとして具現化され得ることを理解するであろう。上記の機能を「システム」によって実行されるものとして提示することは、本発明の範囲を決して限定することを意図するものではなく、本発明の多くの実施形態の一例を提供することを意図している。実際、本明細書に開示する方法、システム、および装置は、クラウドコンピューティングシステムを含むコンピューティング技術と一致する局所化された形態および分散された形態で実装されてもよい。

【 0 0 6 7 】

本明細書に記載されたシステム特徴のいくつかは、それらの実装の独立性をより具体的に強調するために、モジュールとして提示されていることに留意されたい。例えば、モジュールは、カスタムの超大規模集積 ( V L S I ) 回路またはゲートアレイ、ロジックチップ、トランジスタ、または他のディスクリットコンポーネントなどの既製の半導体を含むハードウェア回路として実装されてもよい。モジュールはまた、フィールドプログラマブルゲートアレイ、プログラマブルアレイロジック、プログラマブルロジックデバイス、グラフィックスプロセッシングユニットなどのプログラマブルハードウェアデバイスに実装されてもよい。モジュールはまた、様々なタイプのプロセッサによって実行するためのソフトウェアに少なくとも部分的に実装されてもよい。遂行可能コードの識別されたユニットは、例えば、オブジェクト、プロシージャ、または関数として編成することができるコンピュータ命令の 1 つまたは複数の物理ブロックまたは論理ブロックを含むことができる。それにもかかわらず、識別されたモジュールの遂行可能ファイルは、物理的に共に配置される必要はないが、論理的に共に結合されたときにモジュールを含み、モジュールの記載された目的を達成する異なる場所に格納された異なる命令を含むことができる。さらに、モジュールは、コンピュータ可読媒体に格納されてもよく、それは、例えば、ハードディスクドライブ、フラッシュデバイス、R A M、テープ、および/または本発明の範囲から逸脱することなくデータを格納するために使用される任意の他のそのような非一時的コンピュータ可読媒体であってもよい。実際、遂行可能コードのモジュールは、単一の命令、または多くの命令であってもよく、いくつかの異なるコードセグメント、異なるプログラム、およびいくつかのメモリデバイスに分散されてもよい。同様に、動作データは、本明細書ではモジュール内で識別および図示されてもよく、任意の適切な形態で具現化され、任意の適切なタイプのデータ構造内に編成されてもよい。動作データは、単一のデータセットとして収集されてもよく、または異なるストレージデバイスを含む異なる場所に分散されてもよく、少なくとも部分的に、システムまたはネットワーク上の電子信号としてのみ存在してもよい。

【 0 0 6 8 】

上記は、本開示の原理を例示しているにすぎない。したがって、当業者は、本明細書に明示的に記載または図示されていないが、本開示の原理を具現化し、その趣旨および範囲内に含まれる様々な構成を考案することができることが理解されよう。さらに、本明細書に列挙されたすべての例および条件付き言語は、主に、読者が本開示の原理および本技術を促進するために本発明者によって寄与された概念を理解するのに助けるための教育目的

10

20

30

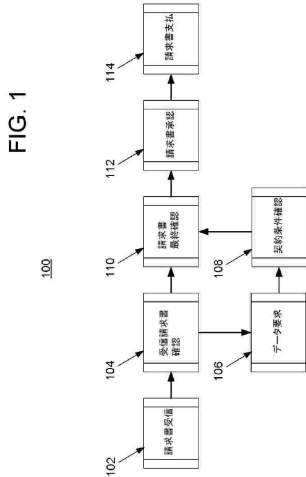
40

50

のためのものにすぎず、そのような具体的に列挙された例および条件に限定されないと解釈されるべきである。さらに、本開示の原理、態様、および実施形態、ならびにその特定の例を列挙する本明細書のすべての記述は、その構造的および機能的均等物の両方を包含することを意図している。さらに、そのような均等物は、現在知られている均等物および将来開発される均等物の両方を含むことが意図される。

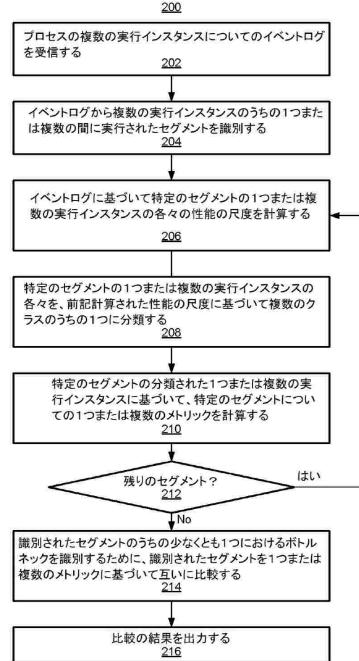
【図面】

【図 1】



【図 2】

FIG. 2



10

20

30

40

50

500

300

304-A

304-B

304-C

302

304

FIG. 3

事例ID	アクティビティ	タイムスタンプ
1	請求書受信	01/01/2020 10:00:00
1	受信請求書確認	01/01/2020 10:00:02
1	請求書最終確認	03/01/2020 13:00:12
1	請求書承認	05/01/2020 9:35:15
1	請求書支払	06/01/2020 16:25:54
2	請求書受信	04/01/2020 14:12:35
2	受信請求書確認	04/01/2020 14:12:42
2	契約条件確認	07/01/2020 12:09:16
2	請求書最終確認	09/01/2020 12:10:25
2	請求書承認	09/25/2020 10:15:15
2	請求書支払	10/15/2020 17:25:55

