

(19) 日本国特許庁 (JP)

(12) 公表特許公報 (A)

(11) 特許出願公表番号

特表2016-508636

(P2016-508636A)

(43) 公表日 平成28年3月22日 (2016. 3. 22)

(51) Int.Cl.	F I	テーマコード (参考)
<b>G06Q 10/00 (2012.01)</b>	G06Q 10/00 140	5 L049
<b>G06Q 10/06 (2012.01)</b>	G06Q 10/06 100	

審査請求 未請求 予備審査請求 未請求 (全 28 頁)

(21) 出願番号 特願2015-555164 (P2015-555164) (86) (22) 出願日 平成25年12月20日 (2013. 12. 20) (85) 翻訳文提出日 平成27年7月22日 (2015. 7. 22) (86) 国際出願番号 PCT/US2013/077213 (87) 国際公開番号 W02014/116384 (87) 国際公開日 平成26年7月31日 (2014. 7. 31) (31) 優先権主張番号 61/755, 892 (32) 優先日 平成25年1月23日 (2013. 1. 23) (33) 優先権主張国 米国 (US) (31) 優先権主張番号 61/833, 770 (32) 優先日 平成25年6月11日 (2013. 6. 11) (33) 優先権主張国 米国 (US)	(71) 出願人 501263810 トムソン ライセンシング Thomson Licensing フランス国, 92130 イッシー レ ムーリノー, ル ジャンヌ ダルク, 1-5 1-5, rue Jeanne d' A rc, 92130 ISSY LES MOULINEAUX, France (74) 代理人 100107766 弁理士 伊東 忠重 (74) 代理人 100070150 弁理士 伊東 忠彦 (74) 代理人 100091214 弁理士 大貫 進介 最終頁に続く
---	--

(54) 【発明の名称】 アセットにより駆動されるワークフローのモデリングにおけるセット処理

## (57) 【要約】

アセットにより駆動されるワークフローの依存管理は、各アクティビティについて入力及び／又は出力として使用されるアセットの記述に基づき、アクティビティ間の接続を確立する。そのような記述的な“コントラクト”は、所望の出力を生成するのに必要な関連するアクティビティを容易に照合するメカニズムを提供する。所望のワークフローのグラフィカルモデルを生成することによって、ユーザは、何が関係しているかについてのより良い理解を得られる。グラフィカルモデルは、現実世界の製品を設計し且つ監視するために使用され得る。アクティビティのグラフィカル表現は、ペンダー、ファシリティ、及び他の生産アクティビティをモデリングするために使用される。アクティビティモデルは、アクティビティ間でやり取りされる成果物を表すアセットを生成及び／又は消費する。アクティビティのモデルを用いて、生産パイプラインのモデルは後ろから前へ構築され得る。よって、最終結果のアクティビティモデルは最初を選択され、選択された最終結果のアクティビティによって必要とされるアセットに基づき、その必要とされるアセッ

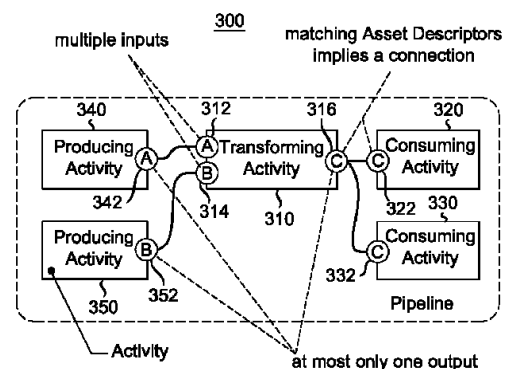


FIG. 3

**【特許請求の範囲】****【請求項 1】**

ワークフローモデルにおけるセットを処理する方法であって、

同じタイプの複数のアセットを有する第 1 のセットを示す関連するアセット記述子を持った少なくとも第 1 の入力を有する第 1 のアクティビティのグラフィカル表現を供給するステップと、

同じタイプの複数のアセットを有する第 2 のセットを示す関連するアセット記述子を持った少なくとも出力を有する第 2 のアクティビティのグラフィカル表現を供給し、前記第 2 のセットのメンバが、前記第 1 のアクティビティのグラフィカル表現の第 1 の入力と関連する前記第 1 のセットのメンバと一致するステップと、

10

前記第 2 のアクティビティのグラフィカル表現の出力と関連する前記第 2 のセットのメンバが前記第 1 のアクティビティのグラフィカル表現の第 1 の入力と関連する前記第 1 のセットのメンバと一致することに基づき、前記第 1 のアクティビティのグラフィカル表現の第 1 の入力を前記第 2 のアクティビティのグラフィカル表現の出力と接続するステップと

を有する方法。

**【請求項 2】**

前記グラフィカル表現の供給は、複数の可能なグラフィカル表現の中からグラフィカル表現を選択することを含む、

請求項 1 に記載の方法。

20

**【請求項 3】**

前記第 1 のアクティビティのグラフィカル表現は、前記第 1 の入力のアセット記述子とは異なるアセット記述子を持つ第 2 の入力を有する、

請求項 1 に記載の方法。

**【請求項 4】**

前記第 1 のアクティビティのグラフィカル表現の第 2 の入力のアセット記述子と一致する関連するアセット記述子を持った少なくとも出力を有する第 3 のアクティビティのグラフィカル表現を供給するステップと、

前記一致するアセット記述子に基づき、前記第 1 のアクティビティのグラフィカル表現の第 2 の入力を前記第 3 のアクティビティのグラフィカル表現の出力と接続するステップと

30

を更に有する請求項 3 に記載の方法。

**【請求項 5】**

前記第 2 のアクティビティのグラフィカル表現は、関連するアセット記述子を持つ少なくとも第 1 の入力を更に有し、

当該方法は、

前記第 2 のアクティビティのグラフィカル表現の第 1 の入力のアセット記述子と一致する関連するアセット記述子を持った少なくとも出力を有する第 3 のアクティビティのグラフィカル表現を供給するステップと、

前記一致するアセット記述子に基づき、前記第 2 のアクティビティのグラフィカル表現の第 1 の入力を前記第 3 のアクティビティのグラフィカル表現の出力と接続するステップと

40

を更に有する、請求項 1 に記載の方法。

**【請求項 6】**

アクティビティのグラフィカル表現のうちの少なくとも 1 つは、アクティビティテンプレートである、

請求項 1 に記載の方法。

**【請求項 7】**

アクティビティのグラフィカル表現のうちの少なくとも 1 つは、アクティビティインスタンスである、

50

請求項 1 に記載の方法。

【請求項 8】

アクティビティインスタンスのアセット記述子の特定のパラメータは、当該アクティビティインスタンスに接続されるあらゆるテンプレートアクティビティへパスされる、  
請求項 7 に記載の方法。

【請求項 9】

ワークフローモデルにおけるセットを処理する装置であって、  
ワークフロー情報を記憶するストレージと、  
処理するためのデータを記憶するメモリと、

同じタイプの複数のアセットを有する第 1 のセットを示す関連するアセット記述子を持った少なくとも第 1 の入力を有する第 1 のアクティビティのグラフィカル表現を供給し、  
同じタイプの複数のアセットを有する第 2 のセットを示す関連するアセット記述子を持った少なくとも出力を有する第 2 のアクティビティのグラフィカル表現を供給し、前記第 2 のセットのメンバが、前記第 1 のアクティビティのグラフィカル表現の第 1 の入力と関連する前記第 1 のセットのメンバと一致し、前記第 2 のアクティビティのグラフィカル表現の出力と関連する前記第 2 のセットのメンバが前記第 1 のアクティビティのグラフィカル表現の第 1 の入力と関連する前記第 1 のセットのメンバと一致することに基づき、前記第 1 のアクティビティのグラフィカル表現の第 1 の入力を前記第 2 のアクティビティのグラフィカル表現の出力と接続するよう構成されるプロセッサと  
を有する装置。

10

20

【請求項 10】

ネットワークへ接続するためのネットワーク接続を更に有する  
請求項 9 に記載の装置。

【請求項 11】

前記供給されるグラフィカル表現は、複数の可能なグラフィカル表現の中から選択されたグラフィカル表現を有する、  
請求項 9 に記載の装置。

【請求項 12】

アクティビティのグラフィカル表現のうちの少なくとも 1 つは、アクティビティテンプレートである、  
請求項 9 に記載の装置。

30

【請求項 13】

アクティビティのグラフィカル表現のうちの少なくとも 1 つは、アクティビティインスタンスである、  
請求項 9 に記載の装置。

【請求項 14】

アクティビティインスタンスのアセット記述子の特定のパラメータは、当該アクティビティインスタンスに接続されるあらゆるテンプレートアクティビティへパスされる、  
請求項 13 に記載の装置。

【請求項 15】

実行される場合に、  
同じタイプの複数のアセットを有する第 1 のセットを示す関連するアセット記述子を持った少なくとも第 1 の入力を有する第 1 のアクティビティのグラフィカル表現を供給するステップと、  
同じタイプの複数のアセットを有する第 2 のセットを示す関連するアセット記述子を持った少なくとも出力を有する第 2 のアクティビティのグラフィカル表現を供給し、前記第 2 のセットのメンバが、前記第 1 のアクティビティのグラフィカル表現の第 1 の入力と関連する前記第 1 のセットのメンバと一致するステップと、  
前記第 2 のアクティビティのグラフィカル表現の出力と関連する前記第 2 のセットのメンバが前記第 1 のアクティビティのグラフィカル表現の第 1 の入力と関連する前記第 1 の

40

50

セットのメンバと一致することに基づき、前記第 1 のアクティビティのグラフィカル表現の第 1 の入力を前記第 2 のアクティビティのグラフィカル表現の出力と接続するステップと

を実行する命令を含む機械可読媒体。

【発明の詳細な説明】

【技術分野】

【0001】

本発明は、ワークフロー及び／又は生産パイプラインのモデリング、特に、生産パイプラインに沿って必要とされ且つ生成されるアセットに基づき生産パイプラインをモデリングする方法及び装置に関する。

10

【背景技術】

【0002】

あらゆる映画、テレビ、又はレコード製品に関し、新しいフォーマットにおいて新しいアセットを生成することができる、異なった製作会社間でやり取りされる幾つかの異なったフォーマットにあるショット、視覚効果、音響などのような幾つかのアセットが存在する。それらのアセットは、テレビ番組、映画、レコード、又は同様のものを作るのに使用されるビルディングブロックである。そのようなアセットの経過を追い、アセットの場所及び状態がどのようであるかを知ることが、途方もなく複雑な試みである。

【0003】

プロジェクトマネージャプログラムが存在するが、それらは、通常は、プロジェクトの段階を表す各ブロックがシステムを作るためにユーザによって次のブロックと関連付けられるトップダウンの静的なシステム設計のために設計されている。ブロックは、アセットに基づき結合せず、また、プログラムは、そのようなアセットを追跡しない。

20

【発明の概要】

【発明が解決しようとする課題】

【0004】

よって、映画、テレビ、音楽アルバムなどの生成において使用されるアセットに基づきワークフロー及び／又は生産パイプラインをモデリングすることができる方法及びシステムが必要とされる。

【課題を解決するための手段】

30

【0005】

アセットにより駆動されるワークフローの依存管理は、各アクティビティについて入力及び／又は出力として使用されるアセットの記述に基づき、アクティビティ間の接続を確立する。そのような記述的な“コントラクト”は、所望の出力を生成するのに必要な関連するアクティビティを容易に照合するメカニズムを提供する。所望のワークフローのグラフィカルモデルを生成することによって、ユーザは、何がワークフローに関係しているかと、どこで問題及び冗長が起こり得るのかとについてのより良い理解を得られる。グラフィカルモデルは、現実世界の製品を設計し且つ監視するために使用され得る。

【0006】

アクティビティのグラフィカル表現は、ベンダー、ファシリティ、及び他の生産アクティビティをモデリングするために使用される。アクティビティモデルは、アクティビティ間でやり取りされる成果物を表すアセットを生成及び／又は消費する。アクティビティのモデルを用いて、生産パイプラインのモデルは後ろから前へ構築され得る。よって、最終結果のアクティビティモデルは最初に選択され、選択された最終結果のアクティビティによって必要とされるアセットに基づき、その必要とされるアセットを生成する適切なアクティビティは選択され得る。このプロセスは、プロセスパイプラインの開始に達するまで繰り返され得る。現実世界のプロセスパイプラインは、次いでモデルに基づき形成され得る、モデルは、現実世界の生産パイプラインの状態を追跡するために使用され得る。そのような技術は、より効率的な追跡のためにアセットのセットの処理を更に支援することができる。

