

(19) 日本国特許庁 (JP)

(12) 特 許 公 報 (B2)

(11) 特許番号

特許第5657183号
(P5657183)

(45) 発行日 平成27年1月21日 (2015. 1. 21)

(24) 登録日 平成26年12月5日 (2014. 12. 5)

(51) Int. Cl.

F I

G O 6 F 15/00 (2006. 01)

G O 6 F 15/00 4 7 0

G O 6 F 9/44 (2006. 01)

G O 6 F 9/06 6 2 0 A

G O 6 F 3/048 (2013. 01)

G O 6 F 3/048 6 5 1 E

請求項の数 8 (全 16 頁)

(21) 出願番号 特願2014-532519 (P2014-532519)
 (86) (22) 出願日 平成24年9月19日 (2012. 9. 19)
 (65) 公表番号 特表2014-532223 (P2014-532223A)
 (43) 公表日 平成26年12月4日 (2014. 12. 4)
 (86) 国際出願番号 PCT/IB2012/054962
 (87) 国際公開番号 W02013/057607
 (87) 国際公開日 平成25年4月25日 (2013. 4. 25)
 審査請求日 平成26年3月28日 (2014. 3. 28)
 (31) 優先権主張番号 11185994.8
 (32) 優先日 平成23年10月20日 (2011. 10. 20)
 (33) 優先権主張国 欧州特許庁 (EP)

早期審査対象出願

(73) 特許権者 390009531
 インターナショナル・ビジネス・マシーンズ・コーポレーション
 INTERNATIONAL BUSINESS MACHINES CORPORATION
 アメリカ合衆国10504 ニューヨーク州 アーモンク ニュー オーチャードロード
 (74) 代理人 100108501
 弁理士 上野 剛史
 (74) 代理人 100112690
 弁理士 太佐 種一

最終頁に続く

(54) 【発明の名称】 第1のコンピュータ・プログラムが第2のコンピュータ・プログラムのアプリケーション・ロジックを実行することを可能にするための方法および装置、第1のコンピュータ・プログラムと第

(57) 【特許請求の範囲】

【請求項1】

第1のコンピュータ・プログラムと、アプリケーション・ロジックが表示ロジックを通じて操作可能であるように前記アプリケーション・ロジックと密接に関連して前記表示ロジックを含む第2のコンピュータ・プログラムとのインターフェースをとるためのコンピュータ・プログラム・コードを生成するための方法であって、

A) 前記第2のコンピュータ・プログラムの選択されたアプリケーション・ロジック・フローに関連する第1の表示ウィンドウを特定するステップと、

B) 前記第1のコンピュータ・プログラムに与えるための前記第1の表示ウィンドウからの1つまたは複数の表示ウィンドウの出力を割り振るステップと、

C) 前記第2のコンピュータ・プログラムに与えるために前記第1の表示ウィンドウに入力するための1つまたは複数のデータ入力を割り振るステップと、

D) 前記第1の表示ウィンドウに関連する前記選択されたアプリケーション・ロジック・フローを開始するために前記第1の表示ウィンドウに入力するための1つまたは複数の実行コマンドを割り振るステップと、

E) 前記第1の表示ウィンドウ、ならびに前記関連する表示ウィンドウの出力、前記データ入力、および前記実行コマンドによって、前記選択されたアプリケーション・ロジック・フローに関して前記第1のコンピュータ・プログラムと前記第2のコンピュータ・プログラムとの間のインターフェースを提供するように構成され、

第1のコンピュータ・プログラムが、アプリケーション・ロジックが表示ロジックを通

10

20

じて操作可能であるように前記アプリケーション・ロジックと密接に関連して前記表示ロジックを含む第2のコンピュータ・プログラムのアプリケーション・ロジックを実行することを可能にするための方法であって、

前記第2のコンピュータ・プログラムの選択されたアプリケーション・ロジック・フローに関連する所定の第1の表示ウィンドウを特定するステップと、

前記第2のコンピュータ・プログラムによる前記第1の表示ウィンドウの提供に応答して、

a) 前記第1のコンピュータ・プログラムに提供するために前記第1の表示ウィンドウの任意の所定の出力からデータを読み出すステップと、

b) 前記第2のコンピュータ・プログラムに入力するために前記第1の表示ウィンドウへの任意の所定の入力にデータを書き込むステップと、

c) 前記第1の表示ウィンドウに関連する前記選択されたアプリケーション・ロジック・フローの処理を開始するために前記第2のコンピュータ・プログラムに任意の所定の実行コマンドを与えるステップと

を含む、方法を実行するように構成されるプログラム・インターフェース・コードを生成するステップと

を含む、方法。

【請求項2】

前記選択されたアプリケーション・ロジック・フローの前記処理に応答して前記第1の表示ウィンドウの直後に提供され得る前記第2のコンピュータ・プログラムからの所定の1組の1つまたは複数の第2の表示ウィンドウを特定するステップと、

それぞれの第2の表示ウィンドウに関して前記ステップB)、C)、およびD)を実行するステップと、

前記第1の表示ウィンドウおよび前記第2の表示ウィンドウ、ならびに前記関連する表示ウィンドウの出力、前記データ入力、および前記実行コマンドによって、前記選択されたアプリケーション・ロジック・フローに関して前記第1のコンピュータ・プログラムと前記第2のコンピュータ・プログラムとの間のインターフェースを提供するように構成され、

第1のコンピュータ・プログラムが、アプリケーション・ロジックが表示ロジックを通じて操作可能であるように前記アプリケーション・ロジックと密接に関連して前記表示ロジックを含む第2のコンピュータ・プログラムのアプリケーション・ロジックを実行することを可能にするための方法であって、

前記第2のコンピュータ・プログラムの選択されたアプリケーション・ロジック・フローに関連する所定の第1の表示ウィンドウを特定するステップと、

前記第2のコンピュータ・プログラムによる前記第1の表示ウィンドウの提供に応答して、

a) 前記第1のコンピュータ・プログラムに提供するために前記第1の表示ウィンドウの任意の所定の出力からデータを読み出すステップと、

b) 前記第2のコンピュータ・プログラムに入力するために前記第1の表示ウィンドウへの任意の所定の入力にデータを書き込むステップと、

c) 前記第1の表示ウィンドウに関連する前記選択されたアプリケーション・ロジック・フローの処理を開始するために前記第2のコンピュータ・プログラムに任意の所定の実行コマンドを与えるステップと