400

402

404

408

410

412

414

416

FIG. 4

セグメント	効果量	損失時間	重み付き影響	効果量のランク付け	損失時間のランク付け	重み付き影響のランク付け	累積ランク付け
<A, B>	1.14	10000	85%	7	7	6	20
<B, C>	0.59	100	85%	2	2	7	11
<C, D>	1.21	8000	13%	6	4	2	9
<A, D>	0.95	2000	4%	5	6	1	12
<C, E>	0.31	300	20%	3	5	3	8
<D, E>	0.73	250	69%	4	1	1	7
<B, E>	0.85	4000	69%	4	5	5	14

500

504

502

FIG. 5

セグメントの概要	セグメント情報	事例属性	イベント属性	プロセスフローの比較
セグメント	事例属性	クラス	効果量のランク付け	損失サイズのランク付け
確認および承認-請求書支払	2,486	4	9	8
受信請求書確認-請求書最終確認	13,079	5	12	6
契約条件確認-請求書最終確認	5,666	4	10	11
請求書最終確認-請求書承認	3,194	3	6	12
請求書最終確認-請求書承認	16,707	3	3	10
受信請求書確認-請求書承認	5,666	4	8	7
請求書承認-請求書支払	840	3	4	5
請求書最終確認-請求書支払	2,772	3	7	4
受信請求書確認-確認および承認	2,486	6	11	3
請求書承認-請求書支払	16,698	3	2	8
請求書支払-請求書最終確認	744	4	5	1
請求書最終確認-契約条件確認	742	4	1	2