40

50

## 【 0 0 0 7 】

本開示の一実施形態は、ワークフローモデルにおけるセットを処理する方法を提供する。方法は、同じタイプの複数のアセットを有する第1のセットを示す関連するアセット記述子を持った少なくとも第1の入力を有する第1のアクティビティのグラフィカル表現を供給するステップと、同じタイプの複数のアセットを有する第2のセットを示す関連するアセット記述子を持った少なくとも出力を有する第2のアクティビティのグラフィカル表現を供給し、前記第2のセットのメンバが、前記第1のアクティビティのグラフィカル表現の第1の入力と関連する前記第1のセットのメンバと一致するステップと、前記第2のアクティビティのグラフィカル表現の出力と関連する前記第2のセットのメンバが前記第1のアクティビティのグラフィカル表現の第1の入力と関連する前記第1のセットのメンバと一致することに基づき、前記第1のアクティビティのグラフィカル表現の第1の入力を前記第2のアクティビティのグラフィカル表現の出力と接続するステップとを有する。

10

## 【 0 0 0 8 】

本開示の他の実施形態は、ワークフローモデルにおけるセットを処理する装置を提供する。装置は、ストレージ、メモリ及びプロセッサを有する。ストレージ及びメモリは、データを記憶するためのものである。プロセッサは、同じタイプの複数のアセットを有する第1のセットを示す関連するアセット記述子を持った少なくとも第1の入力を有する第1のアクティビティのグラフィカル表現を供給し、同じタイプの複数のアセットを有する第2のセットを示す関連するアセット記述子を持った少なくとも出力を有する第2のアクティビティのグラフィカル表現を供給し、前記第2のセットのメンバが、前記第1のアクティビティのグラフィカル表現の第1の入力と関連する前記第1のセットのメンバと一致し、前記第2のアクティビティのグラフィカル表現の出力と関連する前記第2のセットのメンバが前記第1のアクティビティのグラフィカル表現の第1の入力と関連する前記第1のセットのメンバと一致することに基づき、前記第1のアクティビティのグラフィカル表現の第1の入力を前記第2のアクティビティのグラフィカル表現の出力と接続するよう構成される。

20

## 【 0 0 0 9 】

目的及び利点は、特許請求の範囲において特に指し示されている要素及び結合を手段として実現され且つ達成される。留意すべきは、開示される実施形態は、本願では革新的な教示の多数の有利な使用の例でしかない点である。前述の概要及び以下の詳細な説明はいずれも例示及び説明のためのものであり、請求される発明を制限しない点が理解されるべきである。更に、幾つかの記述は、幾つかの発明の特徴に当てはまるが他には当てはまらないことがある。一般に、別なふうを示されない限り、単数要素は、一般性を損なわずに複数個であってよく、その逆も同様である。図面において、同じ符号は、幾つかの図を通じて同じ部分を参照する。

30

## 【 図面の簡単な説明 】

## 【 0 0 1 0 】

【 図 1 】実施形態に従って、アセットにより駆動されるワークフローのモデリングが実装され得るシステムのブロック略図を表す。

【 図 2 】実施形態に従って、アセットにより駆動されるワークフローのモデリングの手順を実装する電子装置のブロック略図を表す。

40

【 図 3 】実施形態に従って、アセットにより駆動されるワークフローモデルのブロック略図を表す。

【 図 4 】実施形態に従って、アセットにより駆動されるワークフローのモデリングのための手順の例となるフローチャートを表す。

【 図 5 】実施形態に従って、図 4 のフローチャートのステップを説明する例となる図を表す。

【 図 6 】実施形態に従って、アセットにより駆動されるワークフローモデルを実装するセットのブロック略図を表す。

【 図 7 】実施形態に従って、アクティビティのグラフィカル表現の例となる図を表す。

50

【図 8】実施形態に従って、アセット記述子に基づくアクティビティの照合の例となる図を表す。

【図 9】実施形態に従って、アセット記述子に基づくアクティビティテンプレート及びアクティビティインスタンスの照合の例となる図を表す。

【図 10】実施形態に従って、例となるアセット記述子及びパラメータに基づくそれらの照合の表を表す。

【図 11 A】実施形態に従って、アセット記述子のパラメータの伝搬の例となる図を表す。

【図 11 B】実施形態に従って、アセット記述子のパラメータの伝搬の例となる図を表す。

【図 12】実施形態に従って、アセットにより駆動されるワークフローのモデリングにおいてアセットの状態を提供する手順の例となるフローチャートを表す。

【図 13】実施形態に従って、図 12 のフローチャートのステップを説明する例となる図を表す。

【図 14】実施形態に従って、共有ファシリティに関連するアセットの追跡の例となる図を表す。

【図 15】実施形態に従って、アセット、アクティビティ、ベンダー、及びファシリティの間の関係を説明する例となる図を表す。

【図 16】実施形態に従って、プロセス情報をアセットデータにマッピングする手順の例となるフローチャートを表す。

【図 17】実施形態に従って、図 16 のフローチャートのステップを説明する例となる図を表す。

【図 18】実施形態に従って、プロデューサ作業空間の例となるスクリーンショットを表す。

【図 19】実施形態に従って、図 18 のプロデューサ作業空間の成果物ダッシュボードの単離されたスクリーンショットを表す。

【図 20】実施形態に従って、図 18 のプロデューサ作業空間のフィルタ処理されたパイプラインの単離されたスクリーンショットを表す。

【図 21】実施形態に従って、図 18 のプロデューサ作業空間のアクティビティ詳細の単離されたスクリーンショットを表す。

【図 22】実施形態に従って、マネージャ作業空間の例となるスクリーンショットを表す。

【図 23】実施形態に従って、データ I / O 作業空間の例となるスクリーンショットを表す。

【図 24】実施形態に従って、エグゼクティブ作業空間の例となるスクリーンショットを表す。

【図 25】実施形態に従って、パイプラインビルダーの例となるスクリーンショットを表す。

【発明を実施するための形態】

【0011】

これより図 1 を参照すると、アセットにより駆動されるワークフローのモデリングを実装するシステム 100 の実施形態のブロック図が与えられている。システムは、サーバ 110 と、例えば、スマートフォン 120、例えばデスクトップ又はラップトップなどのパーソナルコンピュータ (PC) 130、及びタブレット 140 などの、インターネット 150 を介してサーバ 110 と通信する 1 つ以上の電子装置とを有する。ある実施形態では、サーバ 110 は、処理及び記憶を含む、アセットにより駆動されるワークフローのモデリングのための環境を提供する。ユーザは、例えばスマートフォン 120、PC 130 又はタブレット 140 などの電子装置におけるブラウザ又はアプリケーションを用いて、サーバ 110 におけるアセットにより駆動されるワークフローモデルとインタフェース接続する。他の実施形態では、アセットにより駆動されるワークフローのモデリングの一部又

10

20

30

40

50

は全ては、例えば、スマートフォン 120、例えばデスクトップ又はラップトップなどのパーソナルコンピュータ (PC) 130、及びタブレット 140 などの 1 つ以上の電子装置において実行され得る。

【0012】

図 2 は、アセットにより駆動されるワークフローのモデリングのための手順及びシステムを実装するのに使用され得る例となるサーバ 200 又は電子装置を表す。サーバ又は電子装置は、1 つ以上のプロセッサ 210、メモリ 220、ストレージ 230、及びネットワークインタフェース 240 を有する。それらの要素の夫々は、以下でより詳細に論じられる。

【0013】

プロセッサ 210 は、サーバ 210 又は電子装置の動作を制御する。プロセッサ 210 は、サーバ又は電子装置を動作させるソフトウェアを起動するとともに、アセットにより駆動されるワークフローのモデリングアプリケーションの機能性を提供する。プロセッサ 210 は、メモリ 220、ストレージ 230、及びネットワークインタフェース 240 へ接続され、それらの要素の間の情報の転送及び処理を扱う。プロセッサ 210 は、汎用のプロセッサ、又は特定の機能性に専用のプロセッサであることができる。ある実施形態では、複数のプロセッサが存在することができる。

【0014】

メモリ 220 は、プロセッサによって実行される命令及びデータが記憶されるところである。メモリ 220 は、揮発性メモリ (RAM)、不揮発性メモリ (EEPROM)、又は他の適切な媒体を含むことができる。

【0015】

ストレージ 230 は、本開示の低温貯蔵の推奨手順を実行する際にプロセッサによって使用及び生成されるデータが記憶されるところである。ストレージは、磁気媒体 (ハードドライブ)、光媒体 (CD/DVD-ROM)、又はフラッシュに基づくストレージであってよい。他のタイプの適切なストレージは、本開示の利点を鑑みて当業者に明らかであろう。

【0016】

ネットワークインタフェース 240 は、ネットワーク上での他の装置とのサーバ 200 又は電子装置の通信を扱う。適切なネットワークの例には、Ethernet (登録商標) ネットワーク、Wi-Fi 対応ネットワーク、セルラーネットワーク、及び同様のものがある。他のタイプの適切なネットワークは、本開示の利点を鑑みて当業者に明らかであろう。

【0017】

図 2 において挙げられている要素は例示である点が理解されるべきである。サーバ 200 又は他の電子装置は要素を幾つでも有することができ、ある要素は他の要素の機能性の一部又は全てを提供することができる。他の可能な実施は、本開示の利点を鑑みて当業者に明らかであろう。

【0018】

図 3 は、アセットにより駆動されるワークフローのグラフィカルモデル 300 を表す。アセットにより駆動されるワークフローの依存管理は、各アクティビティについて入力及び/又は出力として使用されるアセットの記述に基づき、アクティビティ間の接続を確立する。そのような記述的な“コントラクト”は、所望の出力を生成するのに必要な関連するアクティビティを容易に照合するメカニズムを提供する。図 3 は、変換アクティビティ 310 が入力 312、314 としてアセット A 及びアセット B を必要とし、そして、出力 316 において提供される結果としてアセット C を生成することを示す。消費アクティビティ 320、330 は、入力 322、332 においてアセット C を期待し、一方、生成アクティビティ 340、350 はいずれも、出力 342、352 において、モデリングされるシステムにアセット A 及び B をもたらす。この例となるパイプライン 300 では、生成アクティビティ 340 及び 350 の出力 342、352 は、変換アクティビティ 310 の

10

20

30

40

50

入力 3 1 2、3 1 4 へ接続されている。変換アクティビティ 3 1 0 の出力 3 1 6 は、次いで、消費アクティビティ 3 2 0 及び 3 3 0 の入力 3 2 2、3 3 2 へ接続されている。

【 0 0 1 9 】

[ 幾つかの作業定義 ]

パイプライン：所望の出力を生成するためにつなぎ合わされたアクティビティの集合。パイプラインはワークフローのグラフィカルモデルを提供する。ビデオ又はフィルム製作の例に関し、パイプラインは、所望の製品を生成するのに必要な全てのアクティビティ（例えば、データ、特定のショット、フォーマット、又はオーディオトラックの生成）を表す。

【 0 0 2 0 】

アクティビティ：アセット（例えばデータ、特定のショット、フォーマット、又はオーディオトラックなどの成果物を含む。）を生成し、変換し、又は消費する動作。各アクティビティは入力、出力又はその両方を有してよい。簡略化する前提として、アクティビティは、通常は、単一の出力しか有さない（なお、出力は複合又は合成アセットであってよい。）。アクティビティ（消費アクティビティを除く。）は、その出力によって容易に特性化され得る。アクティビティを一意的にするものは、所与の出力を生成するようアクティビティを介してマッピングされる入力の具体的なコンフィグレーションである。異なるアクティビティは同じ出力を生成してよく、従って、ただ 1 つが所与のパイプライン内で必要とされる。所与のアクティビティの出力は、複数の下流アクティビティに入力を提供することができる。

【 0 0 2 1 】

接続：アクティビティの出力記述が 1 つ以上の入力記述と一致する場合に、接続は暗示される。接続は、アセットの送出及び受取のための同意又はコントラクトの履行を表す。