を含む、方法を実行するように構成される前記プログラム・インターフェース・コードを生成するステップと

をさらに含む、請求項1に記載の方法。

【請求項3】

前記プログラム・コードが、前記第2のコンピュータ・プログラムの前記選択されたアプリケーション・ロジック・フローの所定のフロー定義から少なくとも部分的に自動で生成される、請求項1に記載の方法。

10

20

30

40

50

【請求項 4】

前記フロー定義が、ユーザによって少なくとも部分的にグラフィカルに決定される、請求項 3 に記載の方法。

【請求項 5】

第 1 のコンピュータ・プログラムと、アプリケーション・ロジックが表示ロジックを通じて操作可能であるように前記アプリケーション・ロジックと密接に関連して前記表示ロジックを含む第 2 のコンピュータ・プログラムとのインターフェースをとるためのコンピュータ・プログラム・コードを生成するための装置であって、

A) 前記第 2 のコンピュータ・プログラムの選択されたアプリケーション・ロジック・フローに関連する第 1 の表示ウィンドウを特定し、

B) 前記第 1 のコンピュータ・プログラムに与えるための前記第 1 の表示ウィンドウからの 1 つまたは複数の表示ウィンドウの出力を割り振り、

C) 前記第 2 のコンピュータ・プログラムに与えるために前記第 1 の表示ウィンドウに入力するための 1 つまたは複数のデータ入力を割り振り、

D) 前記第 1 の表示ウィンドウに関連する前記選択されたアプリケーション・ロジック・フローを開始するために前記第 1 の表示ウィンドウに入力するための 1 つまたは複数の実行コマンドを割り振り、

E) 前記第 1 の表示ウィンドウ、ならびに前記関連する表示ウィンドウの出力、前記データ入力、および前記実行コマンドによって、前記選択されたアプリケーション・ロジック・フローに関して前記第 1 のコンピュータ・プログラムと前記第 2 のコンピュータ・プログラムとの間のインターフェースを提供するように構成され、

第 1 のコンピュータ・プログラムが、アプリケーション・ロジックが表示ロジックを通じて操作可能であるように前記アプリケーション・ロジックと密接に関連して前記表示ロジックを含む第 2 のコンピュータ・プログラムのアプリケーション・ロジックを実行することを可能にするための装置であって、

前記第 2 のコンピュータ・プログラムの選択されたアプリケーション・ロジック・フローに関連する所定の第 1 の表示ウィンドウを特定し、

前記第 2 のコンピュータ・プログラムによる前記第 1 の表示ウィンドウの提供に応答して、

a) 前記第 1 のコンピュータ・プログラムに提供するために前記第 1 の表示ウィンドウの任意の所定の出力からデータを読み出し、

b) 前記第 2 のコンピュータ・プログラムに入力するために前記第 1 の表示ウィンドウへの任意の所定の入力にデータを書き込み、

c) 前記第 1 の表示ウィンドウに関連する前記選択されたアプリケーション・ロジック・フローの処理を開始するために前記第 2 のコンピュータ・プログラムに任意の所定の実行コマンドを与える

ように動作可能である、装置をもたらしように構成されるプログラム・インターフェース・コードを生成する

ように動作可能である、装置。

【請求項 6】

前記選択されたアプリケーション・ロジック・フローの前記処理に応答して前記第 1 の表示ウィンドウの直後に提供され得る前記第 2 のコンピュータ・プログラムからの所定の 1 組の 1 つまたは複数の第 2 の表示ウィンドウを特定し、

それぞれの第 2 の表示ウィンドウに関して動作 B)、C)、および D) を実行し、

前記第 1 の表示ウィンドウおよび前記第 2 の表示ウィンドウ、ならびに前記関連する表示ウィンドウの出力、前記データ入力、および前記実行コマンドによって、前記選択されたアプリケーション・ロジック・フローに関して前記第 1 のコンピュータ・プログラムと前記第 2 のコンピュータ・プログラムとの間のインターフェースを提供するように構成され、

第 1 のコンピュータ・プログラムが、アプリケーション・ロジックが表示ロジックを通

10

20

30

40

50

じて操作可能であるように前記アプリケーション・ロジックと密接に関連して前記表示ロジックを含む第2のコンピュータ・プログラムのアプリケーション・ロジックを実行することを可能にするための装置であって、

前記第2のコンピュータ・プログラムの選択されたアプリケーション・ロジック・フローに関連する所定の第1の表示ウィンドウを特定し、

前記第2のコンピュータ・プログラムによる前記第1の表示ウィンドウの提供に応答して、

a) 前記第1のコンピュータ・プログラムに提供するために前記第1の表示ウィンドウの任意の所定の出力からデータを読み出し、

b) 前記第2のコンピュータ・プログラムに入力するために前記第1の表示ウィンドウへの任意の所定の入力にデータを書き込み、

c) 前記第1の表示ウィンドウに関連する前記選択されたアプリケーション・ロジック・フローの処理を開始するために前記第2のコンピュータ・プログラムに任意の所定の実行コマンドを与える

ように動作可能である、装置をもたらしように構成される前記プログラム・インターフェース・コードを生成する

ようにさらに動作可能である、請求項5に記載の装置。

【請求項7】

前記プログラム・コードが、前記第2のコンピュータ・プログラムの前記選択されたアプリケーション・ロジック・フローの所定のフロー定義から少なくとも部分的に自動で生成される、請求項5に記載の装置。

【請求項8】

前記フロー定義が、ユーザによって少なくとも部分的にグラフィカルに決定される、請求項7に記載の装置。

【発明の詳細な説明】

【技術分野】

【0001】

本発明は、第1のコンピュータ・プログラムが、アプリケーション・ロジックが表示ロジックを通じて操作可能であるように前記アプリケーション・ロジックと密接に関連して前記表示ロジックを含む第2のコンピュータ・プログラムのアプリケーション・ロジックを実行することを可能にするように構成されたソフトウェア・インターフェースを提供するための方法、装置、またはコンピュータ・プログラム製品に関する。