FIG. 5

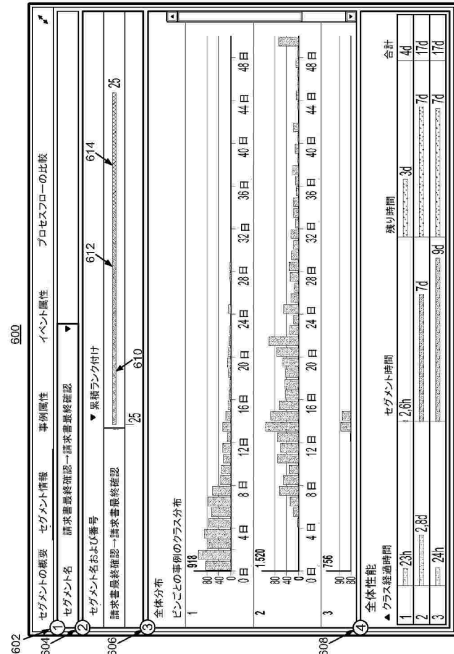


FIG. 6

【 図 7 】

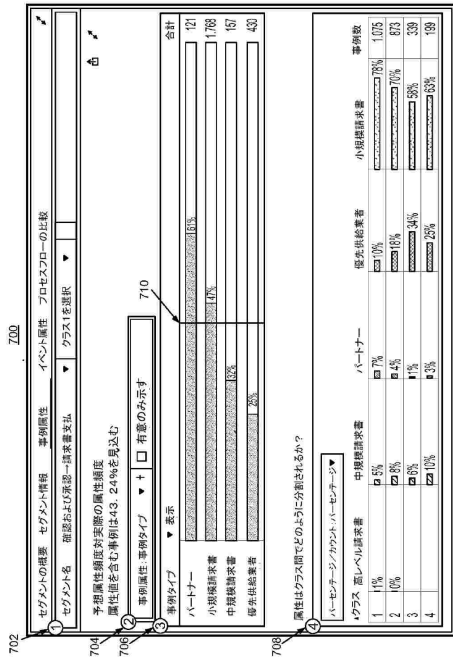


FIG. 7

【 図 8 】

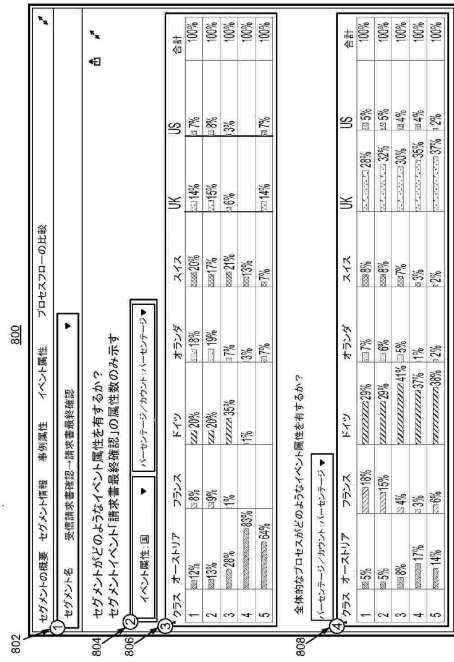


FIG. 8

【 図 9 】

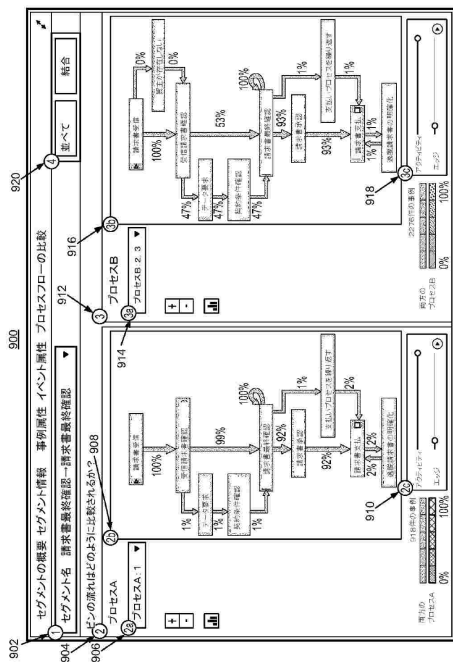


FIG. 9

【 図 10 】

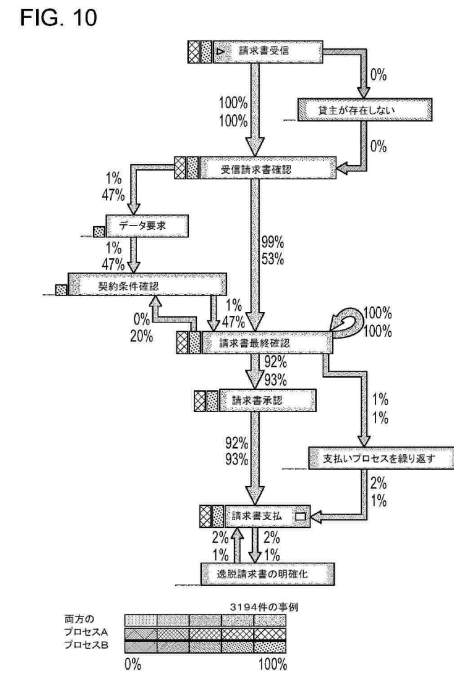


FIG. 10

【 図 1 1 】

【 図 1 2 】

FIG. 11

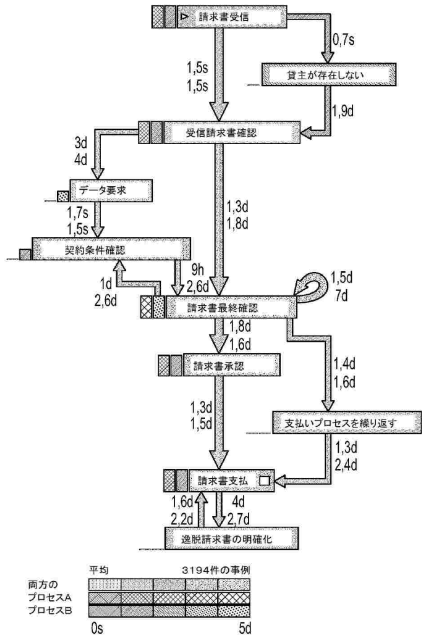
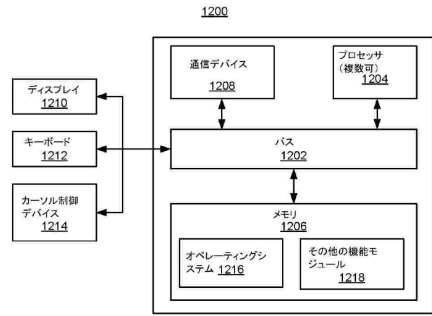


FIG. 12



10

20

30

40

50

## フロントページの続き

- アメリカ合衆国 10016 ニューヨーク, ニューヨーク, パーク アベニュー 90, トウェン  
テイス フロア, ユーアイパス, インコーポレイテッド内
- (72)発明者 スチープンス, ローランド ヨハナス  
アメリカ合衆国 10016 ニューヨーク, ニューヨーク, パーク アベニュー 90, トウェン  
テイス フロア, ユーアイパス, インコーポレイテッド内
- 審査官 山崎 雄司
- (56)参考文献 米国特許出願公開第2003/0236585 (US, A1)  
米国特許出願公開第2009/0177642 (US, A1)  
中国特許出願公開第103218692 (CN, A)
- (58)調査した分野 (Int.Cl., DB名)  
G06Q 10/00 - 99/00