【 0 0 2 2 】

アセット記述子：アクティビティ間の接続を照合し且つ確立するのに使用されるアクティビティ及びその入出力のラベル。

【 0 0 2 3 】

それらの概念の更なる議論は、本明細書において後で与えられている。

【 0 0 2 4 】

図 4 は、ワークフローのグラフィカル表現を生成するプロセスのフロー図 4 0 0 である。基本的に、プロセスは 3 つのステップを含む。関連するアセット記述子を持った少なくとも 1 つの入力を有する第 1 のアクティビティのグラフィカル表現を供給すること（ステップ 4 1 0）、第 1 のアクティビティのグラフィカル表現の入力のアセット記述子と一致する関連するアセット記述子を持った少なくとも 1 つの出力を有する第 2 のアクティビティのグラフィカル表現を供給すること（ステップ 4 2 0）、及び一致したアセット記述子に基づき、第 2 のアクティビティのグラフィカル表現の出力を第 1 のアクティビティのグラフィカル表現の入力とグラフィカルに接続すること（ステップ 4 3 0）である。それらのステップのグラフィカルな例 5 0 0 は図 5 で見られる。

【 0 0 2 5 】

[ ワークフローのモデリング ]

図 5 のグラフィカルな例 5 0 0 において示されているステップ 4 1 0 は、第 1 のアクティビティ 5 1 0 のグラフィカル表現を供給することから開始する。この実施形態では、第 1 のアクティビティのグラフィカル表現は、所望のアセットの記述子（この場合に “ H ”）を持った 1 つの入力 5 1 2 を有する。他の実施形態では、第 1 のアクティビティ 5 1 0 のグラフィカル表現は、異なった関連するアセット記述子を持った複数の入力を有してよい。第 1 のアクティビティの供給されるグラフィカル表現は、アクティビティの複数の供給されるグラフィカル表現から選択されるグラフィカル表現であってよい。グラフィカル表現の選択は、所望の又は必要とされるアクティビティに基づきグラフィカルユーザインタフェースを用いるユーザによって又はシステム自体によって行われ得る。幾つかの環境では、特定のアセット記述子と適合することができるアクティビティの全てが使用される

10

20

30

40

50



わけではない。

#### 【 0 0 2 6 】

図 5 のグラフィカルな例 5 0 0 のステップ 4 2 0 で、第 2 のアクティビティの少なくとも 1 つのグラフィカル表現が供給される。この例では、システムは、第 1 のアクティビティ 5 1 0 のアセット記述子 ( “ H ” ) と一致する関連するアセット記述子を持った出力を有するアクティビティを探す。所望のアクティビティを出力する 1 よりも多いアクティビティが存在し得るが、ただ 1 つが選択される必要がある。選択は、ユーザによって又はシステムによって実行され得る。この例では、第 1 のアクティビティ 5 1 0 の入力 5 1 2 のアセット記述子 ( “ H ” ) と一致する関連するアセット記述子を持った出力を有する 2 つの可能なアクティビティ 5 2 0 、 5 3 0 が存在する。1 つの可能な第 2 のアクティビティ 5 2 0 は、一致するアセット記述子 ( “ H ” ) をを持った出力 5 2 4 とともに、異なった関連するアセット記述子 ( “ A ” ) をを持った入力 5 2 2 を有する。一致するアセット記述子 ( “ H ” ) をを持った出力 5 3 6 を有する他の可能な第 2 のアクティビティ 5 3 0 は、異なった関連するアセット記述子 ( “ D ” 及び “ E ” ) をを持った 2 つの入力 5 3 2 、 5 3 4 を有する。

10

#### 【 0 0 2 7 】

所望の第 2 のアクティビティ、この場合にアクティビティ 5 2 0 をパイプライン内に選択することは、第 2 のアクティビティ 5 2 0 の出力に関連するアセット記述子が第 1 のアクティビティ 5 1 0 の入力 5 1 2 のアセット記述子と一致するので、アクティビティ 5 1 0 、 5 2 0 の間の接続を暗示する。暗示された接続は、ステップ 4 3 0 において、グラフィカル接続 5 4 0 として表されている。

20

#### 【 0 0 2 8 】

この実施形態では、照合及び接続は、アセット自体ではなくアセット記述子に基づく。このことは、実際のアセットが存在する前に完全なパイプラインモデルの生成を可能にする。そのようなアセット駆動のモデリングの幾つかの利点には、アクティビティが出力又は消費するアセットの記述に基づくそれらのアクティビティの明示的なマッピング、アセットの来歴がシステムを通じて明示的に追跡され得ること、及び下流の依存性が容易に計算され得ることがある。

#### 【 0 0 2 9 】

[ ワークフロー / パイプラインのモデリング ( セット ) ]

30

メディア製作の世界では、より大きいセットの部分として多数の同じ要素を生成するアクティビティに遭遇することが一般的である。例えば、 “ デイリーズ ( Dailies ) ” アクティビティは、最初に捕捉されたビデオ及びオーディオの “ ショット ” を、ディレクター又はプロデューサが見直し且つ承認するための容易にレビュー可能なフォーマットへと変換することに関与する。

#### 【 0 0 3 0 】

一例として、デイリーズアクティビティは、数週間の期間にわたって 1 0 0 0 の “ ショット ” を処理することに関与してよい。更に、それらの “ ショット ” は、異なるカメラユニットから不連続に伝来してよい。 “ デイリーズ ” アクティビティの出力は、毎日のように次のステップへ定期的に送られる。

40

#### 【 0 0 3 1 】

そのようなシステムのモデリングに関し、セットの使用が有益であり得る。セットは、同じタイプから成る 1 つ以上のアセットの集合である。セットの各メンバは一意的なアセットであるが、そのセット内の他のアセットと同じタイプ又はクラスから成る。例えば、セットは 5 0 0 のショットを有してよいが、そのセットの各メンバは 1 ショットである。セットは、複数のアクティビティにわたって作業生成物を分配し且つ積み上げるために使用され得る。セットはまた、同様に、サブセットに分割され得る。従って、アクティビティは、異なるアクティビティから異なるサブセットを受け取ってよく、あるいは、アクティビティは、生成されたオリジナルの一部分のみを消費してよい。

#### 【 0 0 3 2 】

50

セットを用いてワークフローをモデリングする手順は、図4で説明される非セットのアクティビティにより駆動されるモデリングのための手順と同様である。第1及び第2のアクティビティは、関連するアセット記述子に基づき供給されて接続される。しかし、この場合に、アセット記述子は、アセットのセットが使用されていることを示す。この例は、図6のワークフローモデル600で見られる。

#### 【0033】

図6のワークフローモデル600において、第1のアクティビティ610のグラフィカル表現が供給される。第1のアクティビティ610は消費アクティビティであり、関連するアセット記述子(この場合に“D”)を持った入力612を有する。この例では、アセット記述子は、入力612で受け取られると期待されるアセットのセット(この場合に、ショット1~25)が存在することを更に示す。第2のアクティビティ620のグラフィカル表現が更に供給される。第2のアクティビティ620は変換アクティビティであり、関連するアセット記述子(“D”)を持った出力622を有する。しかし、この場合に、アセット記述子は、出力622で供給されるべきアセット(ショット1~100)のより大きいセットが存在することを示す。しかし、各セットのメンバアセットの幾つかは一致するので、接続が暗示され、グラフィカルに示される670。

#### 【0034】

図6の実施形態では、第3のアクティビティ630及び第4のアクティビティ640のグラフィカル表現が更に供給される。第3及び第4のアクティビティは消費アクティビティであり、関連するアセット記述子(“D”)を持った入力632、642を有する。アセット記述子は、入力632、642がアセットのセットを受け取るべきことを更に示す。第3のアクティビティ630の場合に、セットはショット26~75を有する。第4のアクティビティ640の場合に、セットはショット76~100を有する。第3及び第4のアクティビティ630、640のセットは第2のアクティビティ620のセットのサブセットであるから、各自のセットの中には一致するメンバが存在し、接続が第2のアクティビティ620と第3のアクティビティ630との間及び第2のアクティビティ620と第4のアクティビティ640との間で暗示され、グラフィカルに示される672、674。

#### 【0035】

上述されたように、図6のモデル600における第2のアクティビティ620は変換アクティビティである。そのようなものとして、第2のアクティビティ620は、関連するアセット記述子(この場合に“S”)を持った入力624を更に有する。この例では、関連するアセット記述子は、入力624で受け取られると期待されるアセット(この場合に、ショット1~100)のセットが存在することを更に示す。よって、第2のアクティビティ620は、その入力624でショット1~100を有するアセット“S”のセットを受け取り、その出力622でショット1~100を有するアセット“D”のセットを生成するプロセス又はオペレータをモデリングする。

#### 【0036】

第5のアクティビティ650及び第6のアクティビティ660のグラフィカル表現がまた、図6のモデル600において供給される。第5のアクティビティ650及び第6のアクティビティ660は生成アクティビティであり、関連するアセット記述子(“S”)を持った出力652、662を有する。アセット記述子は、出力652、662がアセットのセットを生成すべきことを更に示す。第5のアクティビティ650の場合に、セットはショット1~50を有する。第6のアクティビティ660の場合に、セットはショット50~100を有する。第5及び第6のアクティビティ650、660のセットは、第2のアクティビティ620の入力624で受け取られるセットのサブセットであるから、各自のセットの中には一致するメンバが存在し、接続が第5のアクティビティ650と第2のアクティビティ620との間及び第6のアクティビティ660と第2のアクティビティ620との間で暗示され、グラフィカルに示される680、682。

#### 【0037】

10

20

30

40

50

### [ アセット記述子 ]

明らかなように、アセット記述子は、アクティビティ間の接続を生成し且つ潜在的なアクティビティ接続を特定するよう入力及び出力をモデリングするために使用される。ある実施形態では、アセット記述子は、アセットレジストリ内の既存のアセットと関連するために使用される。

#### 【 0 0 3 8 】

ある実施形態では、アセット記述子は、正確な又はパラメータ化された照合のために使用され得る。このとき、パラメータ化された照合は、記述子が比較される場合に、幾つかのワイルドカード様の機能を提供する。本例では、完全に定義されたアセット記述子は、円で囲まれた大文字により表される。例えば：

#### 【 0 0 3 9 】

##### [ 外 1 ]



パラメータ化されたアセット記述子は、円で囲まれた“ プライム付き ”大文字により表され得る。例えば：

#### 【 0 0 4 0 】

##### [ 外 2 ]



### [ 幾つかの更なる定義 ]

- ・アクティビティインスタンス。これは、全ての入力及び出力アセット記述子が完全に定義されているアクティビティであり、未定義のパラメータが存在しないことを意味する。
- ・アクティビティテンプレート。これは、再利用を容易にする1つ以上のパラメータ化されたアセット記述子を有するアクティビティである。しかし、これは必要条件ではない。

#### 【 0 0 4 1 】

アクティビティは、それらの入力及び出力に関して定義される。入力及び出力は、つまり、それらのアセット記述子によって定義される。入力及び出力の特定の組み合わせは、アクティビティの“ シグニチャ ”を決定する（それがどのように名付けられ得るかに関わらず）。図7の例では、アクティビティ1 700は、入力702、704でアセット（A）及びアセット（B）をとり、出力706でアセット（C）を供給する。アクティビティ2 710は、出力でアセット（C）を供給するが、入力712でアセット（X）をとる。この例では、それらのアクティビティの夫々は、それらが異なった入力を必要としながら、いずれも同じ出力を生成する点で、一意的である。

#### 【 0 0 4 2 】

アクティビティインスタンスの間の接続をモデリングするよう、上流のアクティビティの出力は下流のアクティビティの入力と一致する必要がある。複数の下流アクティビティは同じアセットを消費してよい。図8の第1のモデル800では、アクティビティインスタンス1（810）、アクティビティ2（820）、アクティビティインスタンス3（830）、及びアクティビティインスタンス4（840）が存在する。第2のモデル850では、接続は、アクティビティインスタンス1（810）がアセット（A）をインスタンス2（820）及び3（830）へ供給することにより示されている。アクティビティインスタンス3（830）は2つの入力（A）及び（B）を必要とする。アセット（B）はアクティビティインスタンス4（840）によって供給される。

#### 【 0 0 4 3 】

アクティビティインスタンス間の潜在的な接続をアクティビティテンプレートによりモデリングするよう、インスタンスの入力はテンプレートの出力とのテンプレートマッチ（又はその逆）を行うことができる。この例は図9のモデル900で見られる。図9で、ア