【背景技術】

【0002】

一部のコンピュータ・プログラムは、その他のソフトウェア・システムに統合されるか、またはその他のソフトウェア・システムとインターフェースをとることが難しいような方法で設計される。これは、レガシー・プログラムと呼ばれることが多い比較的古いコンピュータ・プログラムで特に問題となり得る。プログラムは、プログラムの処理またはアプリケーション・ロジックと表現または表示ロジックとの間の密接な結びつきが原因で、統合されることが難しい可能性がある。これは、アプリケーションおよび表示ロジックが単一のプログラム・コンポーネントによって提供されることに起因する可能性がある。言い換えると、プログラムは、データまたはコマンド入力およびデータ出力をとまなう表示ウィンドウの形態のユーザ・インターフェース要素を、適切な出力を与えるように入力によって駆動されるデータ処理ロジックから分離することが難しいような方法で設計される可能性がある。

【0003】

そのようなプログラムとインターフェースをとる1つの方法は、スクリーン・スクレイピングなどの知られている方法を使用してプログラムの元のもしくは既存のユーザ・インターフェースによるか、またはプログラムのすべての入力および出力を変換して、そのプログラムをその他のソフトウェア・システムに統合することを可能にする変換システムに

10

20

30

40

50

よる。そのようなプログラムとインターフェースをとる別の方法は、プログラム・コードを、そのプログラム・コードがその他のソフトウェア・システムに統合するために組み直され得るように好適なプログラミング言語に変換することである。そのようなインターフェース技術は、時間がかかり、誤りが起こりやすく、したがって高価であるという点で問題をはらんでいる。さらに、レガシー・プログラムのユーザは、アプリケーションに関する元のコードをもはや保有していない可能性がある。通例、レガシー・プログラムは、ユーザの組織または会社の中核となっており、したがって、ユーザは、そのような重要なシステムへのいかなる変更のリスクも冒したくない可能性がある。

【発明の概要】

【発明が解決しようとする課題】

10

【0004】

本発明の目的は、従来技術における課題を解決することにある。

【課題を解決するための手段】

【0005】

本発明の第1の実施形態は、第1のコンピュータ・プログラムが、アプリケーション・ロジックが表示ロジックを通じて操作可能であるようにアプリケーション・ロジックと密接に関連して表示ロジックを含む第2のコンピュータ・プログラムのアプリケーション・ロジックを実行することを可能にするための方法であって、第2のコンピュータ・プログラムの選択されたアプリケーション・ロジック・フローに関連する所定の第1の表示ウィンドウを特定するステップと、第2のコンピュータ・プログラムによる第1の表示ウィンドウの提供に
20 応答して、a) 第1のコンピュータ・プログラムに提供するために第1の表示ウィンドウの任意の所定の出力からデータを読み出すステップと、b) 第2のコンピュータ・プログラムに入力するために第1の表示ウィンドウへの任意の所定の入力にデータを書き込むステップと、c) 第1の表示ウィンドウに関連する選択されたアプリケーション・ロジック・フローの処理を開始するために第2のコンピュータ・プログラムに任意の所定の実行コマンドを与えるステップとを含む、方法を提供する。

【0006】

方法は、選択されたアプリケーション・ロジック・フローの処理に
30 応答して第1の表示ウィンドウの直後に提供され得る第2のコンピュータ・プログラムからの所定の1組の1つまたは複数の第2の表示ウィンドウを特定するステップと、第2のコンピュータ・プログラムによる第2の表示ウィンドウのうちの1つの提供に
40 応答して、第2の表示ウィンドウに関してステップa) からc) までを実行するステップとをさらに含み得る。

【0007】

方法は、第1のコンピュータ・プログラムが第2のコンピュータ・プログラムの複数の選択されたアプリケーション・ロジック・フローを実行することを可能にするように構成され得る。ステップb) において、データは、第1のコンピュータ・プログラムから入力され得る。実行コマンドは、シミュレーションされたユーザ・コマンドである可能性がある。選択されたアプリケーション・ロジック・フローは、表示ウィンドウの所定のシーケンスによって特定される可能性があり、シーケンス内の表示ウィンドウ間の遷移は、所与の遷移に関連する1つまたは複数の所定のアプリケーション・ロジック要素の処理の結果として起こる。

【0008】

第2の実施形態は、第1のコンピュータ・プログラムと、アプリケーション・ロジックが表示ロジックを通じて操作可能であるようにアプリケーション・ロジックと密接に関連して表示ロジックを含む第2のコンピュータ・プログラムとのインターフェースをとるためのコンピュータ・プログラム・コードを生成するための方法であって、a) 第2のアプリケーション・プログラムの選択されたアプリケーション・ロジック・フローに関連する第1の表示ウィンドウを特定するステップと、b) 第1のコンピュータ・プログラムに与えるための第1の表示ウィンドウからの1つまたは複数の表示ウィンドウの出力を割り振るステップと、c) 第2のコンピュータ・プログラムに与えるために第1の表示ウィンド
50

ウに入力するための1つまたは複数のデータ入力を割り振るステップと、d)第1の表示ウィンドウに関連する選択されたアプリケーション・ロジック・フローを開始するために第1の表示ウィンドウに入力するための1つまたは複数の実行コマンドを割り振るステップと、e)第1の表示ウィンドウ、ならびに関連する表示ウィンドウの出力、データ入力、および実行コマンドによって、選択されたアプリケーション・ロジック・フローに関して第1のコンピュータ・プログラムと第2のコンピュータ・プログラムとの間のインターフェースを提供するように構成され、本発明の第1の実施形態による方法を実行するように構成されるプログラム・インターフェース・コードを生成するステップとを含む、方法を提供する。

【0009】

方法は、選択されたアプリケーション・ロジック・フローの処理に応答して第1の表示ウィンドウの直後に提供され得る第2のコンピュータ・プログラムからの所定の1組の1つまたは複数の第2の表示ウィンドウを特定するステップと、それぞれの第2の表示ウィンドウに関してステップb)、c)、およびd)を実行するステップと、第1の表示ウィンドウおよび第2の表示ウィンドウ、ならびに関連する表示ウィンドウの出力、データ入力、および実行コマンドによって、選択されたアプリケーション・ロジック・フローに関して第1のコンピュータ・プログラムと第2のコンピュータ・プログラムとの間のインターフェースを提供するように構成され、本発明の第1の実施形態による方法を実行するように構成されるプログラム・インターフェース・コードを生成するステップとをさらに含み得る。

【0010】

プログラム・コードは、第2のアプリケーション・プログラムの選択されたアプリケーション・ロジック・フローの所定のフロー定義から少なくとも部分的に自動で生成され得る。フロー定義は、ユーザによって少なくとも部分的にグラフィカルに決定され得る。

【0011】

第3の実施形態は、第1のコンピュータ・プログラムが、アプリケーション・ロジックが表示ロジックを通じて操作可能であるようにアプリケーション・ロジックと密接に関連して表示ロジックを含む第2のコンピュータ・プログラムのアプリケーション・ロジックを実行することを可能にするための装置であって、第2のコンピュータ・プログラムの選択されたアプリケーション・ロジック・フローに関連する所定の第1の表示ウィンドウを特定し、第2のコンピュータ・プログラムによる第1の表示ウィンドウの提供に応答して、a)第1のコンピュータ・プログラムに提供するために第1の表示ウィンドウの任意の所定の出力からデータを読み出し、b)第2のコンピュータ・プログラムに入力するために第1の表示ウィンドウへの任意の所定の入力にデータを書き込み、c)第1の表示ウィンドウに関連する選択されたアプリケーション・ロジック・フローの処理を開始するために第2のコンピュータ・プログラムに任意の所定の実行コマンドを与えるように動作可能である、装置を提供する。

【0012】

第4の実施形態は、第1のコンピュータ・プログラムと、アプリケーション・ロジックが表示ロジックを通じて操作可能であるようにアプリケーション・ロジックと密接に関連して表示ロジックを含む第2のコンピュータ・プログラムとのインターフェースをとるためのコンピュータ・プログラム・コードを生成するための装置であって、a)第2のアプリケーション・プログラムの選択されたアプリケーション・ロジック・フローに関連する第1の表示ウィンドウを特定し、b)第1のコンピュータ・プログラムに与えるための第1の表示ウィンドウからの1つまたは複数の表示ウィンドウの出力を割り振り、c)第2のコンピュータ・プログラムに与えるために第1の表示ウィンドウに入力するための1つまたは複数のデータ入力を割り振り、d)第1の表示ウィンドウに関連する選択されたアプリケーション・ロジック・フローを開始するために第1の表示ウィンドウに入力するための1つまたは複数の実行コマンドを割り振り、e)第1の表示ウィンドウ、ならびに関連する表示ウィンドウの出力、データ入力、および実行コマンドによって、選択されたア

アプリケーション・ロジック・フローに関して第1のコンピュータ・プログラムと第2のコンピュータ・プログラムとの間のインターフェースを提供するように構成され、本発明の第3の実施形態による装置をもたらすように構成されるプログラム・インターフェース・コードを生成するように動作可能である、装置を提供する。

【0013】

さらなる実施形態は、コンピュータ可読媒体に記憶され、コンピュータの内部メモリにロードされ得るコンピュータ・プログラムであって、プログラムがコンピュータで実行されるときに第1の実施形態による方法を実行するように構成されたソフトウェア・コード部分を含む、コンピュータ・プログラムを提供する。

【0014】

別の実施形態は、第1のプログラムが、アプリケーション・ロジックが表示ロジックを通じて操作可能であるように1つまたは複数のユーザ・インターフェース要素を含むアプリケーション・ロジックと密接に関連して表示ロジックを有する第2のプログラムのアプリケーション・ロジックを実行することを可能にするためのソフトウェア・インターフェースを提供するための方法であって、インターフェースが第2のプログラムの実行を引き起こすステップと、インターフェースが、第2のプログラムのアプリケーション・ロジックに対する第1のプログラムからの入力引数を受け付け、1つまたは複数のユーザ・インターフェース要素を介して第2のプログラムに入力引数を与えるステップと、インターフェースが、第2のプログラムの1つまたは複数のユーザ・インターフェース要素から抽出される第2のプログラムからの出力データを第1のプログラムに与えるステップとを含む、方法を提供する。

【0015】

ここで、本発明の実施形態が、添付の図面を参照して単なる例として説明される。

【図面の簡単な説明】

【0016】

【図1】第1のコンピュータ・プログラムがインターフェース・プログラムによって第2のコンピュータ・プログラムとインターフェースをとられるコンピュータ・システムの概略図である。

【図2】図1のインターフェース・プログラムを生成するためのコンピュータ・システムの概略図である。

【図3】図1の第2のコンピュータ・プログラムの選択された処理の間に提供される表示ウィンドウのフローの概略図である。

【図4】図3の表示ウィンドウのフローに対応する選択された処理フローの図式的表現の図である。

【図5】図4の選択された処理フローのインターフェースをとるための図1のインターフェース・プログラムの構造の概略図である。

【図6】第2のコンピュータ・プログラムにインターフェースを提供するときインターフェース・プログラムによって実行される処理を示す流れ図である。

【図7】第2のコンピュータ・プログラムの所与の表示ウィンドウに応じてインターフェース・プログラムによって実行される処理を示す流れ図である。

【図8】図4の選択された処理フローの図式的表現から図5のインターフェース・プログラムを生成するとき図2のコード生成プログラムによって実行される処理を示す流れ図である。

【発明を実施するための形態】

【0017】

図1を参照すると、コンピュータ・システム101が、第2のコンピュータ103に接続された第1のコンピュータ102を含む。第1のコンピュータは、ネットワーク104を介して第3のコンピュータ105にも接続されている。本実施形態において、第1のコンピュータ102は、ウェブ・サーバ・アプリケーション・プログラムの形態の第1のコンピュータ・プログラム106を搭載しており、第2のコンピュータ103は、第2の

ンピュータ・プログラム107を搭載している。本実施形態において、第1のコンピュータ・プログラム106は、第2のコンピュータ・プログラム107によって提供される機能を、第3のコンピュータ105のユーザが第3のコンピュータ105にインストールされたウェブ・ブラウザ・アプリケーション・プログラム108を通じてネットワーク104経由で利用できるようにするように構成される。言い換えると、第1のコンピュータ・プログラム106は、第2のコンピュータ・プログラム107のラップとして構成される。