10

20

30

40

50

クティビティインスタンス 4 ( 9 3 0 ) の入力 9 3 2 のアセット記述子 ( A ) は、アクティビティテンプレート 1 ( 9 1 0 ) の出力 9 1 2 のアセット記述子 ( A ' ) と一致し、アクティビティインスタンス 5 ( 9 4 0 ) の出力 9 4 2 でのアセット記述子 ( B ) は、アクティビティテンプレート 2 ( 9 2 0 ) の入力 9 2 2 のアセット記述子 ( B ' ) と一致する。

#### 【 0 0 4 4 】

この時点までに単一文字を使用することは、開示されている概念を高いレベルで説明するための 1 つのアプローチであった。実際には、アセットを記述するための任意に多数の方法が存在する。一実施形態は、アセット記述子を生成するための人が読むことができ且つ柔軟なメカニズムを提供するために、名称 / 値の対の集合を使用する。アセット記述子は、アセットの一般的なアセット記述子フォーマットを記述すると全体としてとらえられる 1 つ以上の名称 / 値の対から構成され得る：

```
{
    name1:value1,
    name2:value2,
    name3:value3,
    ...
}
```

例：

```
{
    Title: 'The Hobbit',
    Version: 'Trailer',
    Type: 'Netflix Encoding'
}
```

パラメータされた記述は、簡単に、1 つ以上の値をブランクのままとする。以下の例では、“Title” 及び “Version” の両方がパラメータである。

#### 【 0 0 4 5 】

例：

```
{
    Title: ' ',
    Version: ' ',
    Type: 'Netflix Encoding'
}
```

アセット識別子は、アセット記述子の正規化されたバージョンである。最初に、アセット記述子は、名称 / 値の対の順序が比較に影響を及ぼさないように正規化される。アセット記述子を正規化するよう、名称及び値は小文字にされ（任意）、次いで名称によってソートされて、その結果が連結される。

例：

アセット記述子、すなわち、

```
{
    Title: 'The Hobbit',
    Version: 'Trailer',
    Type: 'Netflix Encoding'
}
```

は、アセット識別子、すなわち、

```
title: 'the hobbit', type: 'netflix encoding', version: 'trailer'
```

となる。

#### 【 0 0 4 6 】

任意に、暗号学的ハッシュが、以下で示される一意の数字（１６進）識別子を生成するために、上記の結果に対して実行され得る：

147c21df6e470da7879307dbfb2e2a5d3e9c40719ba2a1a840bf71c732f71b2f

アセット識別子の暗号学的ハッシュ変異は、テキストバージョンが多くの問題を抱えずぎて使い勝手が悪いHTMLにおいてクラス又はIDパラメータとしてユーザインタフェース要素を識別する場合に特に有用である。

#### 【 0 0 4 7 】

アセット参照は、アンダースコア“\_”文字によって分離された原のアセット記述子において定義された順序で値を連結させる。これは、アセットをそれほど形式的に記述しない人が読むことができるショートハンドを提供する。

#### 【 0 0 4 8 】

上記の例は、次のようになる：

‘The\_Hobbit\_Trailer\_Netfix\_Encoding’

アセット記述子及びアセット識別子は両方とも、完全な識別のために使用され得る。しかし、アセット参照は、単に表示の便宜のためであり、一義的な参照のために当てにされるべきでない。

#### 【 0 0 4 9 】

完全に定義されたアセット記述子の正確な照合は、アセット識別子を用いて直接に実行され得る。

#### 【 0 0 5 0 】

完全に定義されたアセット記述子をパラメータ化された記述子と照合する場合に、次の規則が使用される：

- ・ 名称／値の対は、比較のために小文字にされる。
- ・ パラメータ化されたアセット記述子は、完全に定義されたアセット記述子と正確に同じ名称エントリを有すべきである。名称の順序は重要でない。
- ・ パラメータ化されたアセット記述子のエントリがある値についてブランクを有する場合は、それは、完全に定義されたアセット記述子における対応する値にかかわらず一致する。パラメータ化されたアセット記述子がある値について非ブランクのエントリを有する場合は、それは正確に一致すべきである（正規化後）。

#### 【 0 0 5 1 】

例として、図１０の例となる表１０００が示される。

#### 【 0 0 5 2 】

アクティビティのパイプラインを構築する場合に、所望の出力（消費アクティビティ）を選択し、所望のパラメータ値を代入し、次いで、アクティビティがパイプラインを完了すると特定されるにつれて所望の値が伝播することを可能にすれば足りるべきである。図１１の例は、矢印１１０２によって示されるように終わりから始まりへと構築されるパイプライン１１００を詳述する。パイプラインはまた、始まりから終わりへ又は途中から構築されてよい。

#### 【 0 0 5 3 】

図１１Ｂのステップ１．０（１１１０）は終了アクティビティで始まる。この例では、供給されるアクティビティは、選択されたアクティビティテンプレート１１１２である。それに関し、入力のためのアセット記述子“ A ’ ”のパラメータは、アクティビティテンプレートをアクティビティインスタンスとするよう指定される。アセット記述子“ A ’ ”の指定されたパラメータは、次いで、第２の供給されるアクティビティテンプレート１１２２へ接続１１１４を通じて送られ得る。

#### 【 0 0 5 4 】

ステップ２．０（１１２０）で、接続１１１４を通じて送られた指定されたパラメータは、供給された第２のアクティビティテンプレート１１２２の入力に関連するアセット識

10

20

30

40

50

別子（“ B ’ ” 及び “ C ” ）のパラメータを指定するために使用され、このアクティビティテンプレートをアクティビティインスタンスにする。この例では、アセット記述子 “ C ” のためのLanguageのパラメータは、それが予め指定されているので、送られていない。

#### 【 0 0 5 5 】

図 1 1 A のステップ 3 . 0 ( 1 1 3 0 ) で、第 2 のアクティビティインスタンスからの指定されたパラメータは、供給された第 3 のアクティビティテンプレート 1 1 3 2 へ接続 1 1 2 4 を通じて送られる。送られた指定されたパラメータは、供給された第 3 のアクティビティテンプレート 1 1 3 2 の出力に関連するアセット識別子 “ C ’ ” のためのパラメータを指定するために使用され、このアクティビティテンプレートをアクティビティインスタンスにする。

10

#### 【 0 0 5 6 】

ステップ 4 . 0 ( 1 1 4 0 ) で、第 2 のアクティビティインスタンスからの指定されたパラメータは、供給された第 4 のアクティビティテンプレート 1 1 4 2 へ接続 1 1 2 6 を通じて送られる。送られた指定されたパラメータは、供給された第 4 のアクティビティテンプレート 1 1 4 2 の出力に関連するアセット識別子（“ B ’ ”）のためのパラメータを指定するために使用され、このアクティビティテンプレートをアクティビティインスタンスにする。

#### 【 0 0 5 7 】

実際のコンテンツ生成及び分配パイプラインをモデリングすることにおいて、以下のヒューリスティクスが有用であることが分かる：

20

- ・全てのアクティビティ記述子は “ Title ” 及び “ Version ” を含むべきである。それらは、全体のパイプラインインスタンスのためのアセットを区別する。
- ・全てのアクティビティ識別子は “ Type ” を含むべきである。Type フィールドは、アクティビティによって生成されるコンテンツのタイプ（ビデオ、オーディオ、デジタルシネマパッケージなど）を表す。
- ・他の有用なアセット記述子エントリは “ Language ” 、 “ AspectRatio ” 、 “ DubSubOV ” 、及び “ Format ” である。それは、 “ Type ” の値に基づき関連性があってもなくてもよい。他のエントリは、時間とともに使用に発展してよい。

#### 【 0 0 5 8 】

30

##### [ アセットレジストリ ]

ここで論じられているグラフィカルモデルのアクティビティ及びアセットはしばしば実際のアクティビティ又はアセットを表すことができるので、アセットレジストリを提供し且つ保持することが有益であり得る。アセットレジストリは、アセット記述子（又はアセット識別子）を実際のアセットの位置にマッピングする。完全に定義されたアセット記述子に対して既存のアセットを登録することによって、定義時にパイプラインから不必要なアクティビティを削除することが可能である。

#### 【 0 0 5 9 】

以前に定義されたパイプライン構築ストラテジへの変更として、レジストリを確認するステップは、アクティビティテンプレートを照合するあらゆる試みを続けることができる。所与のアセット記述子 / 識別子にマッピングされた異なる位置でのアセットの複数のコピーを有することが可能である。

40

#### 【 0 0 6 0 】

##### [ 履行モデリング ]

ここで記載されるモデリング手順において、アクティビティ及びそれらの接続は、アセット依存性に基づきモデリングされる（詳細はアセット記述子の項目を参照されたい。） 。アクティビティ間の夫々の接続は、1 つのアクティビティの出力についてのその後のアクティビティの入力に対する論理依存性を表す。アクティビティからアクティビティへの進行をたどるために、接続は履行ステータスを有することができる。ワークフローモデルにおいて履行ステータスをモデリングする例となる手順は、図 1 2 のフローチャート 1 2

50

00で見られる。

【0061】

その最も簡単なもので、方法は2つのステップを有する。第1のステップ(1210)は、第1のアクティビティのグラフィカル表現及び第2のアクティビティのグラフィカル表現を少なくとも有するワークフローのモデルを供給することであり、それらのアクティビティは、一致するアセット記述子に基づき接続される。第2のステップ(1220)は、第1及び第2のアクティビティのグラフィカル表現の間の少なくとも接続の根拠である一致するアセット記述子によって示される少なくとも1つのアセットのステータスを決定することである。それらのステップは、図13を参照して以下でより詳細に記載される。

【0062】

図13のダイアグラム1300において、ワークフローのモデルは、図12の手順のステップ1210で説明されたように提供される。この例では、モデルは、第1のアクティビティ1310(ここでは起点アクティビティ)及び第2のアクティビティ1320(ここでは目標アクティビティ)のグラフィカル表現を含む。第1のアクティビティ1310及び第2のアクティビティ1320は、一致するアセット記述子に基づき接続1330される。次いで、履行ステータス1340が、接続1330の根拠である一致するアセット記述子によって示される少なくとも1つのアセットについて決定される。

【0063】

履行ステータスは、1つのアクティビティから次のアクティビティへ移動する物理的/電子的アセットの状態を反映する。アクティビティが期待される出力(アセット)を生成した場合に、それは、従属する下流のアクティビティへ物理的/電子的に送られるので、プロセスは続くことができる。履行メカニズムは、アセット移動の状態を追跡する(例えば、保留中、送信中、受信済み、エラー)。ある実施形態では、履行ステータスは、グラフィカル表示されるか、あるいは、モデルのアクティビティ又は他の要素のグラフィカル表現の一部として別なふうに表示され得る。

【0064】

ある他の実施形態では、アクティビティのステータスは、アクティビティによって生成及び/又は消費されるアセットの履行ステータスに基づき決定され得る。ある更なる実施形態では、アクティビティのステータスは、グラフィカル表示されるか、あるいは、モデルのアクティビティ又は他の要素のグラフィカル表現の一部として別なふうに表示され得る。

【0065】

ある実施形態では、単一の物理的/電子的配信が複数の下流アクティビティによって利用可能であり、複数の履行記録は冗長的である。これを解決するよう、アクティビティ間の依存関係からアクティビティ-ファシリティ間の関係へと変更が行われ得る。この例は図14で見られる。

【0066】

図14の例となるダイアグラム1400において、アクティビティB1420及びアクティビティC1430は同じ共有ファシリティ1450(ファシリティY)にある。よって、接続1402は、起点アクティビティA1410と共有ファシリティ1450(ファシリティY)との間に示されている。目標アクティビティD1440は別のファシリティ(ファシリティZ)にあるので、別個の接続1440がアクティビティA1410とアクティビティD1440との間に設けられている。

【0067】

語“ファシリティ”は、システムにおいて使用される他の“位置”参照と区別するために選択されている。加えて、特定のアセット依存関係は、履行がファシリティのみならず、配信されるべき特定のアセットにも依存するように、依然として維持されるべきである。そのような実施形態において、履行ステータスは、このとき、起点アクティビティからファシリティへの特定のアセットの配信を表す。つまり、ファシリティは、複数の目標アクティビティによって共有されてよい。それらの要素の相互関係のダイアグラム1500