【0018】

本実施形態において、第2のコンピュータ・プログラム107は、第1のコンピュータ・プログラム106などのその他のアプリケーション・プログラムと通信するためのインターフェースなしで設計されたレガシー・アプリケーション・プログラムである。したがって、第1のコンピュータ・プログラム106は、第2のコンピュータ・プログラム107とインターフェースをとるためのインターフェース・プログラム109を与えられる。第2のコンピュータ・プログラム107は、関連するデータ処理ロジックと一体的にコーディングされ、したがって別々に再利用できない画面または表示ウィンドウなどの第2のコンピュータ・プログラム107のデータ表現要素を用いて設計される。言い換えると、第2のコンピュータ・プログラム107の表示ロジックは、アプリケーション・ロジックに密接に関連して設けられる。インターフェース・プログラム109は、第1のコンピュータ・プログラム106が第2のコンピュータ・プログラム107の選択されたアプリケーション・ロジックを実行することを可能にするように、第2のコンピュータ・プログラム107の選択された表示画面を通じて第1のコンピュータ・プログラム106と第2のコンピュータ・プログラム107とのインターフェースをとるように構成される。

【0019】

本実施形態において、第2のコンピュータ103は、IBM（商標）AS/400コンピュータである。第2のコンピュータ・プログラム107は、IBM（商標）5250端末の形態の端末を介して使用するように元々設計されている。本実施形態において、インターフェース・プログラム109は、IBM（商標）社から入手可能なIBM（商標）Host on Demandライブラリを使用して、IBM（商標）5250端末を必要とせずにIBM（商標）AS/400コンピュータ上の第2のコンピュータ・プログラム107の表示ロジックに直接アクセスするように構成される。

【0020】

図2を参照すると、第4のコンピュータ201が、Eclipse Foundation（商標）、Inc. のEclipse（商標）SDEの形態のソフトウェア開発環境（SDE）プログラム202を搭載している。本実施形態において、SDE202は、フロー規定プログラム203およびコード生成プログラム204を提供するように構成される。フロー規定プログラム203は、ユーザが第2のコンピュータ・プログラム107の選択された処理フローを規定することを可能にするように構成される。本実施形態において、フロー規定プログラム203は、図4を参照して以下でさらに説明されるように、ユーザが選択された処理フローをグラフィカルに規定することを可能にするように構成される。そのようなユーザの規定に応じて、フロー規定プログラム203は、コード生成プログラム204への入力であるフロー定義205を出力する。コード生成プログラム204は、以下でさらに説明されるように、入力されたフロー定義205からインターフェース・プログラム109のためのコードを生成するように構成される。

【0021】

本実施形態において、第2のコンピュータ・プログラム107は、適切な端末の画面としてユーザに対して表示するために設計された1組の表示ウィンドウを含むユーザ・インターフェースを用いて構成される。表示ウィンドウは、入力フィールド、出力フィールドの形態の1つまたは複数のユーザ・インターフェース要素を含む可能性があり、所与の表示ウィンドウに関連する処理ロジックを実行するように第2のコンピュータ・プログラム107に命令するためのコマンドの形態の1つまたは複数のユーザ入力を受け付けるよう

に構成される可能性がある。本実施形態において、インターフェース・プログラム109は、選択された機能が表示ウィンドウの関連するフローによって特定されるようにして、第2のコンピュータ・プログラム107の選択された機能または処理ロジックのフローとインターフェースをとるように構成される。図3は、第2のコンピュータ・プログラム107の選択されたデータ表示機能に関連する1組の表示ウィンドウ301、302、303の例を示す。選択されたデータ表示機能は、表示することができる利用可能なデータの範囲のメニューを提供する第1の表示ウィンドウ301によって開始される。第1の表示ウィンドウ301は、ユーザが、データの範囲、すなわち、AからMまでの範囲のデータ、またはNからZまでの範囲のデータを選択するためのコマンドを入力することを可能にする。そのようなコマンドにตอบสนองして、第2のコンピュータ・プログラム107は、第1の表示ウィンドウ301に戻る前に、選択されたデータの範囲に応じて表示ウィンドウのうちの第2のウィンドウ302または第3のウィンドウ303のどちらかに選択された範囲内の利用可能なデータを表示する。

10

【0022】

図4を参照すると、フロー規定プログラム203は、ユーザが、第2のコンピュータ・プログラム107の選択された処理ロジックを、関連する画面401、402、403、404、入力405、および出力406、407を特定するフロー定義205として規定することを可能にするように構成される。画面間の遷移408、409、410が規定され、データまたはコマンドが入力されるかまたはデータが出力される選択された処理内の点を表す。遷移408、409、410のそれぞれは、第2のコンピュータ・プログラム107によって処理される選択されたアプリケーション・ロジック・フローの対応する要素を表す。画面401、402、403、404のそれぞれは、それぞれの画面の一意識別子の働きをするそれぞれの画面種別 (screen description) 411、412、413、414を与えられる。一部の画面は、図4、および同じ画面種別「A」411、414を有する画面1および3 401、404の場合のように、所与の遷移において2回以上現れる可能性があるが、異なる結果を生じる可能性がある。図3から、表示ウィンドウのシーケンスはループと見なされる可能性があるが、処理ロジックの単一のフローの観点から見れば、画面1および3 401、404は、それぞれ、処理の開始点および終了点を表す。インターフェース・プログラム109は、以下でさらに説明されるように、画面種別を、第2のコンピュータ・プログラム107の処理中の任意の所与の時間に現在位置を特定するために使用する。本実施形態においては、上述のようにフロー定義205を生成するために、Eclipse Modelling Framework (EMF) が使用される。

20

30

【0023】

図5を参照すると、本実施形態において、コード生成プログラム204は、フロー定義205からインターフェース・プログラム109のためのJava (商標) のコードを生成するように構成される。図5は、生成されたJava (商標) のコードの機能要素を示す。当業者に理解されるであろうように、インターフェース・プログラム109は、第2のコンピュータ・プログラム107の1組の1つまたは複数の選択された処理ロジック・フローにインターフェースを提供するように構成され得る。本実施形態において、それぞれのそのような選択された処理ロジック・フローは、フロー処理クラス501およびデータ処理クラス502の形態の対応するインターフェース・ルーチンによってインターフェースをとられる。フロー処理クラス501は、インターフェース・プログラム109への入り口点 (entry point) を提供し、第2のコンピュータ・プログラム107とのインターフェース動作を実行する。データ処理クラス502は、入力および出力405、406ならびにコマンド入力407のための必要なデータ構造を提供する。データ処理クラス502は、インターフェース・プログラム109と第1のコンピュータ・プログラム106との間のインターフェースを含む。フロー処理クラス501は、第2のコンピュータ・プログラム107の状態を特定するため、例えば、第2のコンピュータ・プログラム107が選択された機能を実行するためのその第2のコンピュータ・プログラム107の処理の

40

50

開始点にいるときを判定するためにフロー処理クラス501によって使用される1組の予想される画面種別503を与えられる。

【0024】

フロー処理クラス501は、フロー定義205の画面401、402、403、404のそれぞれに対して1つずつ、1組の画面種別メソッド504および1組の画面識別子メソッド505をさらに含む。画面種別メソッド504は、それぞれの画面種別411、412、413、414を保持する。画面識別子メソッド505は、対応する画面種別メソッド504から得られた画面種別411、412、413、414を比較して、第2のコンピュータ・プログラム107の現在の画面を特定するように構成される。画面種別メソッド504および1組の画面識別子メソッド505は、予想される画面種別503などの1組の画面種別を入力し、もしあれば、特定された画面のうちのどれが第2のコンピュータ・プログラム107の現在の画面であるかを特定するように構成される画面チェッカー・メソッド506によって使用される。

10

【0025】

フロー処理クラスは、フロー定義205の遷移408、409、410のそれぞれに対して1つずつ、1組の遷移メソッド507をさらに含む。各遷移メソッド507は、フロー定義205の所与の遷移に割り振られた入力、出力、またはコマンド508のいずれかのインターフェースをとるように構成される。したがって、遷移メソッド507は、データを、第2のコンピュータ・プログラム107の表示ウィンドウから、第1のコンピュータ・プログラム106に出力するためにデータ処理クラス502によって提供される対応するデータ構造に移すように構成され得る。同様に、遷移メソッド507は、データを、データ処理クラス502によって提供される関連するデータ構造から、第2のコンピュータ・プログラム107の表示ウィンドウの指定された入力フィールドに移すように構成され得る。遷移メソッド507は、所与の遷移に関連するレガシー・プログラムの処理を開始するために、シミュレーションされたキーの押下などのコマンドを第2のコンピュータ・プログラム107に入力するようにさらに構成され得る。

20

【0026】

さらに、各遷移メソッド507は、フロー定義205による所与の遷移に続くあり得る遷移先画面を特定する予想される画面の組509を与えられ、それぞれのそのようなあり得る遷移先画面に関連する適切な遷移メソッド507をやはり特定する。したがって、各遷移メソッド507は、入力、出力、またはコマンドの割り振られた処理が完了すると、予想される画面の組509のうちのどの画面が第2のコンピュータ・プログラム107によって指示されているかを特定するようにさらに構成される。このように新しい現在の画面が特定されると、遷移メソッド507は、フロー定義205で定義されたように、新しい現在の画面に対応する次の遷移メソッド507を呼び出すように構成される。要約すると、各遷移メソッド507は、以下の機能を実行するように構成される。

30

1. 現在のレガシー・アプリケーションの表示ウィンドウからの任意の割り振られた出力からデータを読み出すこと。
2. 現在のレガシー・アプリケーションの表示ウィンドウの任意の割り振られた入力にデータを書き込むこと。
3. 現在の遷移の処理の結果として生じると予想される1組の表示ウィンドウを決定すること。
4. レガシー・アプリケーションにコマンドを与えることによって現在の遷移を実行すること。および
5. 結果として生じる画面を特定し、対応する遷移メソッドを実行すること。

40

【0027】

第1のコンピュータ・プログラム106からの、第2のコンピュータ・プログラム107によって提供される所定の選択された機能の要求に応答してインターフェース・プログラム109によって実行される処理が、以降、図6の流れ図を参照してさらに詳細に説明される。処理は、第1のコンピュータ・プログラム106からの要求に応答してステップ

50

601で開始され、ステップ602に移る。ステップ602において、第1のコンピュータ・プログラム106による選択された処理ロジック・フローの要求に対応するインターフェースをとるルーチンが特定され、処理はステップ603に移る。ステップ603において、第2のコンピュータ・プログラム107との通信が、特定されたインターフェースをとるルーチンによって開始され、処理は、ステップ604に移る。ステップ604において、インターフェースをとるルーチンに対する最初の画面が待ち受けられ、与えられると、処理はステップ605に移る。ステップ605において、関連するインターフェース・データ構造から入力およびコマンドを取得し、関連するインターフェース・データ構造に出力を与える、インターフェースをとるルーチンに対して指定された遷移のシーケンスが、処理される。遷移のシーケンスの処理が完了すると、処理はステップ606に移り、終了する。

10

【0028】

所与の遷移メソッド507のインターフェース・プログラム109による処理が、以降、図7の流れ図を参照してさらに詳細に説明される。処理は、ステップ701において、フロー処理クラス501の入り口点からか、または別の遷移メソッド507によってかのいずれかで開始され、処理は、ステップ702に移る。ステップ702において、現在の画面の内容が、適切なHOD（商標）ライブラリ・ルーチンによって第2のコンピュータ・プログラム107から取得され、画面から任意の所定の出力、すなわち、フロー定義205で遷移に割り振られた任意の出力を抽出するために解析される。抽出された出力は、次に、第1のコンピュータ・プログラム106によるアクセスのために関連するデータ処理クラス502に記憶され、処理は、ステップ703に移る。ステップ703において、第2のコンピュータ・プログラム107に対する任意のデータ入力が、適切なHOD（商標）ライブラリ・ルーチンによって、関連するデータ処理クラス502から第2のコンピュータ・プログラム107の現在の画面に移され、処理は、ステップ704に移る。ステップ704において、第2のコンピュータ・プログラム107に対する任意のコマンド入力が、適切なHOD（商標）ライブラリ・ルーチンによって第2のコンピュータ・プログラム107に与えられ、処理は、ステップ705に移る。ステップ705において、現在の画面からのすべてのあり得る遷移先画面が、予想される画面の組509から特定され、処理は、ステップ706に移る。ステップ706において、第2のコンピュータ・プログラム107の出力が、適切なHOD（商標）ライブラリ・ルーチンによって問い合わせられ、あり得る遷移先画面から新しい現在の画面が特定され、対応する遷移メソッド507が呼び出される。次に、処理は、ステップ707に進み、現在の遷移メソッド507に関して終了する。