10

20

30

40

50

は図 15 で見られる。

【 0 0 6 8 】

図 15 のダイアグラム 1500 において、ワークフローのモデルが提供される。この例では、モデルは、第 1 のアクティビティ 1510（ここでは起点アクティビティ）及び第 2 のアクティビティ 1520（ここでは目標アクティビティ）のグラフィカル表現を含む。第 1 のアクティビティ 1510 と第 2 のアクティビティ 1520 との間の関係 1530 は、一致するアセット記述子に基づく。次いで、履行ステータス 1540 は、関係 1530 の根拠である一致するアセット記述子によって示される少なくとも 1 つのアセットについて決定される。この方法に対する実際的な制約として、ファシリティ 1560 は、第 2 のアクティビティ 1520 によって参照されるベンダー 1550 に関連付けられている。このように、ベンダー情報を変更することは、適切なファシリティ割り当てをもたらす。所与のファシリティは複数のベンダーによって参照されてよい。

10

【 0 0 6 9 】

履行ステータスは、アセット記述を共有するアクティビティの各対について参照され得る。履行ステータスを生成する場合に、目標アクティビティのベンダー、ファシリティ記述子は、履行が既に生成されているかどうかを決定するために使用され得る。そのような場合に、既存の履行記録が参照されてよく、そうでない場合には、新しい履行が生成され得る。ある実施形態では、逆ドメイン名シンタックスがファシリティ記述子のために使用され得るので、それは人が読むことができる（例えば、`technicolor.perivale`、`technicolor.perivale.transcodingDept`）。一意のファシリティワラントは履行を分離するが、同じ物理的位置において複数の“ファシリティ”が存在することができる。このことは、履行ステータスを独立に追跡する別個の記録をもたらす。

20

【 0 0 7 0 】

[ アセットデータへのプロセス情報のマッピング ]

ワークフローのモデリングは、この時点までに、モデリングされるパイプラインと一致する駆動アセット生成プロセス（アクティビティ）に焦点を当ててきた。言い換えると、システムは、定義されるモデルに基づき、どのようなアクティビティが何に依存するかを述べ、アセット記述子スキームに従ってアセットの登録を実施した。代替のアプローチは、これより、同様のレベルのパイプライン情報を配信するために、しかしながら、基礎をなすアクティビティの直接の影響を伴わない受動的な状態において、提供される。一般概念は、全体のプロセスのステータスを導出するためにアセットデータにわたってパイプラインモデル（プロセスデータ）をオーバーレイすることである。

30

【 0 0 7 1 】

アセットが生成される場合に、それらがアセットレジストリシステムにおいて登録されると考えられる。この手順は、上記のアセット記述子及びアセットレジストリの概念と一致する更なる構造化されたデータを必要とする。アセット記述子の名称 / 値は、一致する必要がある（例えば、`title`、`language`、`aspect ratio`など）。

【 0 0 7 2 】

アセット記述子を参照するアクティビティから成るプロセスモデルは、（おそらくプロセスレジストリから）取得され、アセットレジストリ内のデータを見てパイプラインステータス情報を得るために使用される。例となる手順は図 16 のフローチャート 1600 で見られる。

40

【 0 0 7 3 】

その最も簡単なもので、方法は 2 つのステップを含む。第 1 のステップ（1610）は、ワークフローのモデルに必要とされるアセットが存在することを決定することである。第 2 のステップ（1620）は、既存のアセットを必要とするアクティビティのグラフィカル表現を供給することである。それらのステップは、図 17 を参照して以下でより詳細に記載される。

【 0 0 7 4 】

図 17 のダイアグラム 1700 は 3 つの部分、すなわち、アセットレジストリ 1710

50



、プロセスモデル 1720、及び予想ステータス 1730 を有する。

【0075】

第1のステップ(1610)が実行されるのはアセットレジストリ 1710 内である。アセットが存在するかどうかを決定するよう、アセットレジストリはクエリされる。アセットレジストリは、例えばデータベースなどの、生成されたか又は予め存在しているアセットの集合である。この例では、アセットレジストリは所与のワークフローのためのアセットのみを含むと考えられる。しかし、当業者には当然ながら、アセットレジストリは、現在のワークフローモデルの部分でないアセットを含む登録されたアセットを幾つでも含んでよい。図17の例では、3つのアセット(A、B、C)が既に存在していると決定される。

10

【0076】

図17のプロセスモデル 1720 で、アセットを必要とするアクティビティのグラフィカル表現が供給される。そのようなアクティビティは、アセットを生成したか又は消費するアクティビティを含むことができる。ある実施形態では、生成及び消費の両アクティビティのグラフィカル表現が供給されてよく、更に接続され得る。複数の先在するアセットがある他の実施形態では、全てのアクティビティが接続されている全体のパイプラインのグラフィカル表現が供給され得る。ある実施形態では、プロセス又はモデルレジストリが供給されてよい。プロセスレジストリは、アセットレジストリと同じく、例えばデータベースなどの、既に生成されているか又は以前に使用されたことがあるパイプラインモデルの集合である。幾つかのそのような実施形態では、登録されているパイプラインモデルは、アセットレジストリ内の登録されているアセットと適合するか、あるいは、別なふう

20

【0077】

予想ステータス 1730 によれば、パイプラインモデルにおける各アクティビティについて、対応するアセット記述子がアセットレジストリからクエリされる。記述子に一致するアセットが見つけれられる場合に、そのアクティビティは完成していると考えられる。その場合に、どのようなアクティビティが進行中であるべきかを推測することは、それらのアクティビティへの入力

【0078】

完全であるが、現在のアクティビティの出力が完全でないかどうかを確かめることによって、可能である。ステータス表の例は1740で見られる。

30

【0079】

この基本メカニズムを用いて、データセットは多数の方法においてアプローチされ得る。

予め定義されたパイプライン：このシナリオでは、パイプラインの仕様は前もって知られている(すなわち、title、version、aspect ratioなどが定義される。)。このことは、アセットレジストリ内のアセットへの直接的なマッピングを可能にする。このアプローチの利点は、如何なる対応するアクティビティも未だ登録されていないパイプラインのステータスを見ることができる点である。

【0080】

既存のアセットからの予想：システムは、レジストリ内に既に存在するアセットについてのみならず既知のパイプラインを前提とすることができる。一例として、変換が特定のtitle、version、languageについて受け取られて登録される場合に、システムは、所与の値を持った入手変換アクティビティのためのインスタンスを生成し、次いで、出力を他のアクティビティ入力と照合し、順にそれらのアクティビティの出力をその後のアクティビティの入力と照合することによって、残りのパイプラインを予想する。プロセスは、見つけられたアクティビティのあらゆる入力を照合し、出力への入力経路をたどることによって、上流方向において同様に行われ得る。

40

【0081】

既存のデータに対してプロセスモデルをオーバレイする方法の特徴は、どのパイプラインのバリエーションが最も良く一致するかを確かめるために“ビューイングレンズ”

50

として複数のパイプラインを試みることができる点である。

#### 【0082】

前述の議論はワークフローのモデルを生成すること及びステータスをモニタリングすることに焦点を当ててきたが、ある実施形態では、パイプラインの概観を提供することが有利であり得る。そのようなものとして、ユーザインタフェースは、ワークフローのグラフィカルモデルを提供するのみならず、ワークフロープロセスの高度な観点を提供するために設けられてよい。そのようなユーザインタフェースの例は図18乃至25において見られる。それらの例は、ユーザが、例えば電子装置におけるウェブブラウザ又はアプリケーションを通じて、システムと対話する場合に提供され得るスクリーンショットである。

#### 【0083】

ある実施形態に従って、本開示のシステムが起動される場合に、ユーザはクレデンシャルをプロンプトされ、次いで、図18のスクリーンショット1800で表されるようなプロデューサの作業空間を提示される。この作業空間1800は、アクティビティ詳細とともにステータス及び依存性を見ながらエントリを生成する手段を提供する。ユーザはまた、アクティビティ詳細のパネルから操作プロセスを駆動することができる。プロデューサの作業空間1800は3つのパネル、すなわち、成果物ダッシュボード1810、フィルタ処理されたパイプラインビュー1820、及び詳細ビュー1830から成る。他の作業空間ビューは、ユーザによる選択1806のために利用可能であり、以下でより詳細に論じられる。

#### 【0084】

図19は、図18の成果物ダッシュボード1810の例である。ダッシュボードの各行は成果物1900に関係があり、成果物に関連するアクティビティ1910を示す。成果物ダッシュボード1810は更に、ユーザがtitle/version/formatをシステムに付加すること1920、既存のtitle/version/formatから選択すること1930、特定の成果物を要求するか又は新しい成果物を付加すること1940、及び新しいlanguage/aspect ratio/DubSub行を加えること1950を可能にする。ここで情報を入力することは、基礎をなすシステムに、必要条件を満たすよう適切なパイプラインを構築させる。システムは、特定のアクティビティが成果物間で共有されるかどうかを知ることができるほど十分に“高性能”である。

#### 【0085】

図20は、フィルタ処理されたパイプラインビュー1820の例を示す。フィルタ処理されたパイプラインビュー1820は、アクティビティ及びそれらの依存性のグラフィック表現を表示する。フィルタパイプラインビューは、リストビュー2010及び従来のパイプラインモデルビューの両方を提供する。このビューで示されるアクティビティは、成果物ダッシュボードにおいて選択された行に関係がある。各アクティビティ2030は、入力（左側にある円）及び出力（右側にある円）を持ったボックスとしてグラフィカルに表現されている。この実施形態では、アセットの履行ステータスは、アセットが受け取られたか又は生成されたことを示すよう入力及び/又は出力を塗りつぶすか又は別なふうに強調表示することによって示される。ある更なる実施形態では、アクティビティのステータスもグラフィカルに示され得る。本例では、ボックスが左下角に設けられており、ステータスを示すよう塗りつぶされ得る。空のボックスは、アクティビティが開始されていないことを意味し、部分的に塗りつぶされているボックスは、アクティビティが進行中であることを意味し、塗りつぶされたボックスは、アクティビティが実行されたことを意味する。

#### 【0086】

このパネルはまた、エグゼクティブ作業空間で利用可能である。その場合に、エグゼクティブパネルからアクティビティを選択することは、選択されたアクティビティに関連してパイプラインを表示する。

#### 【0087】

図21は、詳細ビュー1830の例である。詳細ビューパネル1830は、ボタンが設

10

20

30

40

50

けられ得る 3 つの簡単な動作に基づきアクティビティの状態を交信する能力を提供する：  
・Set to Ready ( 2 1 2 0 ) は、アクティビティが完了したことを示すために使用される。

・Send ( 図示せず。 ) は、アセットが次のアクティビティ ( 複数を含む。 ) へ送信された場合にシステムに知らせる。

・Receive ( 図示せず。 ) は、アセットが上流のアクティビティから受け取られた場合にシステムに知らせる。

#### 【 0 0 8 8 】

ある実施形態では、Revise動作は、アクティビティがレディー状態に設定される ( Set to Ready ) と提供され得る。修正 ( Revise ) は、ユーザが変更の影響を見て、その変更がやり直されるよう表明することを可能にする。

#### 【 0 0 8 9 】

それらの動作によって供給される情報は、アセットの履行ステータスとともに、アクティビティ自体のステータスを決定するために使用され得る。

#### 【 0 0 9 0 】

詳細ビューパネル 1 8 3 0 の現在の表示では、強調表示された “ ACTIONS ” タブによって示される動作が表示されている。 “ DETAILS ” タブが選択される場合は、期限及びベンダー情報に関する情報が表示される。日付が設定されており、その日付が渡過される ( 期限までに行われない ) 場合は、システムは、それに問題としてフラグを立てる。

#### 【 0 0 9 1 】

他の可能な作業空間は、図 2 2 のスクリーンショット 2 2 0 0 で表されるようなマネージャ作業空間である。マネージャ作業空間 2 2 0 0 は、マネージャパネル 2 2 1 0 及びアクティビティ詳細パネル 1 8 3 0 から成る。

#### 【 0 0 9 2 】

マネージャパネル 2 2 1 0 は、特定のアクティビティを通過する全ての作業を提示する。マネージャパネル 2 2 1 0 は、ユーザがフィールド 2 2 7 0 を用いて特定のアクティビティを選択することを可能にする。選択されたアクティビティに関連する作業及びタスクは、次いでパネル 2 2 1 0 において表示される。フィルタ 2 2 3 0 は、何が表示されているかを調整するのに使用され得る。デフォルトのフィルタは、従事している必要があるものしか示さないが、フィルタは、何が生じ、何が既に完了しているかを示すよう設定され得る。作業が行われると、 “ Set to Ready ” 2 2 4 0 が選択され得、タスクはボードから外れる ( フィルタが別なふうに設定されない限り ) 。マネージャ作業空間 2 2 0 0 のアクティビティ詳細パネル 1 8 3 0 は、図 2 1 を参照して記載されたように動作する。

#### 【 0 0 9 3 】

他の可能な作業空間は、図 2 3 のスクリーンショット 2 3 0 0 で表されるようなデータ I / O 作業空間である。データ I / O 作業空間 2 3 0 0 は、データ I / O パネル 2 3 1 0 及びアクティビティ詳細パネル 1 8 3 0 から成る。

#### 【 0 0 9 4 】

データ I / O パネル 2 3 1 0 は、特定のファシリティに関する全ての動作を示すよう設計される。それは、主として、ファシリティの内外でアセットを動かす人々が、アセットを生成又は変更する作業を行う人々とは異なることがあるという考えに適応するために送信及び受信動作を示すよう設計される。データ I / O パネル 2 3 1 0 は、ユーザがフィールド 2 3 2 0 を用いて特定のアクティビティを選択することを可能にする。選択されたアクティビティに関連する作業又はタスク及びそれらのステータス 2 3 4 0 は、次いでパネル 2 3 1 0 において表示される。フィルタ 2 3 3 0 は、何が表示されているかを調整するのに使用され得る。デフォルトのフィルタは、従事している必要があるものしか示さないが、フィルタは、何が生じ、何が既に完了しているかを示すよう設定され得る。例えば “ Set to Ready ” 、 “ Send ” 及び “ Receive ” などの動作 2 3 5 0 が選択され得、ステータス 2 3 4 0 は然るべく更新される。データ I / O パネル 2 3 1 0 のアクティビティ詳細パネル 1 8 3 0 は、図 2 1 を参照して記載されたように動作する。

## 【 0 0 9 5 】

他の可能な作業空間は、図 2 4 のスクリーンショット 2 4 0 0 で表されるようなエグゼクティブ作業空間である。エグゼクティブ作業空間 2 4 0 0 は、エグゼクティブパネル 2 4 1 0、フィルタ処理されたパイプラインパネル 1 8 2 0、及びアクティビティ詳細パネル 1 8 3 0 から成る。

## 【 0 0 9 6 】

エグゼクティブパネル 2 4 1 0 は、例えばグラフ及びタスクリストなどの総括データを提供する。フィルタ 2 4 2 0 は、何が表示されているかを調整するのに使用され得る。この例では、一番上のボックスは、何をフィルタ処理すべきかを決定し、一番下のボックスは、使用すべき値を設定する。結果は、セクタ 2 4 3 0 を用いてグループ化され得る。選択グルーピングヘッダ 2 4 4 0 は、グループが展開し又は折りたたむことを可能にする。パネル 2 4 5 0 においてアイテムを選択することは、関連するアクティビティ及びタスクを、フィルタ処理されたパイプラインパネル 1 8 2 0 及びアクティビティ詳細パネル 1 8 3 0 において表示させる。

## 【 0 0 9 7 】

エグゼクティブ作業空間 2 4 0 0 のフィルタ処理されたパイプラインパネル 1 8 2 0 は、図 2 0 を参照して記載されたように動作する。エグゼクティブ作業空間 2 4 0 0 のアクティビティ詳細パネル 1 8 3 0 は、図 2 1 を参照して記載されたように動作する。

## 【 0 0 9 8 】

最後の例となる作業空間は、図 2 5 のスクリーンショット 2 5 0 0 で表されるようなパイプラインビルダーである。パイプラインビルダー 2 5 0 0 は、テンプレートリスト 2 5 1 0 及び作業空間 2 5 2 0 から成る。

## 【 0 0 9 9 】

テンプレートリスト 2 5 1 0 は、プロジェクト又はワークフローを選択するためのフィールド 2 5 1 2 を提供する。選択されたプロジェクト又はワークフローのための関連するアクティビティテンプレートは、次いで、テンプレートリストにおいて提供される。それらの結果は更に、フィルタ機能性 2 5 1 4 を用いてフィルタ処理され得る。必要に応じて、新しいテンプレートは作成ツール 2 5 1 6 を用いて生成され得る。

## 【 0 1 0 0 】

作業空間 2 5 2 0 は、本開示の全体を通じて論じられたようにパイプラインモデルを構築するための機能性を提供する。本実施形態では、アクティビティ 2 5 3 0 のグラフィカル表現の入力又は出力を選択するにより、テンプレートリスト 2 5 1 0 における結果は、入力又は出力に関連するアセット記述子に基づきフィルタ処理される。

## 【 0 1 0 1 】

ここで開示される様々な実施形態は、ハードウェア、ファームウェア、ソフトウェア、又はそれらのあらゆる組み合わせとして実装され得る。更に、ソフトウェアは、好適に、プログラム記憶ユニット又はコンピュータ可読媒体において有形に具現されるアプリケーションプログラムとして実装される。アプリケーションプログラムは、あらゆる適切なアーキテクチャを有する機械にアップロードされ、そしてその機械によって実行されてよい。好適に、機械は、例えば 1 つ以上の中央演算処理装置 ( C P U )、メモリ、及び入出力インタフェースなどのハードウェアを備えるコンピュータプラットフォームにおいて実装される。コンピュータプラットフォームは、オペレーティングシステム及びマイクロ命令セットを更に含んでよい。ここで記載される様々なプロセス及び機能は、マイクロ命令コードの部分若しくはアプリケーションプログラムの部分のいずれか、又はそれらのあらゆる組み合わせであってよく、そのようなコンピュータ又はプロセッサが明示的に示されていようとなかろうと C P U によって実行されてよい。加えて、例えば追加のデータ記憶装置及び印刷装置などの様々な他の周辺装置がコンピュータプラットフォームに接続されてよい。

## 【 0 1 0 2 】

ここで挙げられている全ての例及び条件付き言語は、当該技術を促進することに発明者

10

20

30

40

50

によって寄与される概念及び実施形態の原理を読者が理解するのを助ける教育的な目的を意図され、そのような具体的に挙げられている例及び条件に制限されないと解釈されるべきである。更に、発明の原理、態様、及び様々な実施形態並びにそれらの具体例を挙げている本願中の全ての記述は、それらの構造上及び機能上の両方の等価物を包含するよう意図される。加えて、そのような等価物は、現在知られている等価物及び将来開発される等価物、すなわち、構造にかかわらず同じ機能を実行する開発されたあらゆる要素の両方を含むことが意図される。

#### 【 0 1 0 3 】

##### [ 関連出願の相互参照 ]

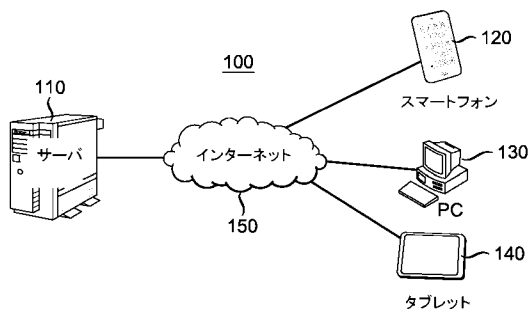
本願は、2013年1月23日付けで出願された米国特許仮出願第61/755892号及び2013年6月11日付けで出願された米国特許仮出願第61/833770号の優先権を主張する。なお、これらの米国出願は、その全文を参照により本願に援用される。

10

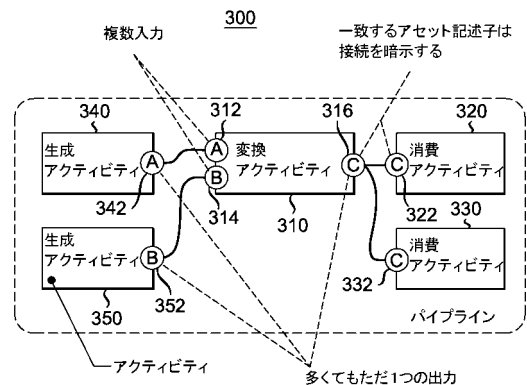
#### 【 0 1 0 4 】

本願はまた、同時に提出された“ASSET-DRIVEN WORKFLOW MODELING”、“FULFILLMENT TRACKING IN ASSET-DRIVEN WORKFLOW MODELING”、及び“METHOD AND APPARATUS FOR MAPPING PROCESS INFORMATION ONTO ASSET DATA”と題された出願に関連する。なお、これらの出願は、その全文を参照により本願に援用される。