20

30

【0029】

フロー定義205からインターフェース・プログラム109を生成するためにコード生成プログラム204によって実行される処理が、以降、図8の流れ図を参照してさらに詳細に説明される。処理は、ステップ801で開始され、ステップ802に移り、ステップ802において、EMFの関連するフロー定義205が入力される。次に、処理はステップ803に移り、ステップ803において、インターフェース・プログラム109が、入力されたフロー定義205に対応するフロー・クラス（flow class）の形態のフロー処理クラス501によって初期化され、処理は、ステップ804に移る。ステップ804において、フロー定義205で定義された各入力および出力405、406、407に対応するデータ記憶要素を含むフロー入力／出力データ処理クラス502が生成され、処理はステップ805に移る。ステップ805において、フロー定義205の画面401、402、403、404のそれぞれに対する画面種別メソッド504および画面識別子メソッドを画面チェッカー・メソッド506とともに含む画面種別クラスが生成され、処理はステップ806に移る。ステップ806において、フロー定義205の各遷移408、409、410に対する遷移メソッド507が生成され、処理はステップ807に移る。ステップ807において、各遷移メソッド507が、対応する予想される画面の組509および対応する遷移メソッドによって補完される。次に、処理はステップ808に移り、ステッ

40

50

プ 8 0 8 において、インターフェース・プログラム 1 0 9 が出力され、処理は終了する。

【0 0 3 0】

インターフェース・プログラム 1 0 9 は、第 2 のコンピュータ・プログラム 1 0 7 の選択された 1 組のフロー定義 2 0 5 のそれぞれにアクセスできるようにするために複数のフロー処理クラス 5 0 1 を与えられる可能性がある。

【0 0 3 1】

別の実施形態においては、インターフェース・プログラムは、レガシー・プログラムの誤ったまたは予期せぬ振る舞いに対処し、例えば、そのレガシー・プログラムがインターフェースをとられているアプリケーション・プログラムにエラーを報告することによってそのような振る舞いを適切に処理するためのエラー処理機能が設けられる。

10

【0 0 3 2】

さらなる実施形態においては、フロー定義が、インターフェースをとられているプログラムのより複雑な処理ロジックの記述を容易にするためのサブルーチン、条件、またはループを含む可能性がある。

【0 0 3 3】

本発明の実施形態は、インターフェースをとられるプログラムの表現を用いて動作するためのインターフェース・コードを生成するように構成される。T N 3 2 7 0 E、T N 5 2 5 0、V T 1 0 0、V T 2 2 0、V T 4 2 0 などのレガシー・システムを含むシステムのためのホスト・アクセス・メソッドをサポートする I B M J a v a H o s t - O n - D e m a n d (商標) などのホスト・アクセス・メソッド。それによって、インターフェース・コードは、レガシー・コンポーネントとインターフェースをとるためにその表現にぴったりとはまることによって、レガシーのロジックを最新のランタイム環境で提供する。

20

【0 0 3 4】

当業者に理解されるであろうように、上記の教示は、J a v a (商標) でのインターフェース・プログラムの生成に限定されず、任意の好適なソフトウェア言語で使用され得る。

【0 0 3 5】

上述のフロー定義はグラフィカルに行われるが、選択されたアプリケーション・ロジック・フローが任意のその他の好適な手段で定義され得ることは、当業者に理解されるであろう。

30

【0 0 3 6】

上述のように、上述のインターフェース・プログラムは、特定のレガシー・システムにアクセスするために既存の H o D (商標) ライブラリ・ルーチンを利用する。当業者に理解されるであろうように、任意の好適なシステムに関してそのようなライブラリ・ルーチンが提供され得る。そのようなシステム・アクセス・ルーチンの機能は、インターフェースをとるプログラムの実施形態の機能の一部として提供され得る。さらに、任意のそのようなライブラリが、サード・パーティの提供者から得られるか、または特に所与のアプリケーション用に、つまり、カスタム開発によって開発される可能性がある。

【0 0 3 7】

40

本発明の実施形態は、いわゆるレガシー・システムでの使用に限定されず、任意のコンピュータ・プログラムと、そのプログラムの選択された処理ロジックを駆動するためのそのプログラムの表示ウィンドウを通じてインターフェースをとるために提供され得る。

【0 0 3 8】

本発明の実施形態は、レガシーのソフトウェア・コンポーネントがロジックと表現との間の堅い結びつきを含む、言い換えると、表現とロジックとが 1 つのコンポーネントでリンクされている場合に、レガシー・コンポーネントをサービス指向アーキテクチャ (S O A) などの最新のソフトウェア環境に統合するためのコードを生成するためのツールを提供する。

【0 0 3 9】

50

コンピュータ・プログラムは、使用されているソフトウェア言語と、そのようなソフトウェアが実行されるように意図されているシステムとに応じて、よくある補助的なソフトウェア要素および構造を含む。そのような要素および構造は、説明を明瞭にするために上記の説明に含まれていないが、当業者には明らかであろう。

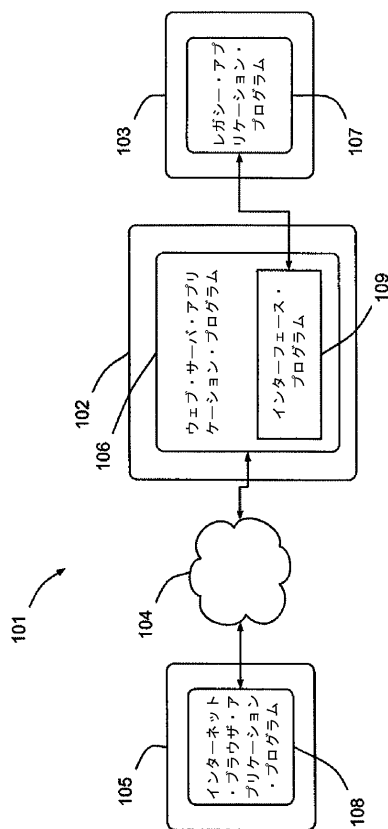
【0040】

本発明の一部またはすべてを具現化する装置が、本発明の実施形態の一部またはすべてを提供するように構成されたソフトウェアを有する汎用デバイスである可能性があることは、当業者に理解されるであろう。デバイスは、単一のデバイスまたはデバイスのグループである可能性があり、ソフトウェアは、単一のプログラムまたは1組のプログラムである可能性がある。さらに、本発明を実装するために使用されるソフトウェアのいずれかまたはすべては、1つまたは複数のデバイスにロードされ得るように任意の好適な送信または記憶手段を介して伝達される可能性がある。