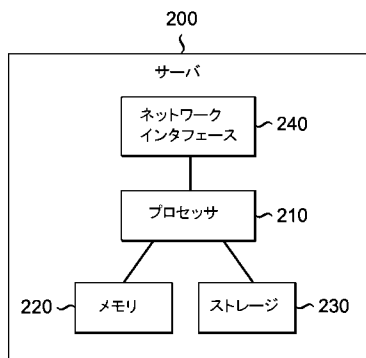
【 図 1 】



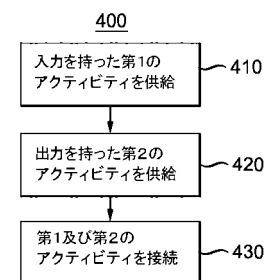
【 図 3 】



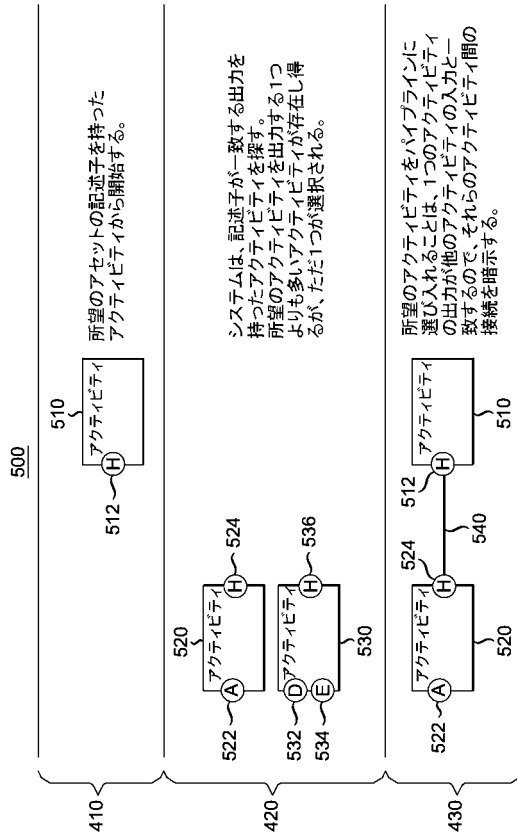
【 図 2 】



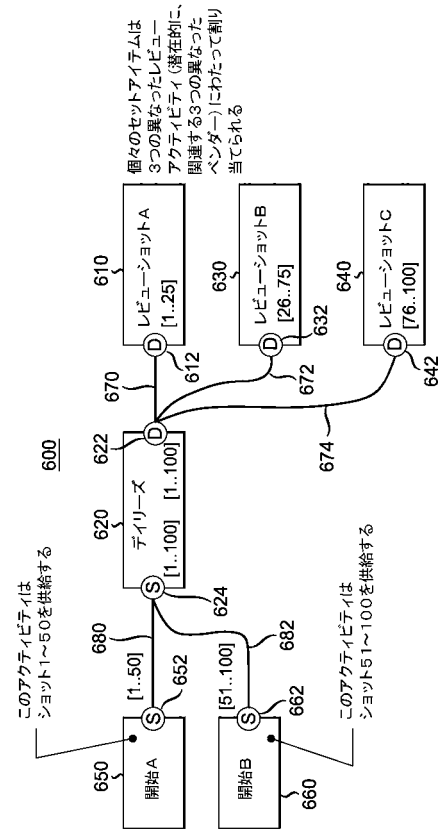
【 図 4 】



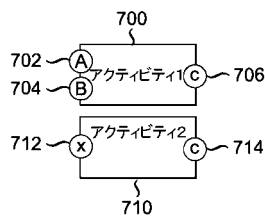
【 図 5 】



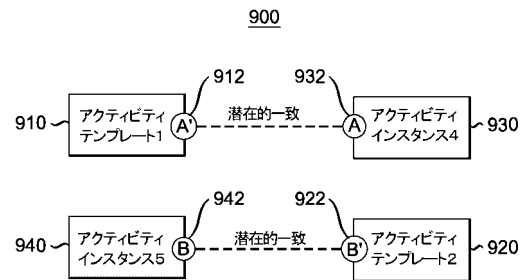
【 図 6 】



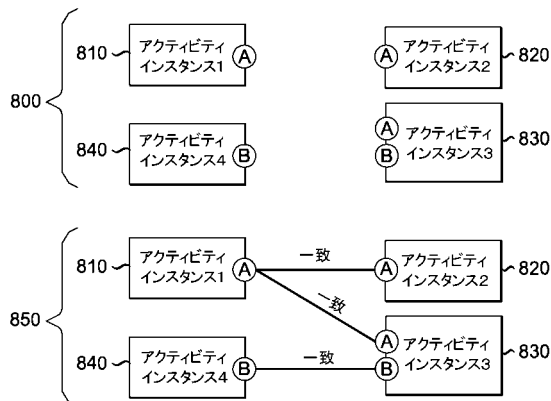
【 図 7 】



【 図 9 】



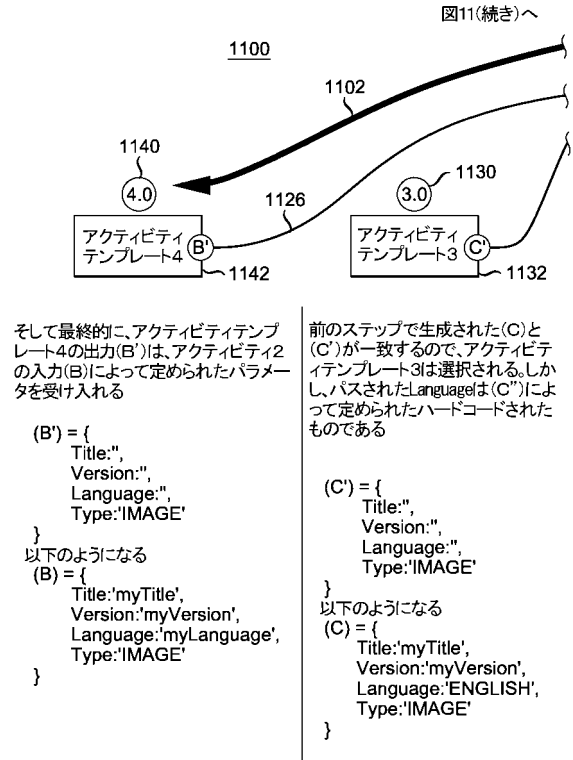
【 図 8 】



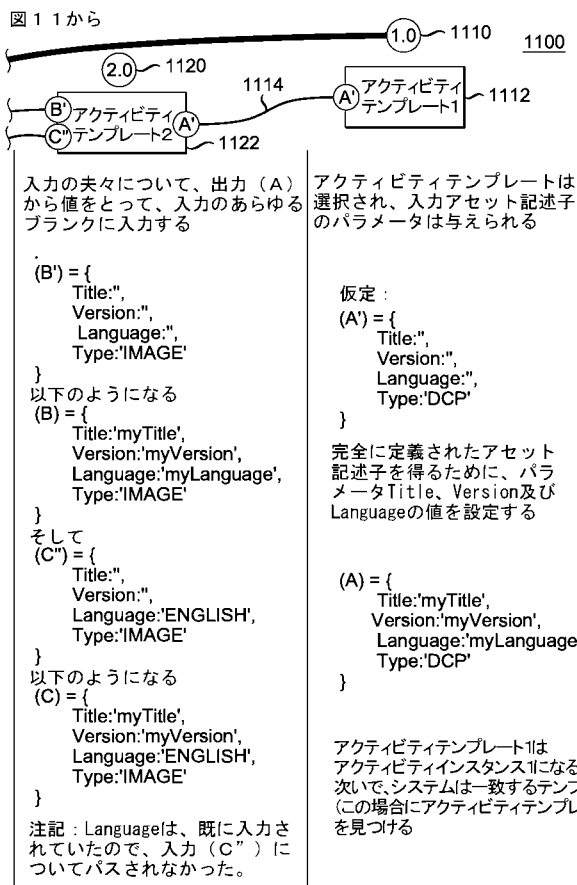
【図 10】

1000	完全に定義されているアセット記述子	パラメータ化されているアセット記述子
<pre>{   Title:"The Hobbit",   Version:"Trailer",   Type:"Netflix Encoding" }</pre>	一致 ブランクはファイルカードである	<pre>{   Title:",",   Version:",",   Type:"" }</pre>
<pre>{   Title:"The Hobbit",   Version:"Trailer",   Type:"Netflix Encoding" }</pre>	一致せず 名称/値の対の異なる組(余分)	<pre>{   Title:",",   Version:",",   Language:",",   Type:"Netflix Encoding" }</pre>
<pre>{   Title:"The Hobbit",   Version:"Trailer",   Type:"Netflix Encoding" }</pre>	一致 入力されている値が正確に一致	<pre>{   Title:",",   Version:",",   Type:"Netflix Encoding" }</pre>
<pre>{   Title:"The Hobbit",   Version:"Trailer",   Type:"SOMETHING ELSE" }</pre>	一致せず 入力されている値が正確に一致せず	<pre>{   Title:",",   Version:",",   Type:"Netflix Encoding" }</pre>

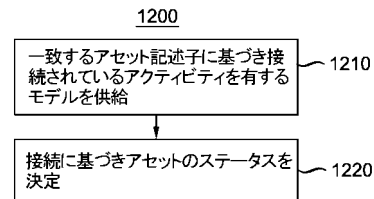
【図 11 A】



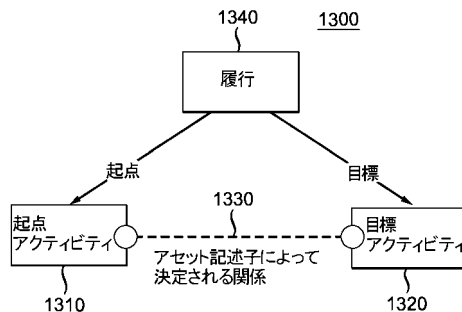
【図 11 B】



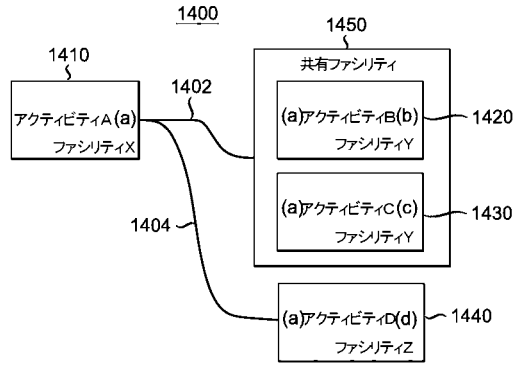
【図 12】



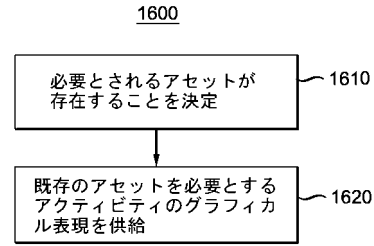
【図 13】



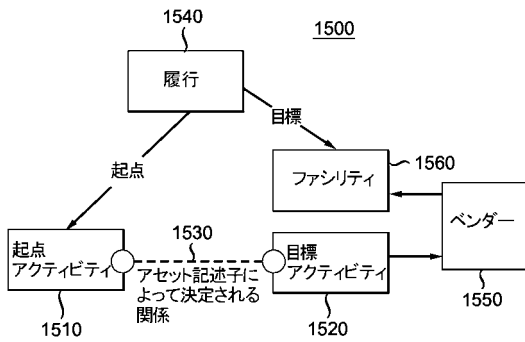
【図 14】



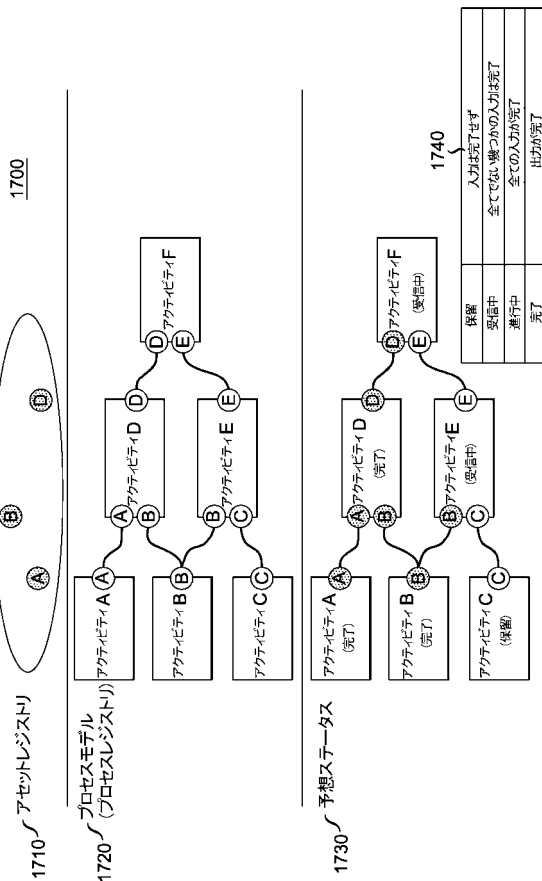
【図 16】



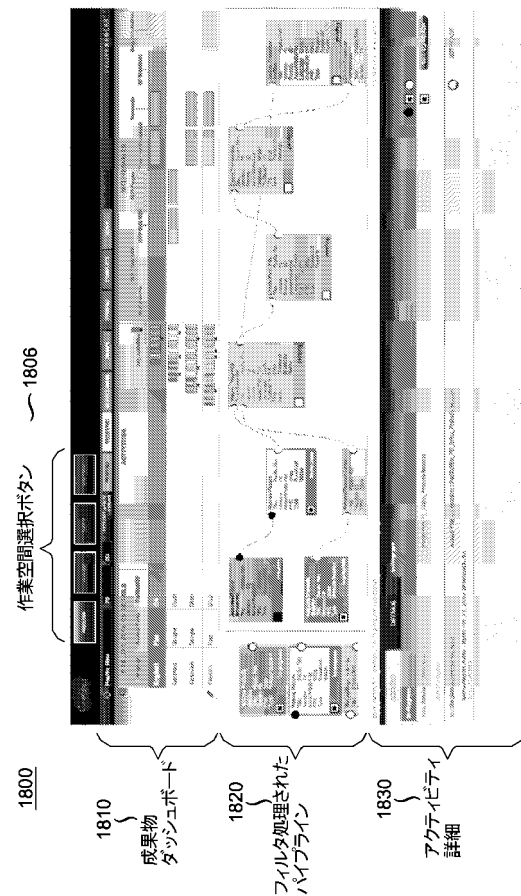
【図 15】



【図 17】

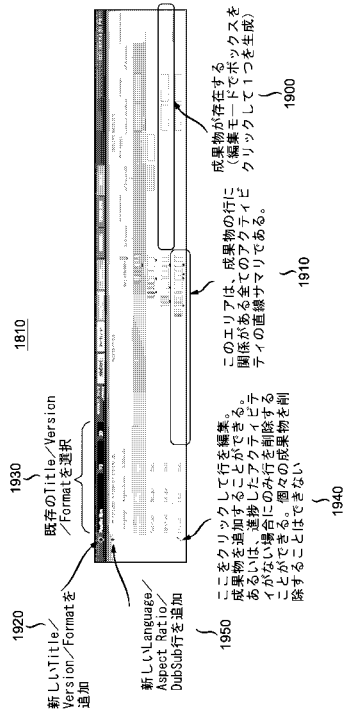


【図 18】

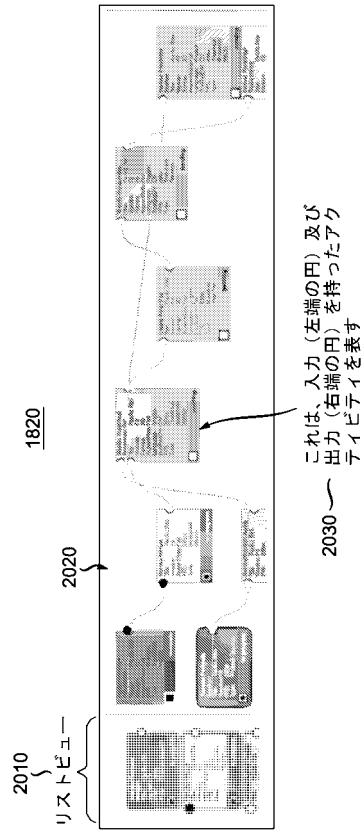




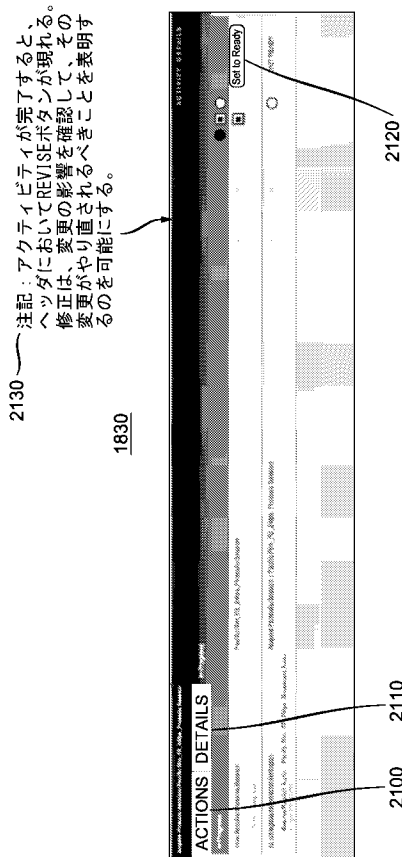
【図 19】



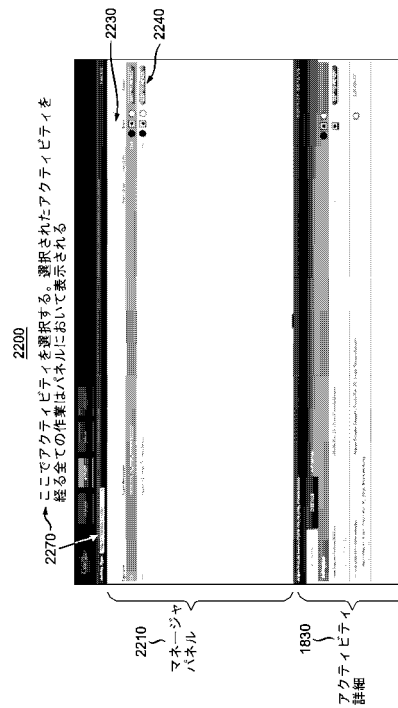
【図 20】



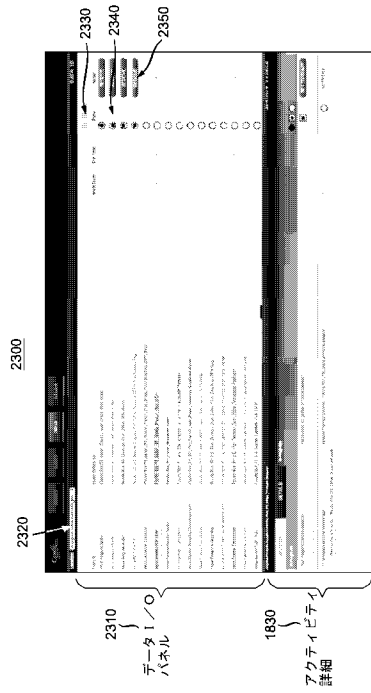
【図 21】



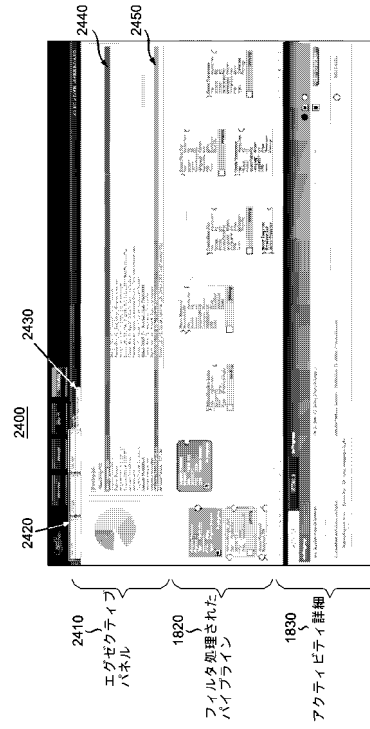
【図 22】



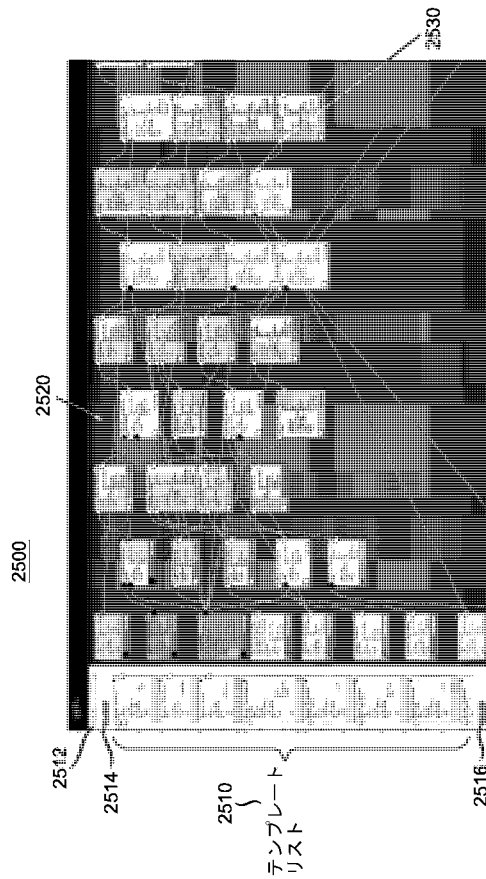
【図 2 3】



【図 2 4】



【図 2 5】



## 【 国際調査報告 】

INTERNATIONAL SEARCH REPORT		International application No. PCT/US 13/77213															
<b>A. CLASSIFICATION OF SUBJECT MATTER</b> IPC(8) - G06Q 10/00 (2014.01) USPC - 705/7.26 According to International Patent Classification (IPC) or to both national classification and IPC																	
<b>B. FIELDS SEARCHED</b> Minimum documentation searched (classification system followed by classification symbols) IPC(8): G06Q 10/00 (2014.01) USPC: 705/7.26 Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched IPC(8): G06Q 10/00 (2014.01) USPC: 705/7.11, 7.12, 7.26, 7.27; 345/418, 581, 630 (keyword limited; terms below) Electronic data base consulted during the international search (name of data base and, where practicable, search terms used) PatBase; Google(Web); Search terms used: asset business property workflow activity task model template input output link connector graphical representation display compatible matching available options menu type requirement rule																	
<b>C. DOCUMENTS CONSIDERED TO BE RELEVANT</b> <table border="1"> <thead> <tr> <th>Category*</th> <th>Citation of document, with indication, where appropriate, of the relevant passages</th> <th>Relevant to claim No.</th> </tr> </thead> <tbody> <tr> <td>X</td> <td>US 2004/0243611 A1 (Alexander et al.) 02 December 2004 (02.12.2004), entire document especially Fig. 1, 2A, 5, 8; abstract; para [0008], [0021], [0033], [0036], [0041]-[0048]</td> <td>1-5, 7, 9-11, 13, 15</td> </tr> <tr> <td>Y</td> <td>US 6,904,181 B1 (Becker et al.) 07 June 2005 (07.06.2005), entire document especially Fig. 2, 3; col. 8, ln. 40-55; col. 9, ln. 35-60</td> <td>6, 8, 12, 14</td> </tr> <tr> <td>A</td> <td>US 2007/0143736 A1 (Morlarty et al.) 21 June 2007 (21.06.2007), entire document especially</td> <td>6, 8, 12, 14</td> </tr> <tr> <td></td> <td></td> <td>1-15</td> </tr> </tbody> </table>			Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.	X	US 2004/0243611 A1 (Alexander et al.) 02 December 2004 (02.12.2004), entire document especially Fig. 1, 2A, 5, 8; abstract; para [0008], [0021], [0033], [0036], [0041]-[0048]	1-5, 7, 9-11, 13, 15	Y	US 6,904,181 B1 (Becker et al.) 07 June 2005 (07.06.2005), entire document especially Fig. 2, 3; col. 8, ln. 40-55; col. 9, ln. 35-60	6, 8, 12, 14	A	US 2007/0143736 A1 (Morlarty et al.) 21 June 2007 (21.06.2007), entire document especially	6, 8, 12, 14			1-15
Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.															
X	US 2004/0243611 A1 (Alexander et al.) 02 December 2004 (02.12.2004), entire document especially Fig. 1, 2A, 5, 8; abstract; para [0008], [0021], [0033], [0036], [0041]-[0048]	1-5, 7, 9-11, 13, 15															
Y	US 6,904,181 B1 (Becker et al.) 07 June 2005 (07.06.2005), entire document especially Fig. 2, 3; col. 8, ln. 40-55; col. 9, ln. 35-60	6, 8, 12, 14															
A	US 2007/0143736 A1 (Morlarty et al.) 21 June 2007 (21.06.2007), entire document especially	6, 8, 12, 14															
		1-15															
<input type="checkbox"/> Further documents are listed in the continuation of Box C. <input type="checkbox"/>																	
* Special categories of cited documents: "A" document defining the general state of the art which is not considered to be of particular relevance "E" earlier application or patent but published on or after the international filing date "L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified) "O" document referring to an oral disclosure, use, exhibition or other means "P" document published prior to the international filing date but later than the priority date claimed "T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention "X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone "Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art "&" document member of the same patent family																	
Date of the actual completion of the international search 19 July 2014 (19.07.2014)		Date of mailing of the international search report <b>22 AUG 2014</b>															
Name and mailing address of the ISA/US Mail Stop PCT, Attn: ISA/US, Commissioner for Patents P.O. Box 1450, Alexandria, Virginia 22313-1450 Facsimile No. 571-273-3201		Authorized officer: Lee W. Young PCT Helpdesk: 571-272-4900 PCT OSP: 571-272-7774															

## フロントページの続き

(81)指定国 AP(BW, GH, GM, KE, LR, LS, MW, MZ, NA, RW, SD, SL, SZ, TZ, UG, ZM, ZW), EA(AM, AZ, BY, KG, KZ, RU, TJ, TM), EP(AL, AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HR, HU, IE, IS, IT, LT, LU, LV, MC, MK, MT, NL, NO, PL, PT, RO, RS, SE, SI, SK, SM, TR), OA(BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, KM, ML, MR, NE, SN, TD, TG), AE, AG, AL, AM, AO, AT, AU, AZ, BA, BB, BG, BH, BN, BR, BW, BY, BZ, CA, CH, CL, CN, CO, CR, CU, CZ, DE, DK, DM, DO, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, GT, HN, HR, HU, ID, IL, IN, IR, IS, JP, KE, KG, KN, KP, KR, KZ, LA, LC, LK, LR, LS, LT, LU, LY, MA, MD, ME, MG, MK, MN, MW, MX, MY, MZ, NA, NG, NI, NO, NZ, OM, PA, PE, PG, PH, PL, PT, QA, RO, RS, RU, RW, SA, SC, SD, SE, SG, SK, SL, SM, ST, SV, SY, TH, TJ, TM, TN, TR, TT, TZ, UA, UG, US

(72)発明者 ウォーカー, マーク リロイ  
アメリカ合衆国, カリフォルニア州 9 1 3 8 4, キャスティーク, ケンブリッジ・アヴェニュー  
3 0 0 2 7

(72)発明者 クローソン, カート パトリック  
アメリカ合衆国, カリフォルニア州 9 1 5 0 4, バーバンク, デラウェア・ロード 5 3 7

(72)発明者 ランデル, スティーヴ  
アメリカ合衆国, カリフォルニア州 9 1 0 0 7, アーケーディア, ポートラ・ドライヴ 1 0 2  
0

Fターム(参考) 5L049 AA07 AA20

## 【要約の続き】

トを生成する適切なアクティビティは選択され得る。このプロセスは、プロセスパイプラインの開始に達するまで繰り返され得る。現実世界のプロセスパイプラインは、次いでモデルに基づき形成され得、モデルは、現実世界の生産パイプラインの状態を追跡するために使用され得る。そのような技術は、より効率的な追跡のためにアセットのセットの処理を更に支援することができる。