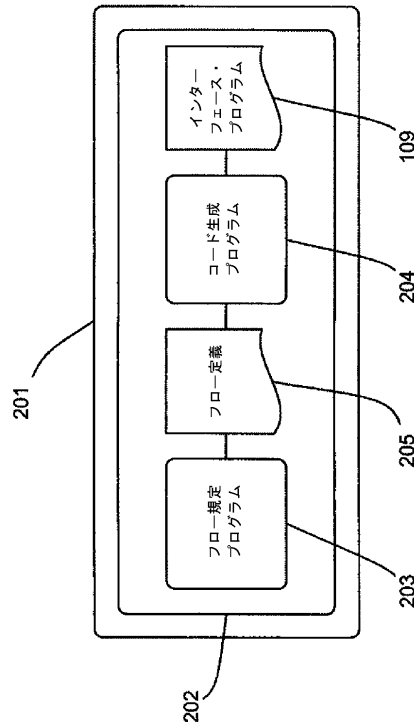
【0041】

本発明が本発明の実施形態の説明によって示され、実施形態が非常に詳細に説明されたが、出願人は、添付の特許請求の範囲をそのような詳細に制限するか、または何らかの形で限定することを意図していない。さらなる利点および改変が、当業者には容易に分かるであろう。したがって、本発明は、本発明のより広い態様において、代表的な装置および方法、ならびに図示され、説明された例示的な例の特定の詳細に限定されない。ゆえに、出願人の包括的な発明の概念の精神または範囲を逸脱することなく、そのような詳細から離れることができる。

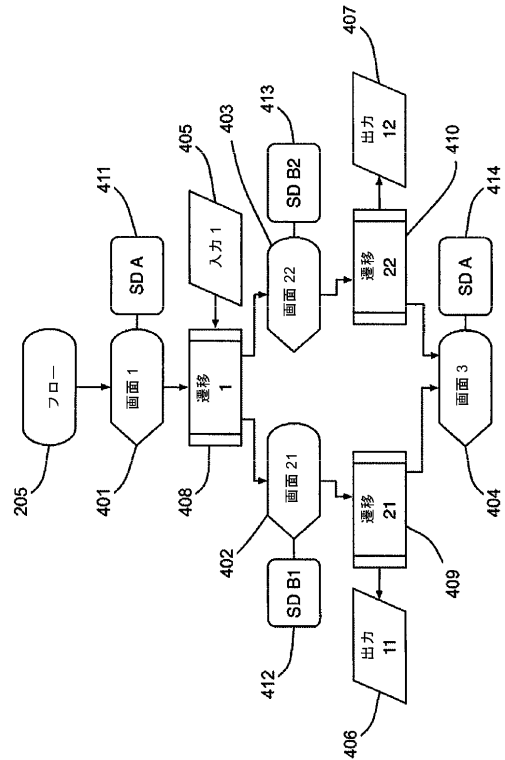
【図1】



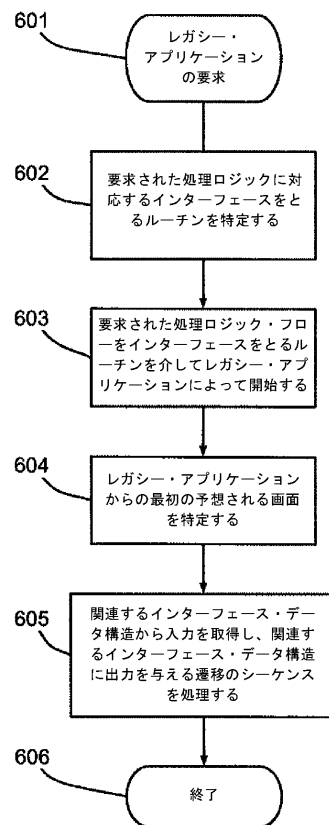
【図2】



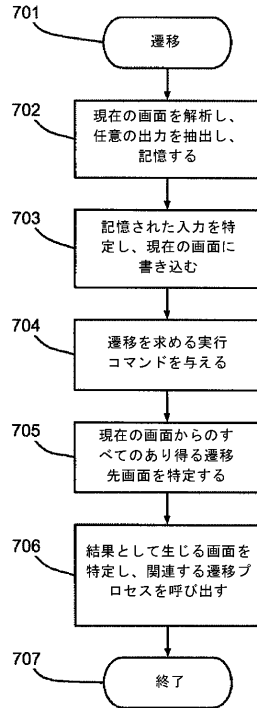
【図 4】



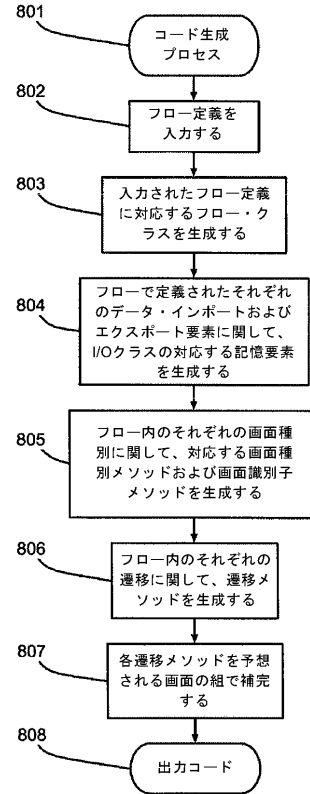
【图 6】



【図 7】



【図 8】



フロントページの続き

(72)発明者 フェルナンデス、ジェナロ、ニエト
スペイン イー-08029 バルセロナ アヴダ・ディアグナル571

審査官 漆原 孝治

(56)参考文献 特開2002-334058 (JP, A)
特開2003-114874 (JP, A)
特開2002-91780 (JP, A)

(58)調査した分野(Int.Cl., DB名)
G06F 15/00
G06F 3/048
G06F 9/44

(54)【発明の名称】第1のコンピュータ・プログラムが第2のコンピュータ・プログラムのアプリケーション・ロジックを実行することを可能にするための方法および装置、第1のコンピュータ・プログラムと第2のコンピュータ・プログラムとのインターフェースをとるためのコンピュータ・プログラム・コードを生成するための方法および装置、コンピュータ・プログラム、ならびに第1のコンピュータ・プログラムが第2のコンピュータ・プログラムのアプリケーション・ロジックを実行することを可能にするためのソフトウェア・インターフェースを提供するための方法（コンピュータ・プログラム・インターフェース）